

Encaminament IP exterior

Protocol BGP4

(Solució)

Arquitectura i Protocols d'Internet

Grau en Enginyeria Telemàtica

Grau en Enginyeria de Sistemes de Telecomunicació

Sergio Machado
Anna Agustí
Frederic Raspall
Olga León

Departament d'Enginyeria Telemàtica
Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
Universitat Politècnica de Catalunya

CONTINGUTS

ENCAMINAMENT IP EXTERIOR. PROTOCOL BGP	1
1 L'estructura d'Internet.....	1
1.1 On som nosaltres?.....	2
2 El protocol BGP (Border Gateway Protocol)	3
2.1 Funcionament del protocol BGP	4
2.2 Format dels missatges BGP	5
2.3 Tipus de missatges BGP.....	6
2.3.1 Missatge Open.....	6
2.3.2 Missatge Update	7
2.3.3 Missatge Notification.....	9
2.3.4 Missatge Keepalive.....	10
2.3.5 Missatge Route-Refresh	10
2.4 Atributs BGP	11
2.4.1 ORIGIN	11
2.4.2 AS_PATH.....	11
2.4.3 NEXT_HOP.....	12
2.4.4 WEIGHT.....	12
2.4.5 LOCAL_PREFERENCE.....	13
2.4.6 MULTI_EXIT_DISCRIMINATOR.....	13
2.5 Diferències entre iBGP i eBGP	14
2.6 Selecció de rutes BGP	14
ACTIVITATS AL LABORATORI.....	16
Objectius de la pràctica.....	16
Part I. Funcionament bàsic del protocol BGP	19
Escratori P05-E01.....	19
Exercici 1. Configuració de l'encaminament interior dels sistemes autònoms.....	19
Exercici 2. Configuració de l'encaminament exterior als sistemes autònoms.....	26
Exercici 3. Intercanvi d'informació quan s'estableix una sessió BGP	38
Exercici 4. Canvis en la topologia de l'escenari.....	45
Part II. Atributs BGP.....	57
Escratori P05-E02.....	57
Exercici 1. Configuració de l'escenari.....	57
Exercici 2. Atribut MED.....	64
Exercici 3. Atribut LOCAL-PREFERENCE	79
Exercici 4. WEIGHT.....	90
Exercici 5. Ordre de preferència dels atributs.....	101
FIGURES	106
Escratori part I.....	106
Escratori part II.....	106

ENCAMINAMENT IP EXTERIOR. PROTOCOL BGP

1 L'ESTRUCTURA D'INTERNET

Internet s'ha definit moltes vegades com “una xarxa de xarxes”. Aquesta afirmació es correspon molt bé amb la realitat: Internet és la interconnexió dels anomenats “sistemes autònoms” (*Autonomous System*, AS), cadascun dels quals sol corresponder a un operador o una organització capaç de gestionar la seva xarxa. Els AS s’identifiquen unívocament per un número anomenat *Autonomous System Number* (ASN). Inicialment els ASN eren de 16 bits. A partir del 2006, es van estendre a 32 bits, permetent 4,294,967,296 possibilitats¹. A data de setembre de 2016 hi havia més de 90.000 ASes registrats². La gestió dels ASNs la fa l'IANA, que en delega l'assignació als RIRs (igual que amb l'espai d'adreses IP).

És important entendre que cada AS funciona de manera autònoma (per exemple, pot establir les seves polítiques internes d'encaminament) però s'ha de coordinar amb els ASs veïns per tenir accés a la resta de la xarxa. Un dels punts més importants és l'establiment d'acords de **peering**, que consisteixen en l'intercanvi de trànsit entre dos AS que tenen connexió directa (*peers*). Això permet als usuaris dels dos ASs comunicar-se directament entre sí i indirectament amb la resta d'Internet. Els acords d'intercanvi poden ser gratuïts (habitual si els dos AS intercanvien més o menys la mateixa quantitat de trànsit en els dos sentits) o bé de pagament (si els dos AS són de mides molt diferents). Alguns operadors IP no tenen usuaris domèstics ni empresarials, i estan especialitzats en oferir serveis de trànsit global; és a dir, els seus clients són exclusivament altres AS³.

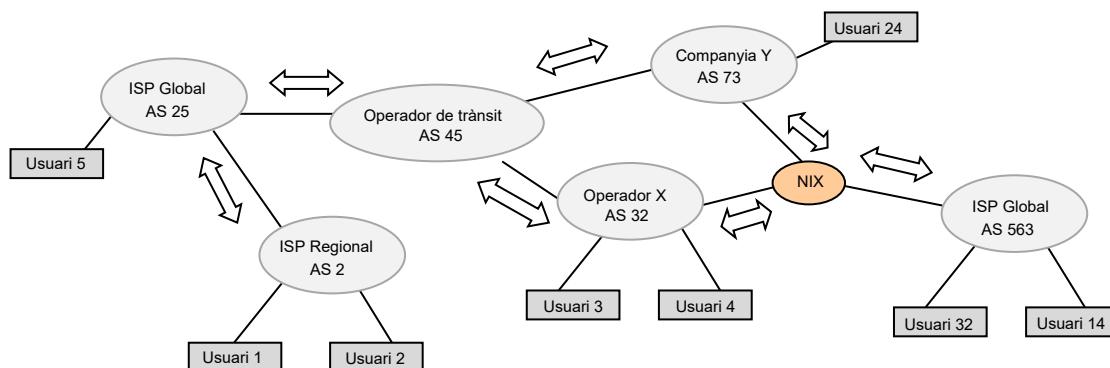


Figura 1. Internet vista com un conjunt d'AS que fan *peering*.

Evidentment, algunes organitzacions tenen més pes (per quantitat d'usuaris, mida de la seva xarxa, capacitat de transport, o simplement per raons econòmiques o polítiques) que altres. Per distingir la seva “importància” existeix una classificació informal que els assigna el nom Tier 1, 2 o 3:

- *Tier 1* és una xarxa IP (típicament un proveïdor de serveis o ISP, però no necessàriament sempre) que no ha de pagar pels seus acords de *peering* (de fet cobra pels acords amb els Tier 2 i 3).
- *Tier 2* és una xarxa IP que paga els seus enllaços amb Tier 1, però que manté acords d'intercanvi gratuït amb altres Tier 2, i que pot cobrar pels enllaços amb els Tier 3.
- *Tier 3* és una xarxa IP que paga per tots els seus enllaços amb Tier 2 (normalment no es connecten directament als Tier 1).

Vol dir això que els Tier 1 són el cor d'Internet, o que donen millor accés que els altres? No necessàriament. Hi ha Tiers 2 que són molt més grans que els Tier 1 i donen millor connectivitat. El món dels Tier sol estar dominat per

¹ Hi ha 1023 números reservats per usos locals o privats

² <http://www.potaroo.net/tools/asn32/>

³ Estrictament parlant tots els AS ofereixen serveis de trànsit.

l'economia i la política, no necessàriament per la tècnica. A l'enllaç https://es.wikipedia.org/wiki/Tier_1 teniu una llista dels Tier 1.

Els intercanvis de trànsit no són necessàriament sempre entre dos AS, sinó que es poden fer per part de molts ASs simultàniament a través dels anomenats Punt Neutres d'Intercanvi (*Neutral Internet eXchange Points*, NIX). Els NIX són normalment un punt de trobada (pagat per un conjunt d'AS) on hi ha un switch o un router "neutral" que connecta els enllaços que provenen de tots els AS que hi contribueixen. En aquests casos els intercanvis són gratuïts i els ASs només paguen la seva part del manteniment de les instal·lacions del NIX. Els dos NIXs més propers a nosaltres són el CATnix i l'Espanix⁴. El NIX més gran és (o sembla ser, ja que els NIX nord-americans no donen informació) l'AMS-IX a Amsterdam, amb 266 membres i un trànsit mig d'intercanvi de 166 Gbit/s⁵.

1.1 ON SOM NOSALTRES?

La UPC no és un sistema autònom, sinó que és una de les xarxes que pertanyen a l'AS13041 de l'Anella Científica⁶ gestionada pel CESCA (Centre de Supercomputació de Catalunya). Aquesta xarxa uneix totes les universitats catalanes i proporciona connectivitat amb RedIRIS (la xarxa universitària espanyola) i la resta d'Internet, tant a través de RedIRIS com per connexions directes a Europa i a EEUU. Existeix una web anomenada CIDR report⁷ que té dades globals sobre ASs i BGP, i permet generar informes sobre ASs concrets i veure les rutes des del punt de vista de l'AS que hostatja la plana web (l'AS4608). Si genereu l'informe per l'Anella (trobareu l'opció al peu de la plana web), obtindreu dades similars a aquestes:

```
13041 CESCA-AC , ES
Adjacency: 7 Upstream: 2 Downstream: 5
Upstream Adjacent AS list
AS12386 ASALPI Barcelona (SPAIN), ES
AS766 REDIRIS RedIRIS Autonomous System, ES
Downstream Adjacent AS list
AS33072 ISC-F-AS - Internet Systems Consortium, Inc., US
AS49638 CATNIX , ES
AS203457 ICFO , ES
AS15633 UOC-AS , ES
AS43115 PIC , ES
```

Això indica que l'Anella té dos ASs adjacents per sobre d'ella (*upstream*), és a dir, que es troben entre l'AS4608 i l'Anella, i que són RedIRIS i l'operadora Al-pi (que és qui gestiona l'Anella).

Si mirem els prefixos anunciats trobarem:

Rank	AS	Type	Originate	Addr Space (pfx)	Transit	Addr space (pfx)	Description			
954	AS13041	ORG+TRN	Originate:	312064 /13.75	Transit:	10240 /18.68	CESCA-AC , ES			
Rank	AS		AS Name		Current	Wthdw	Aggtc	Anncce	Redctn	%
22681	AS13041	CESCA-AC , ES			9	0	0	9	0	0.00%
		Prefix	AS Path							
		84.88.0.0/16	4608 1221 4637 174 766 766 766 766 766 13041							
		84.89.0.0/18	4608 1221 4637 5511 12479 12386 13041							
		84.89.128.0/17	4608 1221 4637 174 766 766 766 766 766 13041							
		147.83.0.0/16	4608 1221 4637 174 766 766 766 766 766 13041							
		158.109.0.0/16	4608 1221 4637 174 766 766 766 766 766 13041							
		161.116.0.0/16	4608 1221 4637 174 766 766 766 766 766 13041							
		192.94.163.0/24	4608 1221 4637 174 766 766 766 766 766 13041							
		192.101.162.0/24	4608 1221 4637 174 766 766 766 766 766 13041							
		193.242.98.0/24	4608 1221 4637 5511 12479 12386 13041							

La xarxa 147.83.0.0/16 us hauria de ser familiar; és la de la UPC. La 158.109.0.0/16 és la de la UAB, la 161.116.0.0/16 és la de la UB, etc. Fixeu-vos que apareixen els camins des de l'AS4608 fins l'Anella i que alguns valors (el 766 de RedIRIS) apareix repetit vàries vegades. Això és una tècnica anomenada *AS prepending* i serveix per penalitzar aquesta ruta fent que la mètrica mesurada en salts d'ASs sigui més alta de la que li pertocaria (compte, perquè aquesta no és la única mètrica utilitzada per escollir la millor ruta en BGP, però sí una de les possibles).

⁴ <http://www.catnix.net> , <http://www.espanix.net>

⁵ http://en.wikipedia.org/wiki/List_of_Internet_Exchange_Points_by_size

⁶ <http://www.cesca.es/comunicaciones/anella.html>

⁷ <http://www.cidr-report.org/>

2 EL PROTOCOL BGP (BORDER GATEWAY PROTOCOL)

El *Border Gateway Protocol* (BGP) és un protocol d'encaminament entre sistemes autònoms. Es va definir amb l'objectiu de resoldre determinats problemes del seu antecessor, l'*Exterior Gateway Protocol* (EGP). La funció principal d'un sistema que corre BGP és intercanviar informació amb altres sistemes BGP per tal de conèixer el camí cap a diferents xarxes. Aquesta informació inclou una llista amb els sistemes autònoms (ASs) que els missatges travessen. Aquesta llista serveix per construir un graf de la connectivitat entre els diferents ASs que permet evitar bucles i establir certes polítiques d'encaminament a nivell d'AS.

Segons la definició clàssica, un sistema autònom és un conjunt de *routers* sota una administració tècnica única, que utilitzen un protocol d'encaminament interior (IGP) amb una mètrica comuna per encaminar els paquets dins l'AS, i un protocol d'encaminament exterior per encaminar els paquets cap a d'altres ASs. No obstant, a la pràctica, és comú que un únic AS utilitzi més d'un protocol d'encaminament interior i, sovint, més d'una mètrica dins l'AS. Per això, el terme AS es pot utilitzar per referir-se a un sistema que, malgrat utilitzar diferents IGPs i varietat de mètriques, es pot percebre des de fora com una unitat d'encaminament coherent. Les xarxes corporatives, com les universitàries o les empresarials, normalment utilitzen un protocol d'encaminament interior (Interior Gateway Protocol o IGP) com el RIP o l'OSPF per intercanviar informació d'encaminament entre les seves xarxes, mentre els usuaris connectats a algun ISP i els propis ISPs entre si, utilitzen el BGP, o encaminament estàtic, per intercanviar les rutes d'usuari i d'ISP. L'elecció d'utilitzar BGP o encaminament estàtic ve donada en funció dels requeriments de cada cas, utilitzant BGP només en aquells casos en els quals és necessari un encaminament dinàmic. Quan el BGP s'utilitza entre sistemes autònoms diferents, rep el nom d'*External BGP* (EBGP). Si un proveïdor de servei l'utilitza per intercanviar rutes dins un sistema autònom, aleshores s'anomena *Interior BGP* (IBGP). La figura 2 il·lustra aquesta distinció:

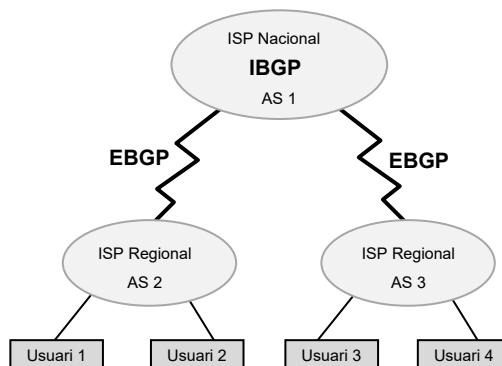


Figura 2. Interior BGP i External BGP

En un escenari real, un ISP manté els protocols d'encaminament següents:

- OSPF o IS-IS per a tenir coneixement de totes les xarxes pròpies de l'ISP. Aquests protocols són del tipus IGP i tenen unes característiques de ràpida convergència i de jerarquizació mitjançant la creació d'àrees.
- BGP Extern (eBGP) per a intercanviar rutes amb els ISPs veïns i clients connectats amb BGP.
- BGP Intern (iBGP) per a comunicar les rutes dels ISPs veïns a tot el sistema autònom de l'ISP.

BGP és un protocol d'encaminament molt robust i escalable, com ho evidencia el fet que sigui el protocol d'encaminament utilitzat a Internet. Actualment, les taules d'encaminament BGP d'Internet tenen més de 700000 prefixes⁸. Per aconseguir escalabilitat a aquest nivell, el BGP utilitza diversos paràmetres de ruta, anomenats atributs, per definir polítiques d'encaminament i mantenir un entorn d'encaminament estable.

La versió 4 del protocol està descrita al *Request for Comments* (RFC) 1771. BGP4 proporciona un conjunt de mecanismes per suportar encaminament entre dominis sense classes (*Classless InterDomain Routing* o CIDR), això significa que es poden advertir prefixes IP eliminant el concepte tradicional de les classes. Per exemple, assumim que

⁸ Dades de: <http://www.cidr-report.org/>

un ISP és propietari del bloc d'adreses IP 192.10.x.x de l'espai d'adreses de classe C tradicional. Aquest bloc conté 256 blocs d'adreses de classe C, del 192.10.0.x/24 al 192.168.255.x/24. Assumim també que aquest ISP assigna un bloc de classe C a cadascun dels seus usuaris. Sense CIDR, l'ISP informarà de 256 blocs de classe C als seus BGP peers. Amb CIDR, el protocol BGP pot agregar tot l'espai d'adreses i informar d'un únic bloc, 192.10.x.x/16. Aquest bloc té la mateixa mida que un bloc de l'espai d'adreses de classe B tradicional. La distinció de classes ha quedat obsoleta amb la introducció del CIDR.

Un exemple de la potència del CIDR sobre la Internet simplificada de la Figura 1: suposant que l'usuari 1 té assignada la xarxa 192.168.10.0/26, l'usuari 2 la 192.168.10.64/26, i l'usuari 5 la 192.168.10.128/25, l'AS2 podria agregar les seves dues xarxes i anunciar cap a l'AS25 només el prefix 192.168.10.0/25, i a la seva vegada l'AS25 podria anunciar cap a l'AS45 només el prefix 192.168.10.0/24.

Per caracteritzar el conjunt de decisions que es poden prendre utilitzant BGP, cal entendre la manera com un *router* BGP es comunica amb els seus *peers* (altres *routers* BGP veïns). Aquest mecanisme segueix el paradigma d'encaminament salt a salt (*hop by hop*) i pot suportar qualsevol control de policia conforme amb tal paradigma. No obstant, certes polítiques no es poden duu a terme amb el paradigma salt a salt i requereixen tècniques específiques com ara l'encaminament de font. El BGP es pot adaptar en alguns d'aquests casos, per exemple, prohibint que un AS envii trànsit a un AS veí seguint un camí diferent del seguit pel trànsit originat a l'AS veí.

BGP ha de córrer sobre un protocol de transport fiable. Això elimina la necessitat d'implementar explícitament mecanismes de fragmentació, retransmissió, reconeixement i ordenació. Qualsevol mecanisme d'autenticació utilitzat pel protocol de transport es pot utilitzar com a complement dels propis mecanismes d'autenticació BGP. El mecanisme de notificació d'errors que utilitza BGP assumeix que el protocol de transport realitza un tancament ordenat, és a dir, que totes les dades en espera es transmeten abans del tancament de la connexió.

El BGP utilitzà TCP com a protocol de transport. El TCP compleix els requisits del BGP com a protocol de transport i està present en gairebé tots els *routers* i *hosts* comercials. El BGP utilitzà el port número 179 per establir les seves connexions. Els *routers* BGP veïns intercanvien tota la taula d'encaminament BGP quan s'estableix la connexió TCP entre veïns per primera vegada. Quan es detecten canvis en la taula d'encaminament, els *routers* BGP envien als seus veïns només les rutes afectades pels canvis. No s'envien actualitzacions periòdiques de les taules d'encaminament, i les actualitzacions d'encaminament BGP només informen del camí òptim cap a cadascuna de les xarxes destí.

2.1 FUNCIONAMENT DEL PROTOCOL BGP

Tot i que es va dissenyar com a protocol d'encaminament entre ASs, el BGP es pot utilitzar tant dins com entre sistemes autònoms. Dos veïns BGP que es comuniquen entre ASs han de residir a la mateixa xarxa física. Dos routers BGP que es comuniquen dins un mateix AS s'han d'assegurar que tenen un punt de vista consistent de l'AS del que formen part. Tots els routers BGP d'un mateix sistema autònom han d'establir, a nivell lògic, una connexió mallada entre sí.

Alguns ASs són veritables canals de pas, és a dir, alguns ASs porten trànsit que no s'ha originat dins l'AS i que no va destinat a cap xarxa de l'AS. El protocol BGP ha d'interactuar amb qualsevol protocol d'encaminament intern existent dins aquests ASs de pas.

Inicialment, dos sistemes formen una connexió a nivell de transport entre ells i s'intercanvien missatges per obrir i confirmar els paràmetres de la connexió. El flux de dades inicial consisteix en la taula d'encaminament BGP completa i s'envien actualitzacions progressives a mesura que les taules d'encaminament canvien, utilitzant missatges Update. Els missatges d'Update contenen una parella formada per una etiqueta de xarxa i uns atributs que la defineixen, entre els quals hi ha un camí d'ASs. El camí d'ASs conté una cadena de caràcters que identifiquen els diferents ASs que cal travessar per arribar a la xarxa especificada.

Per exemple, a la Internet simplificada de la Figura 1, suposeu que l'usuari 5 que penja de l'AS 25 té assignada la xarxa 192.0.1.0/24 i que l'AS 25 anuncia aquest prefix pels seus enllaços BGP. Aleshores, a l'AS563 li arribaria un anunci del prefix 192.0.1.0/24 amb l'AS Path 73, 45, 25 que li hauria enviaït el router de la companyia Y de l'AS 73 i un anunci del mateix prefix amb l'AS Path 32, 45, 25 provinent de l'operador X de l'AS 32. Noteu que el NIX no és cap AS, sinó un concentrador d'acords de peering, i que per tant no apareix en la ruta.

Malgrat que un *router* BGP pot mantenir a la taula d'encaminament BGP tots els camins possibles per arribar a una determinada xarxa, als missatges *d'Update* només informa del camí primari (**best**). En el cas de BGP la selecció de les rutes no es basa en una mètrica única sinó que s'estableix una llista de criteris que s'exploren de forma ordenada per tal de seleccionar la ruta per accedir a un determinat prefix.

BGP no incorpora cap mecanisme d'actualització periòdica de les taules d'encaminament, sinó que estableix una sessió TCP entre els *peers* i delega la responsabilitat de l'entrega de tots els paquets a aquest protocol, tot i que existeixen diversos mecanismes per a forçar el re-enviament de la taula d'encaminament del *peer*. Els missatges de *Keepalive* s'envien periòdicament per assegurar el manteniment de la connexió.

Els missatges de notificació (**Notification**) s'envien en resposta a errors o a condicions especials. Si una connexió troba una condició d'error, s'envia un missatge de notificació i es tanca la connexió.

Els hosts que parlen BGP no cal que siguin *routers*. Un host que no realitza encaminament pot intercanviar informació d'encaminament amb *routers* via EGP o, fins i tot, a través del protocol d'encaminament interior. Aquest host pot utilitzar BGP per intercanviar informació d'encaminament amb un *router* frontera d'un altre sistema autònom.

Si un AS particular té múltiples veïns BGP i proporciona un servei de trànsit per altres ASs, cal tenir un punt de vista consistent sobre l'encaminament dins l'AS. El punt de vista coherent de les rutes internes l'ha de proporcionar el protocol d'encaminament interior. Utilitzant un conjunt de controls de policia comuns, els *routers* BGP poden arribar a un acord sobre quin *router* frontera s'utilitzarà com a punt d'entrada i de sortida per a cada destí particular fora de l'AS. Aquesta informació es pot transmetre als *routers* interiors del domini utilitzant el protocol d'encaminament interior. S'ha d'anar en compte per tal d'assegurar-se que els *routers* interns tenen la informació de trànsit actualitzada abans que els *routers* BGP anunciïn als altres ASs que es pot proporcionar el servei de trànsit.

Les connexions entre *routers* BGP de diferents ASs s'anomenen enllaços exteriors. Les connexions BGP entre *routers* d'un mateix AS s'anomenen enllaços interns. De manera similar, un *peer* amb un *router* d'un AS extern s'anomena *peer* extern i un *peer* entre *routers* d'un mateix sistema autònom s'anomena *peer* intern.

2.2 FORMAT DELS MISSATGES BGP

Els missatges BGP tenen una capçalera comuna de 19 bytes de longitud que conté tres camps:

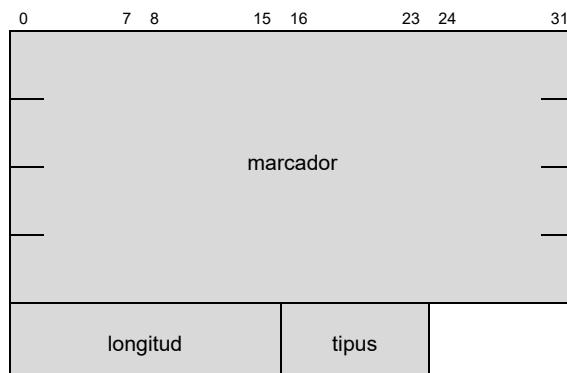


Figura 3. Capçalera paquets BGP

- **Marcador.** S'utilitza per detectar la pèrdua de sincronisme entre un parell de peers BGP i per autenticar els missatges BGP entrants. Si el tipus de missatge és *Open* o si és un missatge *Open* indicant com a paràmetre opcional que no hi ha autenticació, tots els bits són 1. En altre cas, el valor del marcador es pot predir utilitzant el mecanisme d'autenticació.
 - **Longitud.** Conté la longitud total del missatge, inclosa la capçalera, en bytes. Com a mínim ha de valer 19 i no pot ser superior a 4096.
 - **Tipus.** Especifica el tipus de missatge BGP: *Open*, *Update*, *Notification* i *Keepalive*.

2.3 TIPUS DE MISSATGES BGP

Al RFC 1771 s'especifiquen quatre tipus de missatges: Open, Update, Notification i Keepalive.

Al RFC 2918 s'especifica el missatge Route-Refresh.

2.3.1 MISSATGE OPEN

Després d'establir la connexió TCP, el primer missatge que envia el *router* de cadascuna de les dues bandes de la connexió, és un missatge d'*Open*. Si s'accepta el missatge, el receptor envia un missatge *Keepalive* confirmant el missatge d'*Open*. Després de la confirmació del missatge *Open*, es poden intercanviar missatges d'*Update*, *Keepalive* i *Notifications*.

A banda de la capçalera comuna dels paquets BGP, els missatge *Open* defineixen diversos camps:

- **Versió.** Proporciona el número de versió del protocol BGP i permet al receptor comprovar si està corrent la mateixa versió del protocol que l'emissor.
- **Sistema autònom.** Proporciona el número d'AS de l'emissor.
- **Hold-time.** Número màxim de segons que poden transcorre sense rebre un missatge abans de considerar que el transmissor ha mort.
- **Identificador BGP.** Identificador BGP del *router*. Correspon a una adreça IP del *router* que es fixa al principi i que s'utilitza per a totes les interfícies locals del *router* i per cada *peer* BGP.
- **Longitud dels paràmetres opcionals.** Longitud total del camp de paràmetres opcionals, en bytes.
- **Paràmetres opcionals.** Camp que conté una llista de paràmetres opcionals. Cada paràmetre conté la tríade:
 - **Tipus.** 1 byte que identifica el paràmetre de forma unívoca.
 - **Longitud.** 1 bytes que indica la longitud del camp valor del paràmetre, en bytes.
 - **Valor.** Camp de longitud variable que s'interpreta en funció del tipus de paràmetre.

Un dels paràmetres opcionals que es pot incloure als missatges *Open* és la informació d'autenticació. Aquest paràmetre conté dos camps dins el camp valor del paràmetre optional:

- **Codi d'autenticació.** Indica el tipus d'autenticació utilitzada.
- **Dades d'autenticació.** Conté les dades de l'autenticació corresponent.

La figura 4 mostra el format dels missatges *Open*:

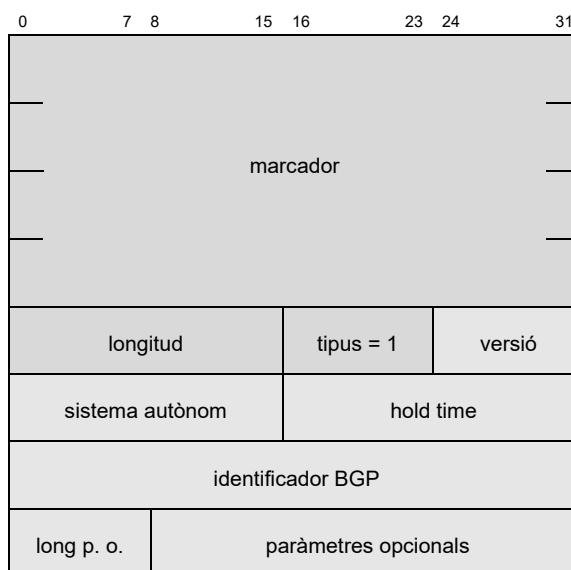


Figura 4. Missatge Open

2.3.2 MISSATGE UPDATE

Els missatges *Update* s'utilitzen per intercanviar informació d'encaminament entre *peers* BGP. Aquesta informació es pot utilitzar per crear un graf que descriu les relacions entre els diferents sistemes autònoms. Alicant determinades normes, es poden detectar i evitar bucles i altres anomalies de l'encaminament entre ASs. Els missatges d'*Update* s'utilitzen per descriure una xarxa (o un conjunt de xarxes resultat d'una agregació) accessible per un *peer* o un conjunt de xarxes que han deixat de ser accessibles (*withdraw unfeasible routes*) a través d'aquest *router*. Les *withdraw routes* serveixen per eliminar les rutes que han deixat de ser vàlides sense haver d'esperar que venci cap temporitzador.

Els missatges *Update* tenen la següent estructura:

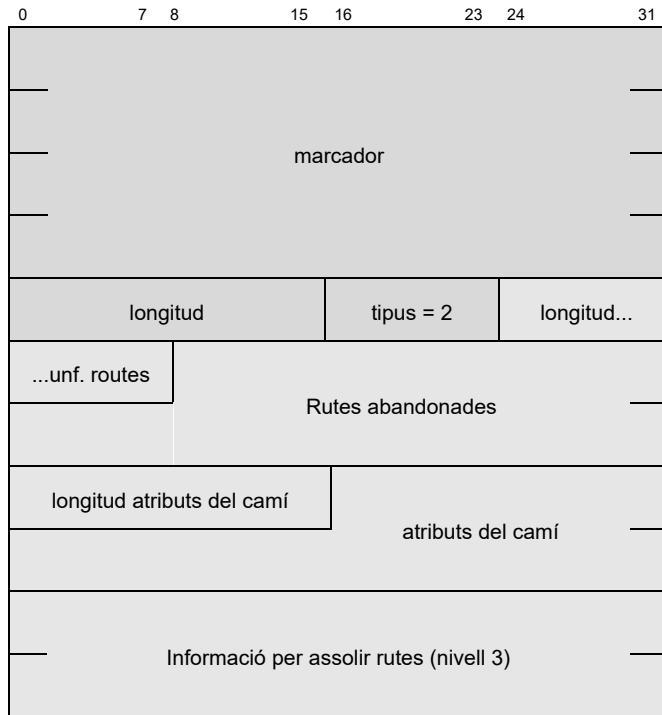


Figura 5. Missatges Update

- **Longitud Unfeasible Routes (Unfeasible routes length).** 2 bytes que indiquen la longitud total del camp *withdraw routes*. Si val 0 indica que no hi ha cap camp de *withdraw routes*.
- **Rutes perdudes o abandonades (Withdrawn routes).** Camp de longitud variable que conté la llista de prefixes d'adreses IP de les rutes que han deixat de ser vàlides. Cada prefix es codifica seguint una notació com la de la figura següent:

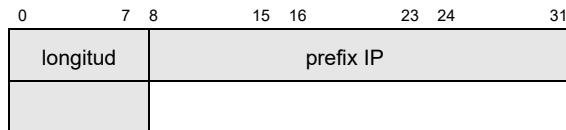
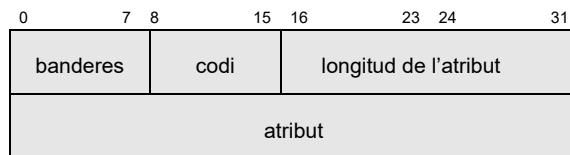


Figura 6. Camp *Withdraw routes*

- **Longitud.** 1 byte que indica la longitud, en bits, del prefix d'adreça IP. Si val 0 indica que el prefix engloba totes les adreces IP (i va acompanya d'un prefix de longitud 0).
- **Prefix IP.** Conté el prefix d'adreses IP acompañyat d'un conjunt de bits de padding per tal de fer coincidir la longitud del camp amb un múltiple de 8 bits (la longitud màxima de cada prefix IP és de 4 bytes).

- **Longitud dels atributs del camí.** 2 bytes que indiquen la longitud total, en bytes, del camp d'atributs del camí.
- **Atributs del camí.** Camp de longitud variable que conté informació de les rutes següent la notació següent:

Figura 7. Camp *Path Attributes*

- **Banderes.** Els primers 4 bits del camp de banderes indiquen:
 - **Bit opcional (bit 0).** Defineix si l'atribut és opcional, és a dir, no obligatori (bit a 1) o ben coneugut, és a dir, *well-known* (bit a 0).
 - **Bit transitiu (bit 1).** Indica si es tracta d'un atribut opcional és transitiu (bit a 1) o no transitiu (bit a 0). Per atributs ben coneuguts, val 1.
 - **Bit parcial (bit 2).** Defineix si la informació de l'atribut opcional transitiu és parcial (bit a 1) o completa (bit a 0). Per atributs ben coneuguts o per atributs opcionals no transitius, val 0.
 - **Bit d'extensió de longitud (bit 3).** Indica si el camp de longitud de l'atribut consta d'un byte (bit a 0) o de dos bytes (bit a 1). Només es pot utilitzar un camp de longitud de dos bytes si la longitud del camp d'atribut superar el valor 255.

Els darrers 4 bits del camp de banderes no s'utilitzen i s'han de posar a 0.

- **Codi.** 1 byte que conté el valor del codi corresponen al tipus d'atribut.

<u>Codi</u>	<u>Opció</u>
1	ORIGIN
2	AS_PATH
3	NEXT_HOP
4	MULTI_EXIT_DISCRIMINATOR
5	LOCAL_PREFERENCE
6	ATOMIC_AGGREGATE
7	AGGREGATOR

El significat de cadascun dels codis possibles, s'explica a l'apartat següent.

- **Longitud del camp d'atribut.** 1 o 2 bytes (en funció del bit 3 de les banderes) que indiquen la longitud del camp d'atribut, en bytes.
- **Atribut.** Valor de l'atribut, que cal interpretar en funció de les banderes i del codi d'atribut corresponent.
- **Informació d'assoliment de rutes.** Camp de longitud variable que conté una llista de prefixes IP de les rutes assolibles. La longitud d'aquest camp no apareix explícitament però es pot deduir utilitzant la fórmula:

$$\text{longitud del camp d'informació} = \begin{aligned} & \text{longitud del missatge d'Update} - \\ & 19 \text{ bytes de la capçalera BGP comuna} - \\ & 2 \text{ bytes del camp de longitud de les withdraw routes} - \\ & \text{longitud del camp withdraw routes} - \\ & 2 \text{ bytes del camp de longitud dels atributs del camí} - \\ & \text{longitud del camp d'atributs} \end{aligned}$$

La informació d'aquest camp apareix en el format següent:

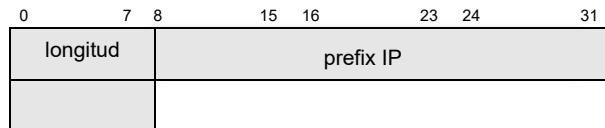


Figura 8. Camp *Network Layer Reachability Information*

- **Longitud.** 1 byte que indica la longitud, en bits, del prefix d'adreça IP. Si val 0 indica que el prefix engloba totes les adreces IP (i va acompanya d'un prefix de longitud 0).
- **Prefix IP.** Conté el prefix d'adreses IP acompañat d'un conjunt de bits de farciment (*padding*) per tal de fer coincidir la longitud del camp amb un múltiple de 8 bits (la longitud màxima de cada prefix IP és de 4 bytes).

2.3.3 MISSATGE NOTIFICATION

Els missatges de notificació s'envien quan es detecten condicions d'error i un router desitja dir-li a un altre per què vol tancar la connexió entre ells.

A banda de la capçalera comuna, els missatge de notificació contenen tres camps:

- **Codi d'error.** Indica el tipus d'error i pot ser:
 - **Message header error.** Indica un problema amb la capçalera del missatge BGP, és a dir, un error al camp de longitud, de marcador o de tipus.
 - **Open message error.** Indica un problema amb el missatge d'obertura com ara un número de versió no suportada, un número d'AS o una adreça IP no acceptable o un codi d'autenticació incorrecte.
 - **Update message error.** Indica un problema amb el missatge *d'Update*, com ara una llista d'atributs incorrecte o un atribut de *next hop* no vàlid.
 - **Hold time expired.** Indica que ha expirat el temps de hold time, després del qual un node BGP es declara mort.
- **Subcodi d'error.** Aporta informació més específica sobre l'error que s'està notificant. Cada codi d'error pot tenir un o més subcodis associats. Si no hi ha cap subcodi d'error definit, aleshores es fixa el valor a 0.
- **Dades d'error.** Camps de longitud variable que s'utilitza per diagnosticar la raó de la notificació. El contingut depèn del codi i el subcodi de l'error.

La figura 9 mostra el format dels missatges de notificació:

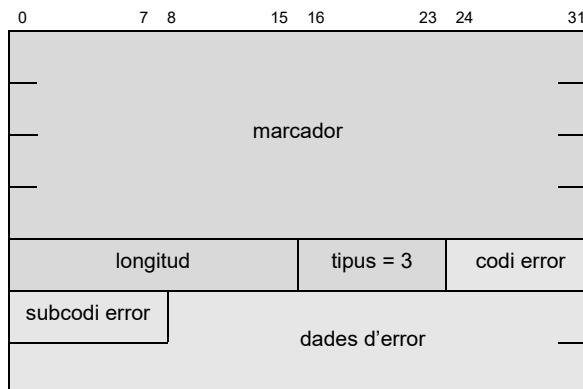


Figura 9. Missatge Notification

2.3.4 MISSATGE KEEPALIVE

Els missatges *Keepalive* no contenen camps addicionals darrera de la capçalera BGP. Aquests missatges s'envien periòdicament per evitar que expiri el temps de hold time. Un temps raonable entre missatges *Keepalive* és un terç del l'interval de hold time, però, com a màxim, es poden enviar a raó d'un missatge per segon. Si l'interval de hold time es negocia a valor 0, no s'envien missatges *Keepalive*.



Figura 10. Missatge Keepalive

2.3.5 MISSATGE ROUTE-REFRESH

Quan es configuren noves polítiques entre peers BGP cal aplicar-les sobre les rutes apreses d'un determinat veí. Una possible manera de fer-ho és reiniciant totalment la sessió entre dos peers. Aquest mecanisme s'anomena “**hard reset**”. El temps per recuperar la sessió entre dos veïns BGP pot ser considerablement elevat (depenent de la longitud de la taula bgp). Per això, el “hard reset” es recomana només com a última opció. El mecanisme de “**configuració suau**” és una possible alternativa que es basa en mantenir dues taules diferents: la Adj-RIB-in, que conté tots els prefixes abans d'aplicar cap política, i la loc-RIB, que és la taula bgp que queda després d'aplicar les polítiques que pertoqui. La principal desavantatge de la configuració suau és que requereix més memòria per emmagatzemar les dues taules i més CPU per processar-les.

El mecanisme de “**soft reset**” basat en l'intercanvi de missatges Route refresh, descrit al RFC 2918, es va definir per tal d'estandarditzar el mecanisme de “configuració suau”. No obstant, no requereix emmagatzemar dues taules i, per tant, estalvia consum de CPU i de memòria. Els dos mecanismes són mútuament excloents, i els peers s'intercanvien informació per saber quin mecanisme s'ha d'utilitzar.

La figura 11 mostra el format dels missatges *Route-Refresh*:

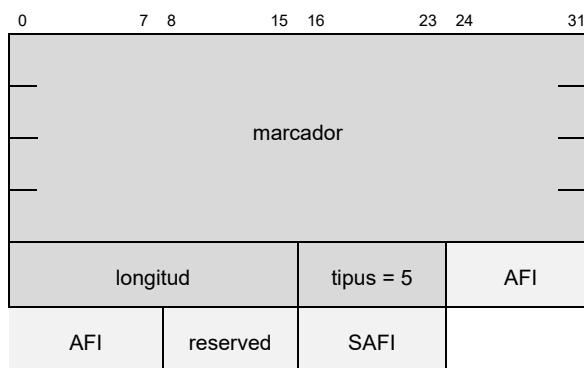


Figura 11. Missatge Route-Refresh

2.4 ATRIBUTS BGP

A diferència de protocols com per exemple el RIP, els prefixes intercanviats via BGP van acompanyats de paràmetres addicionals que permeten, per exemple, determinar el millor camí cap a un prefix quan un *router* coneix més d'un camí per arribar-hi. Aquestes propietats són conegudes com **atributs** BGP i és necessari conèixer el seu significat per tal d'entendre la seva influència en el procés de selecció de rutes BGP. Aquest apartat descriu els atributs més importants del BGP.

2.4.1 ORIGIN

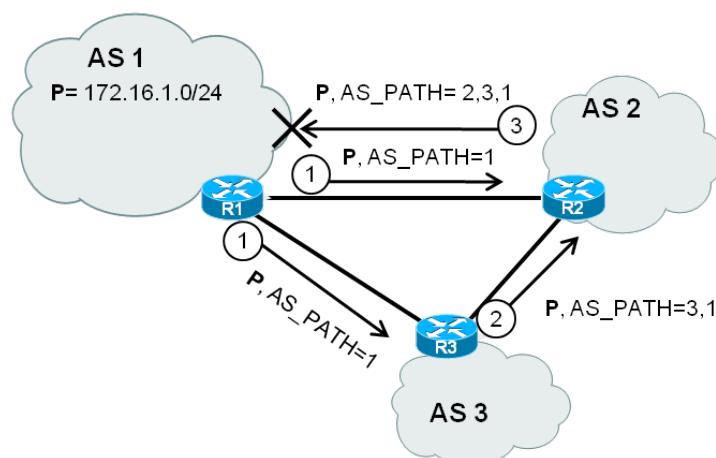
L'atribut *Origin* (codi=1) indica com un *router* BGP ha après una ruta. Pot prendre 3 valors:

- **IGP**. Vol dir que el prefix s'ha après a partir d'un protocol intern del sistema autònom que ha originat el prefix. A les rutes BGP se'ls assigna un *origin IGP* si s'han après a partir de la taula d'encaminament d'un IGP a partir de la comanda de configuració BGP "network".
- **EGP**. La ruta s'ha après a través de l'*External Border Gateway Protocol* (EBGP).
- **Incomplete**. La ruta s'ha après per algun altre mitjà. Una ruta té origen "incomplet" si s'ha redistribuït al BGP des d'un altre protocol.

2.4.2 AS_PATH

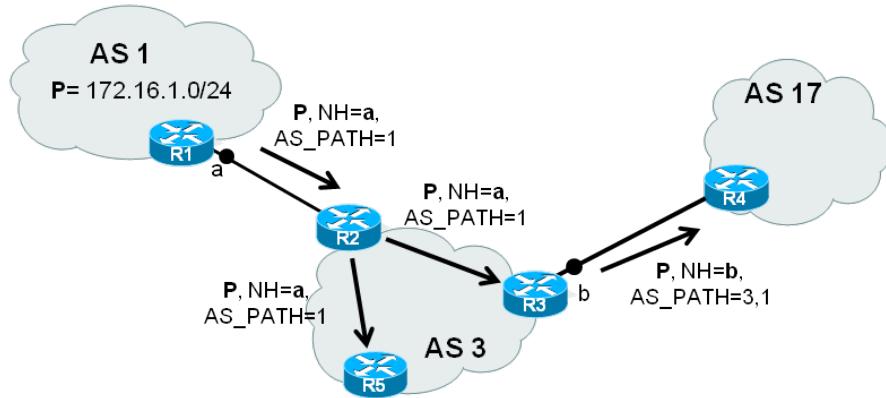
L'atribut *AS_PATH* (codi=2) conté la seqüència d'identificadors de sistema autònom (ASNs) que descriuen el camí cap a un determinat prefix. Quan un *router* origina un prefix i l'adverteix a un *router* d'un sistema autònom veí, afegeix el seu ASN a l'*AS_PATH*. A mesura que altres *routers* BGP anuncien el prefix cap a *peers* externs, afegeixen el seu ASN a l'*AS_PATH*. El resultat és que, en un determinat *router*, l'atribut *AS_PATH* conté la llista dels sistemes autònoms (els seus ASNs) que cal travessar per arribar al prefix associat. Aquest atribut té dues finalitats. Primer, serveix com a mesura de mètrica ja que indica el cost en nombre de sistemes autònoms que cal travessar per arribar a un cert prefix. Segon, és el mecanisme que incorpora el BGP per evitar la creació de bucles d'encaminament: si un *router* BGP rep l'anunci d'un cert prefix via EBGP i l'*AS_PATH* del prefix conté el ASN al qual pertany el *router*, aquest descartarà l'anunci d'aquest prefix.

La figura següent mostra un escenari on un prefix **P** travessa tres sistemes autònoms. L'AS 1 és l'origen de la ruta pel prefix 172.16.1.0/24 i l'adverteix als routers R2 i R3 dels AS veïns, amb l'atribut *AS_PATH* igual a {1}. El router R3 advertirà el prefix al router R2 amb un *AS_PATH*={3,1}. Per tant, el router R2, tindrà dos possibles rutes per arribar a P: una passant per AS3 (amb *AS_PATH*=3,1) i una altra passant directament per l'AS1 (amb *AS_PATH*=1). En absència d'altres atributs, el router R2 triaria la ruta amb l'*AS_PATH* més curt i, per tant, per arribar a la xarxa 172.16.1.0/24 encaminaria el trànsit cap a R1. És a dir, triaria aquesta ruta com a "millor" i aquesta seria la ruta que utilitzaria i advertiria a altres peers BGP. Supposeu, però, que degut a altres atributs, R2 tria com a millor ruta la que passa per AS3. En aquest cas, advertirà la seva millor ruta cap al sistema autònom AS1, amb *AS_PATH*=2,3,1. Quan R1 rebi aquest *update* el descartarà ja que l'*AS_PATH* associat a aquesta ruta conté el seu propi número d'AS.



2.4.3 NEXT_HOP

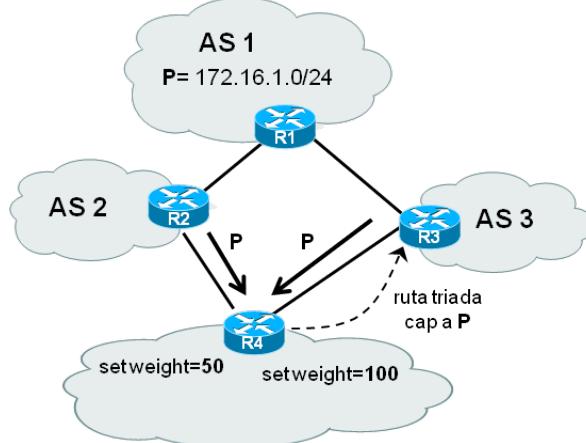
L'atribut NEXT_HOP (codi=3) de l'EBGP indica l'adreça IP del router que anuncia una ruta als sistemes autònoms externs. Als peers EBGP, l'adreça del next-hop és l'adreça IP de la connexió entre els peers. L'IBGP propaga l'adreça next-hop de l'EBGP dins l'AS local. A la figura següent, el router R1 adverteix la xarxa 172.16.1.0/24 amb next hop la seva adreça “a”. Quan el router R2 propaga aquesta ruta via iBGP als altres routers del seu sistema autònom, manté el next hop après via EBGP. Si el router R3 no té una ruta cap a l'adreça del next-hop (a), no considerarà la ruta apresa pel prefix com a vàlida. Aquí és on entra en joc l'encaminament interior al sistema autònom. Els routers R3 i R5 han de tenir saber com arribar a “a”. Existeixen dues solucions per solventar aquest problema. O bé fer que l'IGP del sistema autònom AS3 distribueixi una ruta per la xarxa que interconnecta els routers R1 i R2, o bé habilitar la opció de configuració *next-hop-self* al router R2. Amb aquesta opció, quan el router R2 propaga el prefix als seus peers interns, sobreescriví l'atribut next-hop amb la seva adreça IP; de manera que els seus peers interns tenen una ruta cap al next-hop. Finalment, fixeu-vos com el router R3 adverteix el prefix P al peer extern R4 amb un next-hop de “b” i com afegeix el seu número de sistema autònom (3) a l'AS_PATH.



2.4.4 WEIGHT

El WEIGHT no és realment un atribut BGP. És un paràmetre definit per Cisco i és local, és a dir, no es transmet als routers veïns. Aquest atribut permet assignar més pes (preferència) a les rutes apreses per un determinat veí BGP.

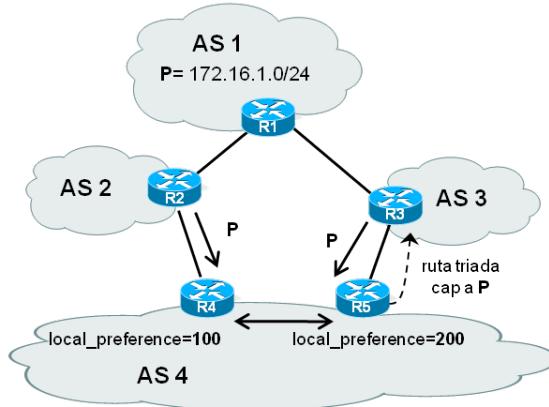
A la figura següent, el router R4 descobreix dues rutes per arribar a la xarxa 172.16.1.0/24, apreses dels seus peers eBGP R2 i R3. Quan rep l'update de R2, assigna un pes de 50 a la ruta cap al prefix P. En canvi, assigna un pes de 100 a la ruta apresa de R3. R4 guarda les dues rutes a la seva taula BGP (amb els seus atributs), però tria com a millor ruta aquella de major pes. Aquesta ruta és la que instal·la a la seva taula d'encaminament i la que adverteix a altres peers.



2.4.5 LOCAL_PREFERENCE

L'atribut LOCAL_PREFERENCE (codi=5) permet triar un router de sortida per al trànsit destinat a un cert prefix. És a dir, és un atribut que afecta la tria de rutes per al trànsit sortint (*outbound*) d'un sistema autònom. A diferència de l'atribut *weight*, la preferència local es propaga a través del sistema autònom local: és a dir, és un atribut que acompaña als prefixos intercanviats via iBGP entre routers del mateix AS.

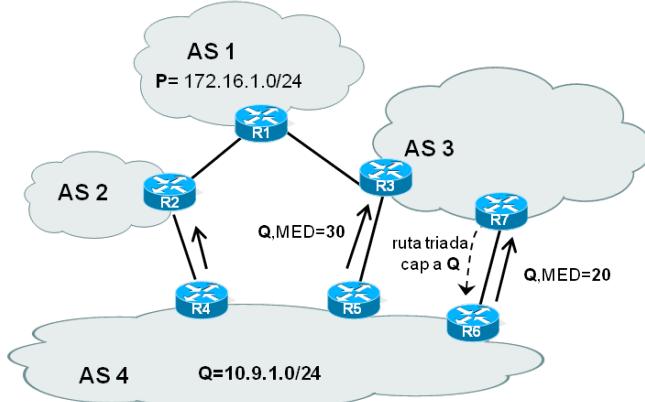
Per veure la utilitat de l'atribut LOCAL_PREFERENCE, suposeu que el sistema autònom AS4 de la següent figura té dos proveïdors que li proporcionen connectivitat a la resta d'Internet; els sistemes autònoms AS2 i AS3. Suposeu que, per algun motiu, al sistema autònom AS4 li interessa que tot el trànsit destinat a un cert prefix $P=172.16.1.0/24$ (de l'AS1) s'encamini pel sistema autònom AS3. A l'escenari, els routers R4 i R5 aprenen una ruta per arribar al prefix P cadascú. Els dos routers es comuniquen via iBGP. Per forçar que tot el trànsit destinat al prefix P surti pel router R5, els administradors de l'AS4 poden configurar R4 i R5 de manera que R5 assigni al prefix un LOCAL_PREFERENCE major que no pas R4. Com que els routers s'intercanvien les rutes (i atributs) via iBGP, els dos routers tindran les dues rutes però ambdós triaran com a millor aquella que surt per R5. Si, per alguna anomalia, R5 deixa de tenir una ruta cap a P (p.ex. perquè l'enllaç entre R1 i R3 cau o perquè R3 deixa de funcionar), el trànsit destinat al prefix originat a l'AS4 serà encaminat per R2.



2.4.6 MULTI_EXIT_DISCRIMINATOR

El Multi-exit Discriminator (o MED o “mètrica”, amb codi=4) s'utilitza per suggerir a un AS extern el punt d'entrada del sistema autònom que adverteix un prefix. És a dir, influeix en l'elecció de la millor ruta per al trànsit entrant (*inbound*). El terme suggerir s'utilitza perquè l'AS extern que rep el MED pot utilitzar altres atributs BGP per escollir les rutes de manera que no necessàriament la ruta amb menor MED sigui triada.

A la figura següent, els routers del sistema autònom AS 4 adverteixen un prefix $Q=10.9.1.0/24$ als sistemes autònoms AS2 i AS3. Suposeu que el router R5 és un model antic i que l'enllaç amb R3 és de baixa capacitat, de manera que l'AS4 només voldria usar aquest enllaç com a “backup”. Per suggerir als routers de l'AS3 que el punt preferent d'entrada al AS4 pel prefix Q és R6, els administradors de l'AS4 configuren R5 per a que adverteixi el prefix amb un MED (mètrica) major que no pas R6. Així, en absència d'altres atributs amb més pes en el criteri de decisió de rutes, els routers de l'AS3 encaminaran el trànsit cap a Q passant per R6.



2.5 DIFERÈNCIES ENTRE IBGP I EBGP

El protocol BGP té dues variants segons si les sessions BGP s'estableixen entre routers del mateix sistema autònom o no. Tot i que el protocol és essencialment el mateix, el funcionament del mateix difereix en certs aspectes que cal entendre bé. Els routers saben si estan dialogant iBGP ó eBGP ja que coneixen el ASN de l'altre peer tot just s'inicia la sessió BGP entre ells: el missatge OPEN inclou l'ASN a què pertany cada router. La següent taula resumeix les diferències més importants pel que fa al comportament dels routers en les sessions iBGP i eBGP.

	eBGP	iBGP
Connexió entre peers	Dos peers eBGP (és a dir, de sistemes autònoms diferents) han d'estar directament connectats.	Els peers iBGP poden estar directament connectats o no. Per tant, és necessari que ambdós tinguin rutes per arribar l'un a l'altre (p.ex. les rutes que els proporcioni un IGP com el RIP o l'OSPF).
Propagació de rutes	Un router adverteix una ruta apresa via eBGP tant a peers eBGP com iBGP.	Un router adverteix via iBGP als peers interns només aquelles rutes que ha après via eBGP, però no aquelles que hagi après d'un altre peer via iBGP. Aquest és el motiu pel qual tots els routers BGP d'un mateix AS han d'estar lògicament connectats via iBGP.
AS_PATH	Quan un router adverteix un prefix a un router extern (eBGP) afegeix el ASN del sistema autònom al que pertany a l'AS_PATH.	Quan un router adverteix un prefix via iBGP a un peer interior (del mateix AS), no afegeix el ASN local a l'AS_PATH. Aquest és el motiu pel qual un router no adverteix en sessions iBGP els prefixes que aprèn d'un altre router via iBGP: es podrien crear bucles ja que l'AS_PATH no s'ha actualitzat.
Local Preference	No s'exporta l'atribut LOCAL_PREFERENCE en sessions eBGP.	Obligatori
Next-Hop	Quan un router adverteix un prefix via eBGP a un router d'un AS veí, fixa el valor del NEXT_HOP amb l'adreça IP de la interfície amb què es comunica amb el veí.	Quan un router adverteix un prefix a un router del mateix AS (iBGP), no modifica el valor del NEXT_HOP; a excepció de si fa servir la opció de <i>next-hop-self</i> .

2.6 SELECCIÓ DE RUTES BGP

Un router BGP pot aprendre més d'un camí cap a un determinat prefix de xarxa, però només en triarà un com a "millor". Aquest serà el que instal·larà a la seva taula d'encaminament i el que advertirà als seus peers BGP, interns o externs. En absència de configuració addicional, un router BGP tria, de tots els possibles camins que aprèn per a arribar a un determinat prefix, aquell amb un AS_PATH més curt. Tot i això, la característica més rellevant del protocol BGP és la possibilitat d'especificar preferències per a triar camins en base a altres criteris a banda de la longitud de l'AS_PATH, per tal de permetre que els sistemes autònoms puguin desplegar polítiques d'encaminament d'acord, per exemple, a interessos econòmics o per tal d'optimitzar els seus recursos de xarxa; el que es coneix com enginyeria de trànsit. Aquestes preferències s'articulen, en la majoria de casos, mitjançant la modificació dels anomenats atributs, que poden entendre's com "propietats" associades als prefixos de xarxa.

Un router BGP fa servir un algorisme de decisió per tal de validar una ruta o bé per a triar-ne una com a "millor", en cas que existeixin vàries rutes cap al mateix prefix. L'algorisme de decisió no està del tot estandarditzat i, segons el fabricant, pot variar una mica. En el cas del zebra, l'algorisme de validació i tria de rutes és molt semblant al que implementen la majoria de fabricants de routers (per exemple Cisco) i és el següent:

1. No s'admet una ruta si el router no té cap camí cap al NEXT_HOP del prefix.
2. S'escull la ruta amb **major** pes (WEIGHT). (Criteri local al router).
3. S'escull la ruta amb **major** LOCAL_PREFERENCE. (Criteri global dins un sistema autònom).
4. Es prefereixen les rutes originades de forma local pel router.
5. S'escull la ruta que travessa un menor número d'AS (AS_PATH **més curt**)
6. S'escull la ruta amb un codi ORIGIN **menor**. L'atribut ORIGIN pot prendre 3 valors:
 - IGP = vol dir que el prefix s'ha generat amb la comanda *network* de BGP.
 - EGP= vol dir que el prefix s'ha generat pel EGP (el protocol precursor del BGP)
 - Incomplete=vol dir que el prefix s'ha redistribuït des d'un altre protocol.
 Es considera IGP < EGP < Incomplete.
7. S'escull la ruta amb **menor** MED (Multi-Exit Discriminator):
 - Si s'activa la opció *bgp deterministic-med*, s'ordenen els camins abans de comparar-los.
 - Si s'activa la opció *bgp always-compare-med*, es comparen tots els camins.
 - En qualsevol altre cas, només es té en compte l'atribut MED si els camins que es comparen són per al mateix AS destí.
8. Es prefereix una ruta apresa via eBGP sobre una apresa via iBGP.
9. Es tria la ruta amb menor mètrica (segons el protocol d'encaminament interior) cap al NEXT_HOP.
10. Quan les rutes que es comparen s'han rebut via eBGP, es queda la ruta que ja està seleccionada (la que ha arribat primer). Aquesta condició no aplica si s'activa l'opció *bgp bestpath compare-routerid* (que per defecte està desactivada).
11. Es tria la ruta apresa del veí amb menor identificador BGP (que sol ser una de les IPs del router).

La millor ruta (best), que s'escull després d'aplicar els criteris de la llista anterior, és la que s'adverteix, si escau, via BGP i és la que passa al procés global de selecció de rutes per ser instal·lada a la FIB, d'acord amb la seva distància administrativa. La taula següent mostra la distància administrativa per defecte dels diferents protocols d'encaminament:

Protocol	Distància
Directament connectat	0
Estàtic	1
eBGP	20
OSPF	110
ISIS	115
RIP	120
iBGP	200
Desconegut	255

Fixeu-vos que la distància administrativa d'una ruta apresa via eBGP és menor que la d'una ruta apresa via iBGP. Finalment, cal dir que existeix una versió estesa del protocol BGP que permet mantenir més d'una ruta activa cap a un determinat prefix. Aquesta extensió del protocol no la veurem a l'assignatura.

ACTIVITATS AL LABORATORI

Objectius de la pràctica

- Entendre el funcionament del protocol BGP.
- Aprendre a configurar els atributs de BGP.

La pràctica està pensada per treballar utilitzant un únic PC, que a partir d'ara s'anomenarà **PC**.

Les figures dels escenaris de les dues parts de la pràctica estan al final de tot d'aquest enunciat.

En aquesta pràctica, s'utilitzarà el software *quagga* per configurar les adreces IP dels *routers* (dimoni *zebra*), el protocol OSPF (dimoni *ospfd*) per a l'encaminament unicast interior i el protocol BGP (dimoni *bgpd*) per a l'encaminament exterior.

La figura següent resumeix les comandes del *vtysh* que s'han vist fins ara en pràctiques anteriors i, a més a més, inclou les comandes de configuració del protocol BGP i les comandes per configurar llistes d'accés i route-maps.

```
#-----#
   | configure terminal                                     (config)
   +-- interface IFNAME                                    (config-if)
   |   +-- ip address A.B.C.D/M
   |   +-- [no] ip rip split-horizon [poisoned-reverse]
   |   +-- ip ospf cost <1-65535>
   |   +-- ip ospf network point-to-point
   |   +-- ip ospf priority <0-255>
   |   +-- exit
   +-- router rip                                         (config-router)
   |   +-- network IFNAME
   |   +-- passive-interface IFNAME
   |   +-- exit
   +-- router ospf                                         (config-router)
   |   +-- router-id A.B.C.D
   |   +-- network A.B.C.D/M area E.F.G.H
   |   +-- area A.B.C.D range E.F.G.H/M
   |   +-- passive-interface IFNAME
   |   +-- exit
   +-- router bgp <1-4294967295>                      (config-router)
   |   +-- bgp router-id A.B.C.D
   |   +-- network A.B.C.D/M
   |   +-- neighbor A.B.C.D remote-as <1-4294967295>
   |   +-- neighbor A.B.C.D route-map WORD in/out
   |   +-- neighbor A.B.C.D soft-reconfiguration in
   |   +-- bgp deterministic-med
   |   +-- bgp always-compare-med
   |   +-- exit
   |   +-- route-map WORD permit <1-65535>                (config-route-map)
   |       +-- match ip address WORD
   |       +-- set local-preference <0-4294967295>
   |       +-- set metric <0-4294967295>
   |       +-- set weight <0-4294967295>
   |       +-- exit
   |   +-- ip route A.B.C.D/M E.F.G.H
   |   +-- access-list WORD permit A.B.C.D/M
   |   +-- clear ip bgp A.B.C.D
   |   +-- clear ip bgp A.B.C.D soft in
   |   +-- clear ip bgp A.B.C.D in
   |   +-- clear ip bgp A.B.C.D out
   |   +-- exit
   +-- show running-config
   +-- show ip route
   +-- show ip rip
   +-- show ip ospf database
   +-- show ip ospf database router
   +-- show ip ospf database network
   +-- show ip ospf database summary
   +-- show ip bgp
   +-- show ip bgp summary
   +-- show ip bgp neighbors
   +-- show ip bgp neighbors A.B.C.D
   +-- show ip bgp neighbors A.B.C.D received-routes
   +-- ping
   +-- write memory
   +-- exit
```

En relació a la configuració dels protocols RIP i OSPF, a la figura anterior apareix una comanda que no es va veure en les pràctiques d'aquests protocols i és: `passive-interface IFNAME`, on `IFNAME` és el nom de la interfície que es vol que sigui passiva. Aquesta comanda permet que el *router* anunci la xarxa connectada a una determinada interfície, anomenada `IFNAME`, via RIP o OSPF, segons correspongui, però no envia paquets d'aquest protocol per aquella interfície.

Per configurar les comandes del protocol BGP cal entrar al submenú de configuració del vtysh (configure terminal) i després entrar al submenú de configuració del router bgp amb la comanda `router bgp ASN on ASN` és el número del sistema autònom que es vol configurar.

Dins aquest submenú, les comandes bàsiques que cal conèixer són:

- `bgp router-id A.B.C.D`. Permet indicar l'indicador del router BGP. Igual que en el cas del protocol OSPF, l'indicador és un número de 32 bits que s'indica en format dotted quad.
- `network A.B.C.D/M`. Permet indicar un prefix del sistema autònom del router que es vol anunciar via BGP. Es poden configurar diversos prefixes per cada router.
- `neighbor A.B.C.D remote-as ASN_veí`. Permet indicar l'adreça IP d'un router amb qui s'ha d'estalvir una sessió BGP, així com el número del sistema autònom del veí. Si el veí està al mateix AS que el router que s'està configurant, la sessió serà iBGP. Si el veí està en un AS diferent del router que s'està configurant, la sessió serà eBGP.

Per comprovar l'estat de les sessions BGP hi ha diverses comandes que es poden executar des de l'arrel del vtysh:

- `show ip bgp summary`. Mostra una llista dels veïns BGP que té configurats els routers i amb els que el router intenta establir una sessió BGP. Si la darrera columna és un número, significa que la sessió està establerta (Established), en cas contrari, apareix una paraula que indica l'estat en el que està la sessió (Idle, Active, etc.)
- `show ip bgp neighbors`. Mostra el detall de les sessions del router amb cada veí.
- `show ip bgp neighbors A.B.C.D`. Mostra el detall de la sessió BGP amb el veí A.B.C.D.
- `show ip bgp neighbors A.B.C.D received-routes`. Mostra el detall de les rutes rebudes pel veí amb adreça IP A.B.C.D. (Només disponible si s'activa la característica de "configuració suau" amb la comanda `neighbor A.B.C.D soft-reconfiguration in` del submenú de configuració del router).

Des de l'arrel del vtysh es pot executar la comanda: `show ip bgp` que mostra la llista de rutes apreses via BGP (amb els corresponents atributs associats).

Per modificar els atributs associats a un determinat prefix (o prefixes), en aquesta pràctica es configuraran llistes d'accés (access-list) i filtres (route-map) des del submenú de configuració (configure terminal).

A continuació es mostra un exemple de configuració d'un route-map que permet modificar l'atribut local-preference i fixar-lo a un valor de 200 per al prefix 10.0.0.0/16 (o prefixes inclosos en aquest rang d'adreses) que està definit a l'access-list ACL_TEST. La segona condició del route-map RM_TEST, que té prioritat 20, és necessària per a que la resta de prefixes diferents del 10.0.0.0/16 que siguin evaluats pel route-map es transmetin (si no es posés, els prefixes es descartarien i no apareixerien a la taula BGP). Aquest route-map s'aplica a tots els paquets rebuts del veí BGP que té l'adreça IP 20.1.1.1:

```
access-list ACL_TEST permit 10.0.0.0/16
route-map RM_TEST 10
    match ip address ACL_TEST
    set local-preference 200
route-map RM_TEST 20
router bgp 102
    neighbor 20.1.1.1 route-map RM_TEST in
```

A continuació es mostra un exemple de configuració d'un route-map que permet modificar l'atribut metric (és a dir, l'atribut MED) i fixar-lo a un valor de 5 per al prefix 10.0.0.0/16 (o prefixes inclosos en aquest rang d'adreses) que

està definit a l'access-list `ACL_TEST`. La segona condició del route-map `RM_TEST`, que té prioritat 20, és necessària per a que la resta de prefixes diferents del 10.0.0.0/16 que siguin evaluats pel route-map es transmetin (si no es posés, els prefixes es descartarien i no apareixerien a la taula BGP). Aquest route-map s'aplica a tots els paquets enviats al veí BGP que té l'adreça IP 20.1.1.2:

```
access-list ACL_TEST permit 10.0.0.0/16
route-map RM_TEST 10
    match ip address ACL_TEST
    set metric 5
route-map RM_TEST 20
router bgp 202
neighbor 20.1.1.2 route-map RM_TEST out
```

En el cas de modificar els atributs associats a un prefix, cal aplicar els canvis. Per fer-ho, hi ha tres maneres:

- `clear ip bgp A.B.C.D`. Permet fer un “hard reset” amb el veí A.B.C.D. Si es posa un * es fa un hard reset amb tots els veïns.
- `clear ip bgp A.B.C.D in`. Si s'utilitza el mecanisme de “soft reset” aquesta comanda envia un missatge ROUTE-REFRESH al veí A.B.C.D per a que aquest envii les rutes on poder aplicar la nova política.
- `clear ip bgp A.B.C.D soft in`. Aquesta és la comanda que cal fer servir per aplicar els canvis sobre les rutes rebudes del veí A.B.C.D si s'utilitza la característica de “configuració suau” (que recordeu que és excloent amb el mecanisme de soft reset).

Si el que es vol és aplicar una política determinada sobre les rutes que s'envien a un determinat veí, aleshores cal posar la comanda: `clear ip bgp A.B.C.D out`

Hi ha dues comandes que influencien en la selecció de ruta segons l'atribut MED i que es poden configurar des del menú de configuració del router bgp:

- `bgp deterministic-med`. Quan s'activa assegura que es compara l'atribut MED quan s'escullen rutes advertides per diferents peers del mateix AS.
- `bgp always-compare-med`. Quan s'activa assegura que es compara l'atribut MED quan s'escullen rutes advertides des de diferents AS.

Per entendre com s'utilitzen aquestes comandes, suposeu l'exemple següent, en el qual un router té tres opcions per arribar al prefix 40.1.0.0/16:

Network	Next Hop	Metric	Path
40.1.0.0/16	10.0.0.10	7	105 104 ← ruta 1
40.1.0.0/16	10.0.0.1	10	102 104 ← ruta 2
i 40.1.0.0/16	10.0.0.14	5	105 104 ← ruta 3

Quan un router BGP rep múltiples rutes cap a un determinat destí, les ordena en ordre invers de com les ha rebut (de la més nova a la més vella). Aleshores, les compara dos a dos, començant per la més nova i anant cap a la més vella. La millor de cada parella es compara amb la següent fins que s'acaba la llista. Suposeu que la llista de l'exemple ja està ordenada de la més nova a la més vella.

Cas 1. `bgp deterministic-med` no habilitat i `bgp always-compare-med` no habilitat.

Quan es comparen les dues primeres rutes, s'escull la 2 perquè les dues són apreses via eBGP i la ruta 2 s'ha après abans. Quan es compara la ruta 2 amb la ruta 3, guanya la ruta 2 perquè és externa. **Es tria la del next hop 10.0.0.1 (ruta 2).**

Cas 2. `bgp deterministic-med` no habilitat i `bgp always-compare-med` habilitat.

Quan es comparen les dues primeres rutes s'escull la ruta 1 perquè el MED és menor (les rutes són de diferents ASs però es comparen perquè està habilitada l'opció `bgp always-compare-med`). Quan es compara la ruta 1 amb la ruta 3 guanya la ruta 3 perquè té menor MED. **Es tria la del next hop 10.0.0.14 (ruta 3).**

Cas 3. bgp deterministic-med habilitat i bgp always-compare-med no habilitat.

Quan s'habilita l'opció bgp deterministic-med, les rutes que provenen del mateix AS s'ajunten i es comparen entre sí. Aleshores, les millors de cada grup es comparen entre sí. És a dir, en aquest exemple primer es compara la ruta 1 amb la ruta 3 i guanya la ruta 3 perquè té menor MED. Després es compara la ruta 3 amb la ruta 2 (sense tenir en compte el MED perquè no està habilitada l'opció always-compare-med) i guanya la ruta 2 perquè és externa. **Es tria la del next hop 10.0.0.1 (ruta 2).**

Cas 4. bgp deterministic-med habilitat i bgp always-compare-med habilitat.

Com en el cas 3, primer es compara la ruta 1 amb la ruta 3 i guanya la ruta 3 perquè té menor MED. Després es compara la ruta 3 amb la ruta 2 (tenint en compte el MED perquè està habilitada l'opció always-compare-med) i guanya la ruta 3 perquè té menor MED. **Es tria la del next hop 10.0.0.14 (ruta 3).**

A la pràctica només es preguntarà el cas 3 (bgp deterministic-med habilitat i bgp always-compare-med no habilitat)

PART I. FUNCIONAMENT BÀSIC DEL PROTOCOL BGP

Descarregueu el fitxer P05.zip que conté els fitxers de la pràctica 5 de l'Atenea i **guardieu-lo a l'escriptori** (no canvieu el nom del fitxer).

Obriu un terminal del **PC** i executeu la comanda: `unzip-files P05`

(Si a l'executar l'unzip-files us pregunta si voleu substituir (replace) algun fitxer, contesteu: `A + Enter`)

Escenari P05-E01

A continuació arrancareu els scripts per configurar la topologia de l'escenari i les adreces IP de les interfícies dels routers i PCs tal i com teniu representat a la figura **Escenari part I** del final de la pràctica. Com s'observa a la figura, i fins que no s'indiqui el contrari, la interfície eth2 de R05 i la interfície eth0 de R06 han d'estar desconnectades dels bridges.

En un terminal del **PC**, executeu la comanda: `P05-E01-start-zebra`

Triga una mica. Espereu a que acabi d'executar-se l'script i surti el prompt del terminal.

Exercici 1. Configuració de l'encaminament interior dels sistemes autònoms

A continuació, arrancareu els dimonis del protocol d'encaminament interior que s'utilitzarà en els diferents sistemes autònoms amb els corresponents scripts de configuració.

En un terminal del **PC**, executeu la comanda: `P05-E01-start-int`

Dins de cada sistema autònom (AS a partir d'ara) s'utilitza un protocol d'encaminament interior (per exemple RIP o OSPF) i/o rutes estàtiques per permetre que tots els routers de l'AS sàpiguen arribar a totes les xarxes d'aquest AS.

A més a més, és important tenir en compte que, en sessions iBGP (és a dir, sessions BGP entre routers del mateix AS), quan un router adverteix un prefix, per defecte no modifica el next-hop (consulteu l'explicació de la pàgina 14). Per aquest motiu és important que els routers BGP d'un AS sàpiguen arribar a les xarxes que connecten aquest AS amb els AS veïns. En aquest exercici serà el protocol d'encaminament interior de cada AS qui s'encarregarà de permetre que s'anunciïn aquestes xarxes dins l'AS però prenen les mesures necessàries per no enviar missatges del protocol d'encaminament interior als routers que pertanyen a un AS diferent.

Al sistema autònom 102 s'utilitza OSPF com a protocol d'encaminament interior.

Captureu paquets a les interfícies del *router* R02 externes a l'AS 102.

Nota: no s'adjunta cap captura per fer aquest exercici perquè la captura estaria buida. No s'envien paquets OSPF als routers externs a l'AS.

Mireu les rutes de la taula d'encaminament de R01:

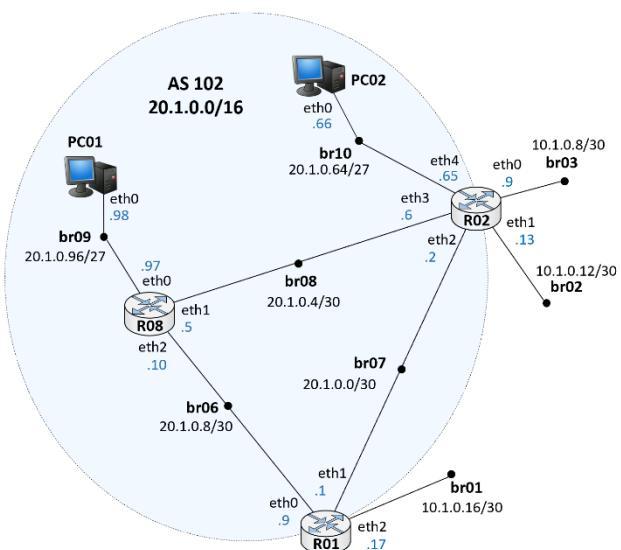
```
lxc-attach -n R01 -- vtysh -c 'show ip route'

root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip route'
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route
O>* 10.1.0.8/30 [110/20] via 20.1.0.2, eth1, 00:00:16
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth1, 00:00:16
O  10.1.0.16/30 [110/10] is directly connected, eth2, 00:01:01
C>* 10.1.0.16/30 is directly connected, eth2
O  20.1.0.0/30 [110/10] is directly connected, eth1, 00:00:21
C>* 20.1.0.0/30 is directly connected, eth1
O>* 20.1.0.4/30 [110/20] via 20.1.0.10, eth0, 00:00:06
      *               via 20.1.0.2, eth1, 00:00:06
O  20.1.0.8/30 [110/10] is directly connected, eth0, 00:01:01
C>* 20.1.0.8/30 is directly connected, eth0
O>* 20.1.0.64/27 [110/20] via 20.1.0.2, eth1, 00:00:16
O>* 20.1.0.96/27 [110/20] via 20.1.0.10, eth0, 00:00:16
C>* 127.0.0.0/8 is directly connected, lo
```

Nota: les rutes de les xarxes **directament connectades** que apareixen a la taula com a rutes OSPF sense >* (per exemple: O 10.1.0.16/30 [110/10] is directly connected, eth1, 00:01:01) no caldrà que les escrivíssiu a la taula si en un examen es demana la sortida de la comanda 'show ip route'. Tampoc us caldrà escriure la ruta C>* 127.0.0.0/8 is directly connected, lo

a. Quines rutes surten? Raoneu la resposta.

A la taula d'encaminament de R01 es veuen les rutes següents:



1. Les xarxes directament connectades al router


```
C>* 10.1.0.16/30 is directly connected, eth2
C>* 20.1.0.0/30 is directly connected, eth1
C>* 20.1.0.8/30 is directly connected, eth0
```
2. Les xarxes internes a l'AS que el router ha après via OSPF


```
O>* 20.1.0.4/30 [110/20] via 20.1.0.10, eth0
      *               via 20.1.0.2, eth1
O>* 20.1.0.64/27 [110/20] via 20.1.0.2, eth1
O>* 20.1.0.96/27 [110/20] via 20.1.0.10, eth0
```
3. Les xarxes que connecten l'AS 102 amb els AS externs que el router ha après via OSPF:


```
O>* 10.1.0.8/30 [110/20] via 20.1.0.2, eth1
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth1
```

Mireu la configuració de R02 amb la comanda: `lxc-attach -n R02 -- vtysh -c 'show running-config'`

```
root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show run'
Building configuration...

Current configuration:
!
interface eth0
  ip address 10.1.0.9/30
!
interface eth1
  ip address 10.1.0.13/30
!
interface eth2
  ip address 20.1.0.2/30
!
interface eth3
  ip address 20.1.0.6/30
!
interface eth4
  ip address 20.1.0.65/27
!
router ospf
  passive-interface eth0
  passive-interface eth1
  network 10.1.0.8/30 area 0.0.0.0
  network 10.1.0.12/30 area 0.0.0.0
  network 20.1.0.0/30 area 0.0.0.0
  network 20.1.0.4/30 area 0.0.0.0
  network 20.1.0.64/27 area 0.0.0.0
!
ip forwarding
ipv6 forwarding
!
line vty
!
end
```

- b. Quina comanda de la configuració de R02 permet que R01 aprengui a arribar a les xarxes directament connectades a R02 que són externes al sistema autònom?

La comanda que ho permet és `network A.B.C.D/E area X.Y.Z.V` especificant que R02 ha d'anunciar les xarxes externes a l'AS102 com a xarxes de l'àrea 0.0.0.0:

```
router ospf
  network 10.1.0.8/30 area 0.0.0.0
  network 10.1.0.12/30 area 0.0.0.0
```

Observeu les captures del Wireshark i verifiqueu que R02 no envia missatges OSPF per les interfícies externes al seu sistema autònom.

No s'envien paquets OSPF per les interfícies externes a l'AS.

c. Quina comanda evita que R02 envii paquets OSPF a routers de fora del seu sistema autònom?

La comanda que ho evita és `passive-interface IFNAME` que permet especificar per quines interfícies R02 no ha d'enviar paquets OSPF:

```
router ospf
  passive-interface eth0
  passive-interface eth1
```

Verifiqueu que els tres routers de l'AS 102 tenen rutes per arribar a les mateixes xarxes (i que aquestes són subxarxes del rang 20.1.0.0/16 o bé són xarxes directament connectades als routers frontera de l'AS 102).

Els tres routers tenen a la taula d'encaminament les rutes comentades a l'apartat a) de l'exercici 1. A continuació s'adjunten les taules d'encaminament dels routers R01, R02 i R08 de l'AS 102 per comprovar-ho.

Nota: a partir d'ara, en la sortida de la comanda 'show ip route' ja no s'adjunta la llegenda de codis:

Codes: K - kernel route, C - connected, S - static, R - RIP,
 (...)

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip route'
O>* 10.1.0.8/30  [110/20] via 20.1.0.2, eth1, 00:01:28
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth1, 00:01:28
C>* 10.1.0.16/30 is directly connected, eth2
C>* 20.1.0.0/30 is directly connected, eth1
O>* 20.1.0.4/30  [110/20] via 20.1.0.10, eth0, 00:01:18
  *                  via 20.1.0.2, eth1, 00:01:18
C>* 20.1.0.8/30 is directly connected, eth0
O>* 20.1.0.64/27 [110/20] via 20.1.0.2, eth1, 00:01:28
O>* 20.1.0.96/27 [110/20] via 20.1.0.10, eth0, 00:01:28

root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip route'
C>* 10.1.0.8/30 is directly connected, eth0
C>* 10.1.0.12/30 is directly connected, eth1
O>* 10.1.0.16/30 [110/20] via 20.1.0.1, eth2, 00:01:31
C>* 20.1.0.0/30 is directly connected, eth2
C>* 20.1.0.4/30 is directly connected, eth3
O>* 20.1.0.8/30  [110/20] via 20.1.0.1, eth2, 00:01:21
  *                  via 20.1.0.5, eth3, 00:01:21
C>* 20.1.0.64/27 is directly connected, eth4
O>* 20.1.0.96/27 [110/20] via 20.1.0.5, eth3, 00:01:21

root@api-mv:~# lxc-attach -n R08 -- vtysh -c 'show ip route'
O>* 10.1.0.8/30  [110/20] via 20.1.0.6, eth1, 00:01:30
O>* 10.1.0.12/30 [110/20] via 20.1.0.6, eth1, 00:01:30
O>* 10.1.0.16/30 [110/20] via 20.1.0.9, eth2, 00:01:30
O>* 20.1.0.0/30  [110/20] via 20.1.0.6, eth1, 00:01:30
  *                  via 20.1.0.9, eth2, 00:01:30
C>* 20.1.0.4/30 is directly connected, eth1
C>* 20.1.0.8/30 is directly connected, eth2
O>* 20.1.0.64/27 [110/20] via 20.1.0.6, eth1, 00:01:30
C>* 20.1.0.96/27 is directly connected, eth0
```

Dins el sistema autònom 104 s'utilitza RIP com a protocol d'encaminament interior.

Atureu les captures del Wireshark i captureu els paquets a les interfícies de R03 externes al seu sistema autònom.

Nota: no s'adjunta cap captura per fer aquest exercici perquè la captura estaria buida. No s'envien paquets RIP als routers externs a l'AS.

Mireu les rutes de la taula d'encaminament de R04:

```
lxc-attach -n R04 -- vtysh -c 'show ip route'

root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip route'
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

C>* 10.1.0.0/30 is directly connected, eth1
R>* 10.1.0.4/30 [120/3] via 40.1.0.5, eth0, 00:02:58
R>* 10.1.0.12/30 [120/3] via 40.1.0.5, eth0, 00:02:58
R>* 10.1.0.16/30 [120/3] via 40.1.0.5, eth0, 00:02:58
R>* 40.1.0.0/30 [120/2] via 40.1.0.5, eth0, 00:02:59
C>* 40.1.0.4/30 is directly connected, eth0
R>* 40.1.0.64/27 [120/2] via 40.1.0.5, eth0, 00:02:59
C>* 40.1.0.128/27 is directly connected, eth2
C>* 127.0.0.0/8 is directly connected, lo
```

d. Quines rutes surten? Raoneu la resposta.

A la taula d'encaminament de R04 es veuen les rutes següents:

1. Les xarxes directament connectades al router

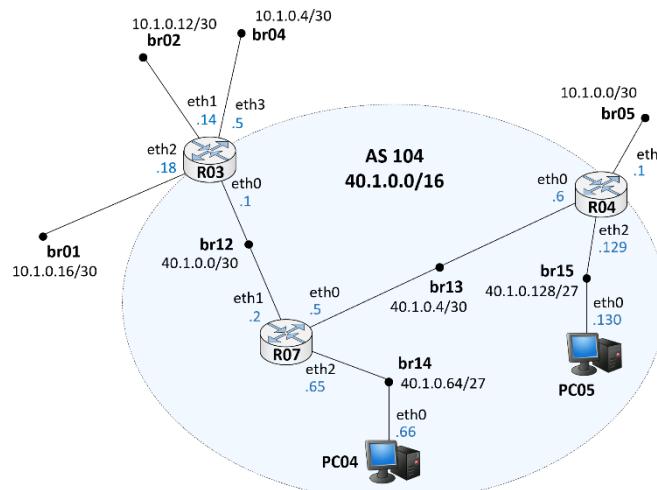
```
C>* 10.1.0.0/30 is directly connected, eth1
C>* 40.1.0.4/30 is directly connected, eth0
C>* 40.1.0.128/27 is directly connected, eth2
```

2. Les xarxes internes a l'AS que el router ha après via RIP

```
R>* 40.1.0.0/30 [120/2] via 40.1.0.5, eth0, 00:02:59
R>* 40.1.0.64/27 [120/2] via 40.1.0.5, eth0, 00:02:59
```

3. Les xarxes que connecten l'AS 104 amb els AS externs que el router ha après via RIP

```
R>* 10.1.0.4/30 [120/3] via 40.1.0.5, eth0, 00:02:58
R>* 10.1.0.12/30 [120/3] via 40.1.0.5, eth0, 00:02:58
R>* 10.1.0.16/30 [120/3] via 40.1.0.5, eth0, 00:02:58
```



Mireu la configuració de R03 amb la comanda: `lxc-attach -n R03 -- vtysh -c 'show running-config'`

```
root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show run'
Building configuration...

Current configuration:
!
!
interface eth0
    ip address 40.1.0.1/30
!
interface eth1
    ip address 10.1.0.14/30
!
interface eth2
    ip address 10.1.0.18/30
!
interface eth3
    ip address 10.1.0.5/30
!
interface lo
!
router rip
    network eth0
    network eth1
    network eth2
    network eth3
    passive-interface eth1
    passive-interface eth2
    passive-interface eth3
!
ip forwarding
ipv6 forwarding
!
line vty
!
end
```

- e. Quina comanda de la configuració de R03 permet que R04 aprengui a arribar a la xarxa directament connectada a R03 però externa al sistema autònom 104?

La comanda que ho permet és `network IFNAME` especificant que R03 vol anunciar via RIP les xarxes que el connecten amb AS veïns:

```
router rip
    network eth1
    network eth2
    network eth3
```

Mirant les captures del Wireshark verifiqueu que R03 no envia missatges RIP per les interfícies externes al seu sistema autònom.

No s'envien paquets RIP per les interfícies externes a l'AS.

f. Quina comanda evita que R03 envii paquets RIP a routers de fora del seu sistema autònom?

La comanda que ho evita és `passive-interface IFNAME` que permet especificar per quines interfícies R03 no ha d'enviar paquets RIP:

```
router rip
  passive-interface eth1
  passive-interface eth2
  passive-interface eth3
```

Verifiqueu que els tres *routers* de l'AS 104 tenen rutes per arribar a les mateixes xarxes (i que aquestes són subxarxes del rang 40.1.0.0/16 o bé són xarxes directament connectades als *routers* frontera de l'AS 104).

Els tres routers tenen a la taula d'encaminament les rutes comentades a l'apartat d) de l'exercici 1 A continuació s'adjunten les taules d'encaminament dels routers R03, R07 i R04 de l'AS 104 per comprovar-ho.

```
root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip route'
R>* 10.1.0.0/30 [120/3] via 40.1.0.2, eth0, 00:04:05
C>* 10.1.0.4/30 is directly connected, eth3
C>* 10.1.0.12/30 is directly connected, eth1
C>* 10.1.0.16/30 is directly connected, eth2
C>* 40.1.0.0/30 is directly connected, eth0
R>* 40.1.0.4/30 [120/2] via 40.1.0.2, eth0, 00:04:05
R>* 40.1.0.64/27 [120/2] via 40.1.0.2, eth0, 00:04:05
R>* 40.1.0.128/27 [120/3] via 40.1.0.2, eth0, 00:04:05
C>* 127.0.0.0/8 is directly connected, lo

root@api-mv:~# lxc-attach -n R07 -- vtysh -c 'show ip route'
R>* 10.1.0.0/30 [120/2] via 40.1.0.6, eth0, 00:04:11
R>* 10.1.0.4/30 [120/2] via 40.1.0.1, eth1, 00:04:10
R>* 10.1.0.12/30 [120/2] via 40.1.0.1, eth1, 00:04:10
R>* 10.1.0.16/30 [120/2] via 40.1.0.1, eth1, 00:04:10
C>* 40.1.0.0/30 is directly connected, eth1
C>* 40.1.0.4/30 is directly connected, eth0
C>* 40.1.0.64/27 is directly connected, eth2
R>* 40.1.0.128/27 [120/2] via 40.1.0.6, eth0, 00:04:11
C>* 127.0.0.0/8 is directly connected, lo

root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip route'
C>* 10.1.0.0/30 is directly connected, eth1
R>* 10.1.0.4/30 [120/3] via 40.1.0.5, eth0, 00:04:14
R>* 10.1.0.12/30 [120/3] via 40.1.0.5, eth0, 00:04:14
R>* 10.1.0.16/30 [120/3] via 40.1.0.5, eth0, 00:04:14
R>* 40.1.0.0/30 [120/2] via 40.1.0.5, eth0, 00:04:15
C>* 40.1.0.4/30 is directly connected, eth0
R>* 40.1.0.64/27 [120/2] via 40.1.0.5, eth0, 00:04:15
C>* 40.1.0.128/27 is directly connected, eth2
C>* 127.0.0.0/8 is directly connected, lo
```

Atureu les captures del Wireshark.

Als sistemes autònoms 103.i 105 d'aquest escenari no s'ha configurat cap router intern ni cap protocol d'encaminament interior.

Exercici 2. Configuració de l'encaminament exterior als sistemes autònoms

A continuació, configurareu el protocol BGP per tal que els *routers* BGP aprenguin a arribar als prefixes d'altres sistemes autònoms.

En un terminal del **PC**, executeu la comanda: P05-E01-start-bgpd

En aquest escenari tots els *routers* són BGP i l'script que heu executat arranca el dimoni *bgpd* amb un fitxer de configuració que permet que els *routers* frontera anunciïn el prefix del seu sistema autònom amb els atributs per defecte. Els routers interns dels sistemes autònom no anuncien cap prefix BGP.

Fixeu-vos en la configuració del *router* R01 de l'AS 102 amb la comanda:

```
lxc-attach -n R01 -- vtysh -c 'show running-config'

root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show run'
Building configuration...

Current configuration:
!
!
interface eth0
 ip address 20.1.0.9/30
!
interface eth1
 ip address 20.1.0.1/30
!
interface eth2
 ip address 10.1.0.17/30
!
router bgp 102
 bgp router-id 10.1.0.17
 network 20.1.0.0/16
 neighbor 10.1.0.18 remote-as 104
 neighbor 20.1.0.2 remote-as 102
 neighbor 20.1.0.10 remote-as 102
!
address-family ipv6
 exit-address-family
 exit
!
router ospf
 passive-interface eth2
 network 10.1.0.16/30 area 0.0.0.0
 network 20.1.0.0/30 area 0.0.0.0
 network 20.1.0.8/30 area 0.0.0.0
!
ip forwarding
 ipv6 forwarding
!
line vty
!
end
```

La configuració BGP del router R01 és:

```
router bgp 102
bgp router-id 10.1.0.17
network 20.1.0.0/16
neighbor 10.1.0.18 remote-as 104
neighbor 20.1.0.2 remote-as 102
neighbor 20.1.0.10 remote-as 102
!
```

La comanda `router bgp 102` indica que R01 pertany a l'AS 102.

La comanda `bgp router-id 10.1.0.17` indica que l'identificador BGP de R01 és 10.1.0.17. Si no s'hagués especificat un identificador concret, el router utilitzaria l'adreça IP més alta configurada en alguna de les seves interfícies.

La comanda `network 20.1.0.0/16` indica que R01 ha d'anunciar el prefix 20.1.0.0/16 via BGP.

La comanda `neighbor A.B.C.D remote-as X` indica que es vol establir una relació de veïnatge (peering) amb el router que té l'adreça IP A.B.C.D de l'AS X. Si l'AS del veí coincideix amb l'AS del propi router, la sessió serà iBGP, si l'AS del veí és diferent de l'AS del propi router, la sessió serà eBGP.

Nota: En les configuracions BGP que es faran en aquesta pràctica cal tenir en compte que:

- Tots els routers BGP d'un mateix AS han de ser veïns BGP (encara que no estiguin directament connectats). En el cas que els routers no estiguin directament connectats, es pot escollir qualsevol adreça IP del veí per a establir la sessió (tenint en compte que el protocol d'encaminament interior i/o les rutes estàtiques del router li permetin enviar paquets a l'adreça IP indicada).
- Quan s'especifica l'adreça IP d'un veí eBGP (és a dir, d'un AS diferent al del propi router) cal especificar una adreça IP del veí que pertanyi a una xarxa directament connectada al router que s'està configurant.

a. **Quants peers (veïns) BGP té configurats R01? Com ha de ser la relació de veïnatge (iBGP o eBGP) amb cadascun d'ells? Per què?**

R01 (que pertany a l'AS 102) té configurats tres peers:

```
neighbor 10.1.0.18 remote-as 104
neighbor 20.1.0.2 remote-as 102
neighbor 20.1.0.10 remote-as 102
```

El veí 10.1.0.18 (R03) pertany a l'AS 104 i, per tant, és un veí eBGP (pertany a un AS diferent al de R01).

Els veïns 20.1.0.2 (R02) i 20.1.0.10 (R08) pertanyen a l'AS 102 i, per tant, són veïns iBGP (pertanyen al mateix AS que R01).

Comproveu el número de sessions BGP que té establertes R01 amb la comanda:

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp summary'

root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp summary'
BGP router identifier 10.1.0.17, local AS number 102
RIB entries 5, using 560 bytes of memory
Peers 3, using 27 KiB of memory

Neighbor      V     AS  MsgRcvd  MsgSent   TblVer  InQ  OutQ  Up/Down  State/PfxRcd
10.1.0.18      4    104       6        8          0      0      0  00:01:03      1
20.1.0.2       4    102       6        8          0      0      0  00:01:04      3
20.1.0.10      4    102       3        8          0      0      0  00:01:04      0

Total number of neighbors 3

Total num. Established sessions 3
Total num. of routes received 4
```

BGP estableixen sessions BGP entre veïns (peers). Per saber quines operacions ha de fer amb cada peer, utilitza una màquina d'estats finits de sis estats: *Idle*, *Connect*, *Active*, *OpenSent*, *OpenConfirm* i *Established*. Per cada sessió es manté una variable d'estat que monitoritza en quin estat està la sessió.

- *Idle*. En aquest estat no s'accepten intents de connexió entrants. El BGP inicialitza tots els *triggers* d'esdeveniments, inicia el procés d'establiment de la sessió TCP i canvia a l'estat *Connect*.
- *Connect*. En aquest estat, el BGP intenta completar l'establiment de la sessió TCP (anomenat 3-way-handshake). Si es completa el 3-way-handshake amb èxit, s'envia un missatge **Open** i es passa a l'estat *OpenState*. Si expira un determinat temporitzador abans que es completi aquesta etapa, es canvia a l'estat *Active*.
- *Active*. Si el BGP no ha pogut establir la sessió TCP amb èxit, entra en aquest estat i intenta establir de nou la sessió TCP. Si ho aconsegueix, envia un missatge **Open** i canvia a l'estat *OpenState*. Si no ho aconsegueix, es passa a l'estat *Idle*.
- *OpenSent*. En aquest estat, el BGP espera rebre un **Open** del peer. Quan rep l'**Open**, en verifica el contingut. Si és vàlid, envia un missatge **Keepalive** i passa a l'estat *OpenConfirm*. Si no és vàlid, envia un missatge **Notification** indicant l'error.
- *OpenConfirm*. En aquest estat, el BGP espera rebre un **Keepalive** del peer. Si es rep el **Keepalive** abans que expiri cap temporitzador, es canvia a l'estat *Established*. Si expira algun temporitzador, es passa a l'estat *Idle*.
- *Established*. En aquest estat, s'envien missatges **Update** amb les rutes best de la taula BGP de cada router. Si hi ha algun error als missatges **Update**, s'envia un missatge **Notification** al peer i es canvia a l'estat *Idle*.

Per cada peer, la comanda 'show ip bgp summary' mostra la següent informació

- **Neighbor**. Adreça IP del veí (la que s'ha configurat)
- **V**. Versió del protocol BGP
- **AS**. AS del veí (el que s'ha configurat)
- **MsgRcvd**. Nombre de missatges BGP rebuts del veí
- **MsgSent**. Nombre de missatges BGP enviats al veí
- **TblVer**. Darrera versió de la taula BGP enviada al veí
- **InQ**. Nombre de missatges del veí que estan en espera de ser processats
- **OutQ**. Nombre de missatges que s'està pendent d'enviar al veí.
- **Up/Down**. Temps que la sessió BGP està en estat *Established* (o en l'estat actual)
- **State/PfxRcd**. Estat actual de la sessió BGP o, quan està *Established*, nombre de prefixes rebuts del veí.

b. Quantes sessions hi ha? En quin estat estan?

R01 té configurats tres peers (veure apartat a) de l'exercici 2). Els tres estan en estat *Established*.

Podeu obtenir informació més detallada de cada sessió amb la comanda:

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp neighbors'
```

La comanda 'show ip bgp neighbors' mostra informació més detallada de cada sessió BGP. No us capfiqueu en mirar tots els camps amb detall, fixeu-vos només en aquells camps que estan en negreta, dels que ja s'ha parlat o es parlarà més endavant a la pràctica.

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp neighbors'

BGP neighbor is 10.1.0.18, remote AS 104, local AS 102, external link
  BGP version 4, remote router ID 10.1.0.18
  BGP state = Established, up for 00:01:14
  Last read 00:00:14, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    4 Byte AS: advertised and received
    Route refresh: advertised and received(old & new)
    Address family IPv4 Unicast: advertised and received
    Graceful Restart Capabilty: advertised and received
      Remote Restart timer is 120 seconds
      Address families by peer: none
  Graceful restart informations:
    End-of-RIB send: IPv4 Unicast
    End-of-RIB received: IPv4 Unicast
  Message statistics:
    Inq depth is 0
    Outq depth is 0
      Sent          Rcvd
    Opens:           2          0
    Notifications:  0          0
    Updates:        3          4
    Keepalives:     3          2
    Route Refresh:  0          0
    Capability:    0          0
    Total:          8          6
  Minimum time between advertisement runs is 3 seconds

  For address family: IPv4 Unicast
    Community attribute sent to this neighbor(all)
    1 accepted prefixes

    Connections established 1; dropped 0
    Last reset never
    External BGP neighbor may be up to 1 hops away.
    Local host: 10.1.0.17, Local port: 179
    Foreign host: 10.1.0.18, Foreign port: 52166
    Nexthop: 10.1.0.17
    Nexthop global: fe80::34d1:aff:fe1c:d460
    Nexthop local: ::

    BGP connection: non shared network
    Read thread: on  Write thread: off
```

```

BGP neighbor is 20.1.0.2, remote AS 102, local AS 102, internal link
BGP version 4, remote router ID 10.1.0.9
BGP state = Established, up for 00:01:15
Last read 00:00:15, hold time is 180, keepalive interval is 60 seconds
Neighbor capabilities:
  4 Byte AS: advertised and received
  Route refresh: advertised and received(old & new)
  Address family IPv4 Unicast: advertised and received
  Graceful Restart Capabilty: advertised and received
    Remote Restart timer is 120 seconds
  Address families by peer:
    none
Graceful restart informations:
  End-of-RIB send: IPv4 Unicast
  End-of-RIB received: IPv4 Unicast
Message statistics:
  Inq depth is 0
  Outq depth is 0
      Sent          Rcvd
  Opens:           2            0
  Notifications:  0            0
  Updates:        3            4
  Keepalives:     3            2
  Route Refresh:  0            0
  Capability:    0            0
  Total:          8            6
Minimum time between advertisement runs is 1 seconds
For address family: IPv4 Unicast
  Community attribute sent to this neighbor(all)
  3 accepted prefixes
    Connections established 1; dropped 0
    Last reset never
    Internal BGP neighbor may be up to 255 hops away.
Local host: 20.1.0.1, Local port: 179
Foreign host: 20.1.0.2, Foreign port: 38832
Nexthop: 20.1.0.1
Nexthop global: fe80::6883:c3ff:fe24:a7da
Nexthop local: ::

BGP connection: non shared network
Read thread: on  Write thread: off

```

```

BGP neighbor is 20.1.0.10, remote AS 102, local AS 102, internal link
BGP version 4, remote router ID 20.1.0.10
BGP state = Established, up for 00:01:15
Last read 00:00:15, hold time is 180, keepalive interval is 60 seconds
Neighbor capabilities:
  4 Byte AS: advertised and received
  Route refresh: advertised and received(old & new)
  Address family IPv4 Unicast: advertised and received
  Graceful Restart Capabilty: advertised and received
    Remote Restart timer is 120 seconds
  Address families by peer:
    none
Graceful restart informations:
  End-of-RIB send: IPv4 Unicast
  End-of-RIB received: IPv4 Unicast
Message statistics:
  Inq depth is 0
  Outq depth is 0
      Sent          Rcvd
  Opens:           2          0
  Notifications:  0          0
  Updates:        3          1
  Keepalives:     3          2
  Route Refresh:  0          0
  Capability:    0          0
  Total:          8          3
Minimum time between advertisement runs is 1 seconds
For address family: IPv4 Unicast
  Community attribute sent to this neighbor(all)
  0 accepted prefixes
  Connections established 1; dropped 0
  Last reset never
  Internal BGP neighbor may be up to 255 hops away.
Local host: 20.1.0.9, Local port: 179
Foreign host: 20.1.0.10, Foreign port: 41022
Nexthop: 20.1.0.9
Nexthop global: fe80::8ce5:45ff:fe8a:4a1
Nexthop local: ::

BGP connection: non shared network
Read thread: on  Write thread: off

```

Observeu la taula BGP de R01 amb la comanda:

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp'

root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp'
BGP table version is 0, local router ID is 10.1.0.17
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

      Network          Next Hop            Metric LocPrf Weight Path
* i20.1.0.0/16        20.1.0.2          0       100      0 i
* >                  0.0.0.0           0         32768 i
*>i30.1.0.0/16       10.1.0.10         0       100      0 103 i
*> 40.1.0.0/16       10.1.0.18         0         0 104 i
* i                  10.1.0.14         0       100      0 104 i

Displayed 3 out of 5 total prefixes
```

Cada línia de la sortida de la comanda ‘show ip bgp’ és un prefix i la seva llista d’atributs. La primera columna especifica el prefix. Si està en blanc (com a les rutes 2 i 5 de la taula anterior), significa que s’està considerant el mateix prefix que a la línia de sobre. La lletra “i” que es veu al principi de les línies 1, 3 i 5 indica que aquestes rutes s’han après via iBGP (d’un router del mateix AS); mentre que la resta de rutes s’han après via eBGP (d’un router d’un AS diferent). Per altra banda, la “i” que surt al final de cada línia és l’atribut ORIGIN. Si hi ha més d’una ruta per a un prefix, només se n’escull una com a millor ruta (best) indicada amb el símbol “>” i és la que s’envia als veïns.

Network	Next Hop	MED	LocalPreference	Weight	AS_Path	Origin
* i20.1.0.0/16	20.1.0.2	0	100	0		i
* > 20.1.0.0/16	0.0.0.0	0		32768		i
* >i30.1.0.0/16	10.1.0.10	0	100	0	103	i
* > 40.1.0.0/16	10.1.0.18	0		0	104	i
* i40.1.0.0/16	10.1.0.14	0	100	0	104	i

Fixeu-vos que algunes rutes tenen un o més atribut sense cap valor associat. Quan un router rep un prefix d’un veí, el rep amb una llista concreta d’atributs. El valor d’aquests atributs és el que es veu a la taula BGP. Els atributs obligatoris (que s’han d’enviar sempre acompañant un prefix) són: NEXT_HOP, AS_PATH i ORIGIN. En les sessions iBGP, a més a més, també és obligatori enviar el LOCAL_PREFERENCE (de fet, aquest atribut no es pot enviar a veïns eBGP i, per tant, només apareix quan la ruta s’ha après d’un veí del mateix AS o quan, com veureu a la segona part de la pràctica, s’utilitza un filtre per modificar-ne el valor).

Com s’explica a la pàgina 15, els criteris que fa servir el BGP del quagga per seleccionar la millor ruta (best) de cada prefix consisteix en comparar la llista d’atributs de les diferents rutes del prefix aplicant els criteris següents:

1. No s’admet una ruta si el router no té cap camí cap al NEXT_HOP del prefix.
2. S’escull la ruta amb **major** pes (WEIGHT). (Criteri local al router).
3. S’escull la ruta amb **major** LOCAL_PREFERENCE. (Criteri global dins un sistema autònom).
4. Es prefereixen les rutes originades de forma local pel router.
5. S’escull la ruta que travessa un menor número d’AS (AS_PATH **més curt**)
6. S’escull la ruta amb un codi ORIGIN **menor**: IGP < EGP < Incomplete.
7. S’escull la ruta amb **menor** MED (Multi-Exit Discriminator):
8. Es prefereix una ruta apresa via eBGP sobre una apresa via iBGP.
9. Es tria la ruta amb menor mètrica (segons el protocol d’encaminament interior) cap al NEXT_HOP.
10. Quan les rutes que es comparen s’han rebut via eBGP, es queda la ruta que ha arribat primer.
11. Es tria la ruta apresa del veí amb menor identificador BGP (que sol ser una de les IPs del router).

En aquesta primera part de la pràctica, els criteris en color gris no serviran per desempatar entre diferents rutes cap a un mateix prefix perquè es considerarà el valor per defecte d’aquests paràmetres i atributs. Sobre el criteri número 6, en aquesta pràctica totes les rutes tindran ORIGIN = IGP i, per tant, tampoc serà un atribut que permeti desempatar. Els criteris 9, 10 i 11 serveixen per desempatar però fixeu-vos que no depenen del valor concret de cap paràmetre (WEIGHT) o atribut (LOCAL_PREFERENCE, AS_PATH, ORIGIN o MED).

- c. Quants prefixes ha après R01 via BGP? Per a cadascun d'ells, apunteu-vos quantes opcions ha après i indiqueu quina ha triat com a **best** i per què (consulteu la llista de criteris de la pàgina 15).

A la taula BGP hi ha rutes cap a tres prefixes: 20.1.0.0/16, 30.1.0.0/16 i 40.1.0.0/16.

1) Prefix 20.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* i20.1.0.0/16	20.1.0.2	0	100	0	i
*>	0.0.0.0	0		32768	i

El prefix 20.1.0.0/16 és de l'AS 102 (que és l'AS de R01). A la taula es veuen dues opcions per arribar-hi. La primera s'aprèn del veí R02 (perquè R02 té configurada la comanda `network 20.1.0.0/16` per anunciar aquest prefix via BGP). La segona hi és perquè R01 té configurada la comanda `network 20.1.0.0/16` per anunciar aquest prefix via BGP i, com que és el propi R01 qui ho anuncia, el next hop és 0.0.0.0. L'opció que surt com a seleccionada “>” és la del propi R01 (en aquest cas perquè el weight per defecte és més gran). Cal tenir present que el weight per defecte sempre val 0 excepte en el cas dels prefixes del propi AS que anuncia un router BGP. D'altra banda, tot i que R08 pertany a l'AS 102, R01 no té cap ruta pel prefix 20.1.0.0/16 on R08 sigui el next hop perquè R08 no anuncia aquest prefix (no té la comanda `network 20.1.0.0/16` a la seva configuració BGP).

2) Prefix 30.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i30.1.0.0/16	10.1.0.10	0	100	0	103 i

Només hi ha una ruta pel prefix 30.1.0.0/16. És vàlida i és la que s'escull.

3) Prefix 40.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 40.1.0.0/16	10.1.0.18	0		0	104 i
* i	10.1.0.14	0	100	0	104 i

Per al prefix 40.1.0.0/16 hi ha dues opcions.

Criteri 1. No s'admet una ruta si el router no té cap camí cap al NEXT_HOP del prefix → Les dues tenen un next-hop accessible per R01 i per tant no és un criteri que permeti escollir-ne una.

Per comprovar-ho, mireu la taula d'encaminament de R01 i busqueu quina ruta es fa servir. Podeu fer servir les següents comandes:

- `ip route list match IPdestí` per veure quines entrades de la taula serien vàlides per enviar paquets a l'adreça IP que es passa com a paràmetre.
- `ip route get IPdestí` per saber quin nexthop es fa servir per enviar paquets a l'adreça IP que es passa com a paràmetre.

```
root@api-mv:~# lxc-attach -n R01 -- ip route list match 10.1.0.18
10.1.0.16/30 dev eth2 proto kernel scope link src 10.1.0.17

root@api-mv:~# lxc-attach -n R01 -- ip route get 10.1.0.18
10.1.0.18 dev eth2 src 10.1.0.17

→ R01 està directament connectat a la xarxa 10.1.0.16/30 per la interfície eth2
```

```
root@api-mv:~# lxc-attach -n R01 -- ip route list match 10.1.0.14
10.1.0.12/30 via 20.1.0.2 dev eth1 proto zebra metric 20
root@api-mv:~# lxc-attach -n R01 -- ip route get 10.1.0.14
10.1.0.14 via 20.1.0.2 dev eth1 src 20.1.0.1
→ R01 utilitzà el next hop 20.1.0.2 per enviar paquets a l'adreça 10.1.0.14
```

Criteri 5. S'escull la ruta que travessa un menor número d'AS (AS_PATH més curt) → Les dues opcions tenen un AS_PATH de la mateixa longitud i, per tant, aquest criteri no serveix per desempatar.

Criteri 6. S'escull la ruta amb un codi ORIGIN menor → Les dues opcions tenen el mateix ORIGIN (és la “i” que surt al final de les dues línies) i, per tant, aquest criteri no serveix per desempatar.

Criteri 8. Es prefereix una ruta apresa via eBGP sobre una apresa via iBGP → La segona opció s'ha après via iBGP (com ho identifica la “i” que hi ha al principi de la línia). **Com que la ruta apresa via eBGP preval sobre la iBGP, s'escull la primera ruta** (la que té next-hop 10.1.0.18).

- d. Observeu la taula BGP de R02, R03 i R05 i, com en l'apartat anterior, raoneu quins prefixes ha après via BGP, quantes opcions té per arribar a cadascun d'ells i, si n'hi ha més d'una, com ha escollit la best (consulteu la llista de criteris de la pàgina 15).

```
root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
* i20.1.0.0/16        20.1.0.1           0       100     0 i
* >                  0.0.0.0             0           32768 i
*> 30.1.0.0/16        10.1.0.10          0           0 103 i
* i40.1.0.0/16        10.1.0.18          0       100     0 104 i
*>                  10.1.0.14          0           0 104 i
```

- 1) Prefix 20.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* i20.1.0.0/16	20.1.0.1	0	100	0	i
* >	0.0.0.0	0		32768	i

L'opció que surt com a seleccionada “>” és la del propi R02 (el weight per defecte és més gran).

- 2) Prefix 30.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* > 30.1.0.0/16	10.1.0.10	0		0	103 i

Només hi ha una ruta pel prefix 30.1.0.0/16. És vàlida i és la que s'escull.

- 3) Prefix 40.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
* >	10.1.0.14	0		0	104 i

Hi ha dues rutes. Les dues són vàlides (criteri 1), les dues tenen la mateixa longitud d'AS_PATH (criteri 5), les dues tenen el mateix ORIGIN (criteri 6) però la primera és iBGP i la segona és eBGP . **Com que la ruta apresa via eBGP preval sobre la iBGP (criteri 8), s'escull la segona ruta** (la que té next-hop 10.1.0.14).

```
root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
*  20.1.0.0/16        10.1.0.17          0        0 102 i
*>                  10.1.0.13          0        0 102 i
*  30.1.0.0/16        10.1.0.17          0        0 102 103 i
*>                  10.1.0.13          0        0 102 103 i
* i40.1.0.0/16       40.1.0.6           0     100      0 i
*>                  0.0.0.0           0        32768 i
```

1) Prefix 20.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.17	0	100	0	104 i
*>	10.1.0.13	0		0	104 i

Hi ha dues rutes. Les dues són vàlides (**criteri 1**), les dues tenen la mateixa longitud d'AS_PATH (**criteri 5**), les dues tenen el mateix ORIGIN (**criteri 6**) les dues s'han après via eBGP (**criteri 8**).

Criteri 9. Es tria la ruta amb menor mètrica (segons el protocol d'encaminament interior) cap al NEXT_HOP → Les dues rutes són externes i, per tant, aquest criteri no serveix per desempatar.

Criteri 10. Quan les rutes que es comparen s'han rebut via eBGP, es queda la ruta que ha arribat primer. L'ordre en el que apareixen les rutes a la taula indica l'ordre en que s'ha après els prefixes: la ruta que està més avall és la més antiga i, a mesura que es van aprenent, es van col·locant a sobre. Per tant, **la ruta amb next-hop 10.1.0.13 s'ha après abans i és la que es selecciona**.

2) Prefix 30.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* 30.1.0.0/16	10.1.0.17	0	102	103	i
*>	10.1.0.13	0		0	102 103 i

Hi ha dues rutes. Les dues són vàlides (**criteri 1**), les dues tenen la mateixa longitud d'AS_PATH (**criteri 5**), les dues tenen el mateix ORIGIN (**criteri 6**) les dues s'han après via eBGP (**criteri 8**), les dues són externes (no aplica el **criteri 9**), però com que **de les dues rutes apreses via eBGP, la que té el next-hop 10.1.0.13 s'ha après abans, és la seleccionada** (**criteri 10**).

4) Prefix 40.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	40.1.0.6	0	100	0	i
*>	0.0.0.0	0		32768	i

L'opció que surt com a seleccionada ">" és la del propi R03 (el weight per defecte és més gran).

```
root@api-mv:~# lxc-attach -n R05 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
*> 20.1.0.0/16        10.1.0.9           0        0 102 i
*> 30.1.0.0/16        0.0.0.0           0        32768 i
*> 40.1.0.0/16        10.1.0.9           0        0 102 104 i
```

A R05, per cada prefix només hi ha una ruta i és la que es selecciona.

Observeu la taula d'encaminament d'aquests routers (show ip route) i fixeu-vos amb les rutes que hi ha.

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
* i20.1.0.0/16        20.1.0.2            0       100      0 i
*->                  0.0.0.0             0           32768 i
*>i30.1.0.0/16        10.1.0.10           0       100      0 103 i
*> 40.1.0.0/16        10.1.0.18           0           0 104 i
* i                   10.1.0.14           0       100      0 104 i

root@api-mv:~# lxc-attach -n R01 -- vtysh -c "show ip route"
O>* 10.1.0.8/30 [110/20] via 20.1.0.2, eth1, 02:04:03
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth1, 02:04:03
O  10.1.0.16/30 [110/10] is directly connected, eth2, 02:04:49
C>* 10.1.0.16/30 is directly connected, eth2
O  20.1.0.0/30 [110/10] is directly connected, eth1, 02:04:08
C>* 20.1.0.0/30 is directly connected, eth1
O>* 20.1.0.4/30 [110/20] via 20.1.0.10, eth0, 02:03:53
*               via 20.1.0.2, eth1, 02:03:53
O  20.1.0.8/30 [110/10] is directly connected, eth0, 02:04:03
C>* 20.1.0.8/30 is directly connected, eth0
O>* 20.1.0.64/27 [110/20] via 20.1.0.2, eth1, 02:04:03
O>* 20.1.0.96/27 [110/20] via 20.1.0.10, eth0, 02:03:53
B> 30.1.0.0/16 [200/0] via 10.1.0.10 (recursive), 01:56:51
*               via 20.1.0.2, eth1, 01:56:51
B>* 40.1.0.0/16 [20/0]  via 10.1.0.18, eth2, 01:56:52
C>* 127.0.0.0/8 is directly connected, lo
```

Fixeu-vos en el següent:

- La ruta BGP seleccionada del prefix del propi AS (20.1.0.0/16) és el propi R01 (next-hop 0.0.0.0) i no surt a la taula d'encaminament.
- La ruta BGP seleccionada del prefix 30.1.0.0/16 s'ha après via iBGP i el next-hop BGP (que és l'adreça IP 10.1.0.10) no està directament connectat a R01. Per poder accedir a aquest next-hop, R01 fa servir la primera ruta de la taula d'encaminament amb next-hop 20.1.0.2. Per això, a la ruta de la taula d'encaminament del prefix 30.1.0.0/16 surt el missatge via 10.1.0.10 (recursive) via 20.1.0.2.

```
root@api-mv:~# lxc-attach -n R05 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
*> 20.1.0.0/16        10.1.0.9            0       0 102 i
*> 30.1.0.0/16        0.0.0.0             0           32768 i
*> 40.1.0.0/16        10.1.0.9            0       0 102 104 i

root@api-mv:~# lxc-attach -n R05 -- vtysh -c "show ip route"
C>* 10.1.0.4/30 is directly connected, eth2
C>* 10.1.0.8/30 is directly connected, eth0
B>* 20.1.0.0/16 [20/0] via 10.1.0.9, eth0, 01:57:54
C>* 30.1.0.96/27 is directly connected, eth1
B>* 40.1.0.0/16 [20/0] via 10.1.0.9, eth0, 01:57:54
C>* 127.0.0.0/8 is directly connected, lo
```

```
root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* i20.1.0.0/16    20.1.0.1            0     100      0 i
* >              0.0.0.0             0           32768 i
*> 30.1.0.0/16    10.1.0.10           0           0 103 i
* i40.1.0.0/16    10.1.0.18           0     100      0 104 i
*>              10.1.0.14           0           0 104 i
```

```
root@api-mv:~# lxc-attach -n R02 -- vtysh -c "show ip route"
O  10.1.0.8/30 [110/10] is directly connected, eth0, 02:04:53
C>* 10.1.0.8/30 is directly connected, eth0
O  10.1.0.12/30 [110/10] is directly connected, eth1, 02:04:52
C>* 10.1.0.12/30 is directly connected, eth1
O>* 10.1.0.16/30 [110/20] via 20.1.0.1, eth2, 02:04:07
O  20.1.0.0/30 [110/10] is directly connected, eth2, 02:04:52
C>* 20.1.0.0/30 is directly connected, eth2
O  20.1.0.4/30 [110/10] is directly connected, eth3, 02:04:52
C>* 20.1.0.4/30 is directly connected, eth3
O>* 20.1.0.8/30 [110/20] via 20.1.0.1, eth2, 02:04:07
*                      via 20.1.0.5, eth3, 02:04:07
O  20.1.0.64/27 [110/10] is directly connected, eth4, 02:04:52
C>* 20.1.0.64/27 is directly connected, eth4
O>* 20.1.0.96/27 [110/20] via 20.1.0.5, eth3, 02:04:07
B>* 30.1.0.0/16 [20/0] via 10.1.0.10, eth0, 01:56:56
B>* 40.1.0.0/16 [20/0] via 10.1.0.14, eth1, 01:56:56
C>* 127.0.0.0/8 is directly connected, lo
```

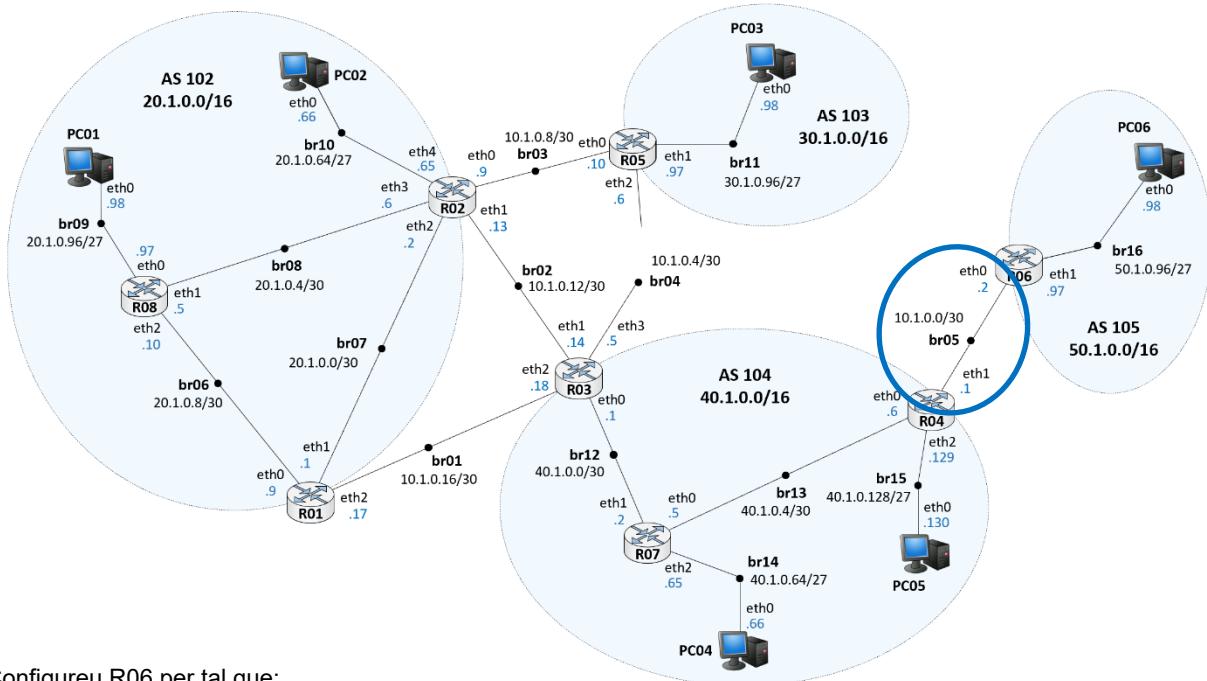
```
root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* 20.1.0.0/16    10.1.0.17           0           0 102 i
*>              10.1.0.13           0           0 102 i
* 30.1.0.0/16    10.1.0.17           0           0 102 103 i
*>              10.1.0.13           0           0 102 103 i
* i40.1.0.0/16    40.1.0.6            0     100      0 i
*>              0.0.0.0             0           32768 i
```

```
root@api-mv:~# lxc-attach -n R03 -- vtysh -c "show ip route"
R>* 10.1.0.0/30 [120/3] via 40.1.0.2, eth0, 02:04:53
C>* 10.1.0.4/30 is directly connected, eth3
C>* 10.1.0.12/30 is directly connected, eth1
C>* 10.1.0.16/30 is directly connected, eth2
B>* 20.1.0.0/16 [20/0] via 10.1.0.13, eth1, 01:57:00
B>* 30.1.0.0/16 [20/0] via 10.1.0.13, eth1, 01:56:57
C>* 40.1.0.0/30 is directly connected, eth0
R>* 40.1.0.4/30 [120/2] via 40.1.0.2, eth0, 02:04:53
R>* 40.1.0.64/27 [120/2] via 40.1.0.2, eth0, 02:04:53
R>* 40.1.0.128/27 [120/3] via 40.1.0.2, eth0, 02:04:53
C>* 127.0.0.0/8 is directly connected, lo
```

Exercici 3. Intercanvi d'informació quan s'estableix una sessió BGP

A l'AS 105 de l'escenari no hi ha cap *router* intern (i no s'ha configurat cap protocol d'encaminament interior).

Connecteu R06 a la xarxa on hi ha la interfície eth1 de R04: plug-if-br R06-eth0 br05



Configureu R06 per tal que:

- Tingui l'adreça IP de la seva interfície eth0 com a identificador BGP del router.
- Anunciï el prefix 50.1.0.0/16.
- Tingui a R04 del sistema autònom 104 com a veí BGP.

Per fer la configuració que es demana, cal fer el següent:

```
root@api-mv:~# lxc-attach -n R06 -- vtysh
```

```
Hello, this is Quagga (version 1.2.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
R06# configure terminal
R06(config)# router bgp 105
R06(config-router)# bgp router-id 10.1.0.2          → ID del rotuer = 10.1.0.2
R06(config-router)# network 50.1.0.0/16            → prefix que anuncia = 50.1.0.0/16
R06(config-router)# neighbor 10.1.0.1 remote-as 104   → veí BGP = 10.1.0.1 de l'AS 104
R06(config-router)# exit
R06(config)# exit
```

La configuració de R06 queda de la següent manera (només es mostra la part del router BGP):

```
R06# show running-config
( ... )
router bgp 105
bgp router-id 10.1.0.2
network 50.1.0.0/16
neighbor 10.1.0.1 remote-as 104
( ... )
```

Amb el Wireshark, captureu paquets als bridges br05, br13, br01, br02, br03, br07, br08 i br06.

La captura de paquets la teniu al fitxer **CapturaPart1Exercici3.pcapng**

Observeu i apunteu-vos la taula BGP de R04 i R06 actual amb la comanda:

```
lxc-attach -n R04 -- vtysh -c 'show ip bgp'
lxc-attach -n R06 -- vtysh -c 'show ip bgp'
```

```
root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip bgp'
BGP table version is 0, local router ID is 10.1.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

      Network          Next Hop            Metric LocPrf Weight Path
*>i20.1.0.0/16      10.1.0.13          0       100      0 102 i
*>i30.1.0.0/16      10.1.0.13          100     0 102 103 i
* i40.1.0.0/16      40.1.0.1           0       100      0 i
*>                  0.0.0.0            0           32768 i

Displayed 3 out of 4 total prefixes
```

```
root@api-mv:~# lxc-attach -n R06 -- vtysh -c 'show ip bgp'
BGP table version is 0, local router ID is 10.1.0.2
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

      Network          Next Hop            Metric LocPrf Weight Path
*> 50.1.0.0/16      0.0.0.0            0           32768 i

Displayed 1 out of 1 total prefixes
```

Com que encara no s'ha configurat el veí R06 a la configuració BGP de R04, quan s'observa la informació dels peers configurats a R06, el que s'obté és el següent:

```
root@api-mv:~# lxc-attach -n R06 -- vtysh -c 'show ip bgp summary'
BGP router identifier 10.1.0.2, local AS number 105
RIB entries 1, using 112 bytes of memory
Peers 1, using 9088 bytes of memory

Neighbor      V   AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.0.1      4   104      0       38          0     0     0 never    Active

Total number of neighbors 1
Total num. Established sessions 0
Total num. of routes received 0
```

Fixeu-vos que l'estat BGP de la sessió és *Active* perquè la sessió TCP entre R04 i R06 es pot establir (es pot completar amb èxit el 3-way-handshake perquè s'ha connectat el cable entre R04 i R06) i R06 pot enviar l'Open per intentar iniciar la sessió BGP però no rep l'Open de R04 perquè R04 encara no ha configurat R06 com a veí.

Obriu un terminal al **PC** i acobleu-lo a R04 (`lxc-attach -n R04`). Afegiu el nou veí R06 a la configuració BGP de R04.

```
root@api-mv:~# lxc-attach -n R04 -- vtysh

R04# show run
( ... )
router bgp 104
bgp router-id 10.1.0.1
network 40.1.0.0/16
neighbor 40.1.0.1 remote-as 104
neighbor 40.1.0.5 remote-as 104
( ... )
R04# configure terminal
R04(config)# router bgp 104
R04(config-router)# neighbor 10.1.0.2 remote-as 105 → veí BGP = 10.1.0.2 de l'AS 105
R04(config-router)# exit
R04(config)# exit
R04# show run
( ... )
router bgp 104
bgp router-id 10.1.0.1
network 40.1.0.0/16
neighbor 10.1.0.2 remote-as 105 → ara apareix R06 com a veí
neighbor 40.1.0.1 remote-as 104
neighbor 40.1.0.5 remote-as 104
( ... )
```

Verifiqueu que s'estableix la sessió BGP entre els dos *routers* (`show ip bgp summary`).

```
root@api-mv:~# lxc-attach -n R06 -- vtysh -c 'show ip bgp summary'
BGP router identifier 10.1.0.2, local AS number 105
RIB entries 7, using 784 bytes of memory
Peers 1, using 9088 bytes of memory

Neighbor      V     AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.0.1      4    104      5      54          0      0    0 00:00:28        3

Total number of neighbors 1
Total num. Established sessions 1
Total num. of routes received      3

root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip bgp summary'
BGP router identifier 10.1.0.1, local AS number 104
RIB entries 7, using 784 bytes of memory
Peers 3, using 27 KiB of memory

Neighbor      V     AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.0.2      4    105      4      7          0      0    0 00:00:36        1
40.1.0.1      4    104     15     15          0      0    0 00:09:40        3
40.1.0.5      4    104     11     16          0      0    0 00:09:40        0

Total number of neighbors 3
Total num. Established sessions 3
Total num. of routes received      4
```

Observant la captura de paquets al br05, respondeu a les preguntes següents:

La captura de paquets la teniu al fitxer [CapturaPart1Exercici3.pcapng](#)

a. Quin protocol de transport i quin port utilitza BGP?

Observant qualsevol paquet BGP de la captura es pot comprovar que s'utilitza el protocol de transport TCP. El port de servei reservat per al protocol BGP és el 179. Fixeu-vos que, com que R04 i R06 estan connectats (hi ha link entre tots dos) i el primer router on s'ha configurat el peer BGP és R06, és aquest router el que envia missatges Open al veí R04 (que inicialment encara no està configurat). Com que R06 és el primer en enviar l'Open, escull un port local lliure (a l'exemple de la captura és el port 36486) i fixa el port destí a 179 (port reservat per BGP).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.1.0.2	10.1.0.1	BGP	125	OPEN Message
> Frame 1: 125 bytes on wire (1000 bits), 125 bytes captured (1000 bits) on interface br05, id 3						
> Ethernet II, Src: c6:02:bd:83:bd:65 (c6:02:bd:83:bd:65), Dst: 2a:b6:3a:e1:2e:fa (2a:b6:3a:e1:2e:fa)						
> Internet Protocol Version 4, Src: 10.1.0.2, Dst: 10.1.0.1						
▼ Transmission Control Protocol, Src Port: 36486, Dst Port: 179, Seq: 1, Ack: 1, Len: 59						
Source Port: 36486						
Destination Port: 179						
[Stream index: 0]						
[TCP Segment Len: 59]						
Sequence number: 1 (relative sequence number)						

b. Quins missatges s'intercanvien per establir la sessió BGP? Quina informació contenen?

Per establir la sessió BGP, els routers s'intercanvien **missatges OPEN**. Un missatge OPEN conté els camps:

- **Version:** Versió del protocol BGP
- **My AS:** Número d'AS del router que envia el missatge OPEN
- **Hold Time:** Determina el temps màxim que un sessió BGP es manté establerta sense que s'enviïn missatge Update o Keepalive. La implementació de BGP que s'utilitza a la pràctica fixa el valor per defecte d'aquest temporitzador a 180 segons.
- **BGP Identifier:** Identificador del router que envia el missatge OPEN. Si no es fixa manualment, la implementació de BGP que s'utilitza en aquesta pràctica escull l'adreça IP més alta configurada al router.
- **Optional Parameters:** Permet enviar altres paràmetres opcionals (*però no es treballarà en aquesta pràctica*)

A la captura, els missatges Open que permeten l'establiment de la sessió són els paquets 21 i 22.

c. Quins paquets s'utilitzen per confirmar l'establiment de sessió i mantenir la relació de veïnatge? Cada quant s'envien?

Per confirmar l'establiment de sessió i per mantenir la relació de veïnatge s'utilitzen els **missatges KEEPALIVE**. Per defecte, s'envien cada 1/3 del temps de Hold Time. Com que en aquesta pràctica el Hold Times és de 180 segons, s'envien missatges Keepalive cada 60 segons.

A la captura, els missatges Keepalive que serveixen per confirmar l'establiment de la sessió són els paquets 22, 23 (que conté 2 Keepalive) i 24.

- d. Quin tipus de missatge s'utilitza per intercanviar informació d'encaminament entre els dos routers? Quines rutes s'intercanvien a l'establir la sessió?

Per intercanviar la informació d'encaminament s'utilitzen els **missatges UPDATE**. Quan s'estableix la sessió entre dos routers BGP, s'intercanvien tots les best de la seva taula BGP.

Paquet 25: 10.1.0.2 (R06) → 10.1.0.1 (R04)

Prefix: 50.1.0.0/16
ORIGIN: IGP
AS_PATH: 105
NEXT_HOP: 10.1.0.2
MULTI_EXIT_DISC: 0

R04 i R06 són peers eBGP. Els atributs obligatoris que s'han d'anunciar acompañant a cada prefix en una sessió eBGP són: ORIGIN, AS_PATH i NEXT_HOP.

Paquet 26: 10.1.0.1 (R04) → 10.1.0.2 (R06)

Prefix: 20.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 102
NEXT_HOP: 10.1.0.1

Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 102 103
NEXT_HOP: 10.1.0.1

Prefix: 40.1.0.0/16
ORIGIN: IGP
AS_PATH: 104
NEXT_HOP: 10.1.0.1
MULTI_EXIT_DISC: 0

ORIGIN (en aquesta pràctica sempre serà IGP perquè els prefixes dels ASs que anuncien els routers es configuren amb la comanda network)

MED (es veurà a la segona part de la pràctica, per defecte és 0)

Torneu a observar la taula BGP de R04 i R06 (show ip bgp) després d'haver-se establert la sessió BGP.

- e. Han aparegut noves rutes? Raoneu quines i com s'han après?

Comparant les rutes de la taula BGP abans i després d'afegir el peer, es pot observar que s'han après les rutes cap als prefixes anunciats pel veí que no es coneixien. És a dir, la ruta al prefix 50.1.0.0/16 a R04 i les rutes als prefixes 20.1.0.0/16, 30.1.0.0/16 i 40.1.0.0/16 a R06.

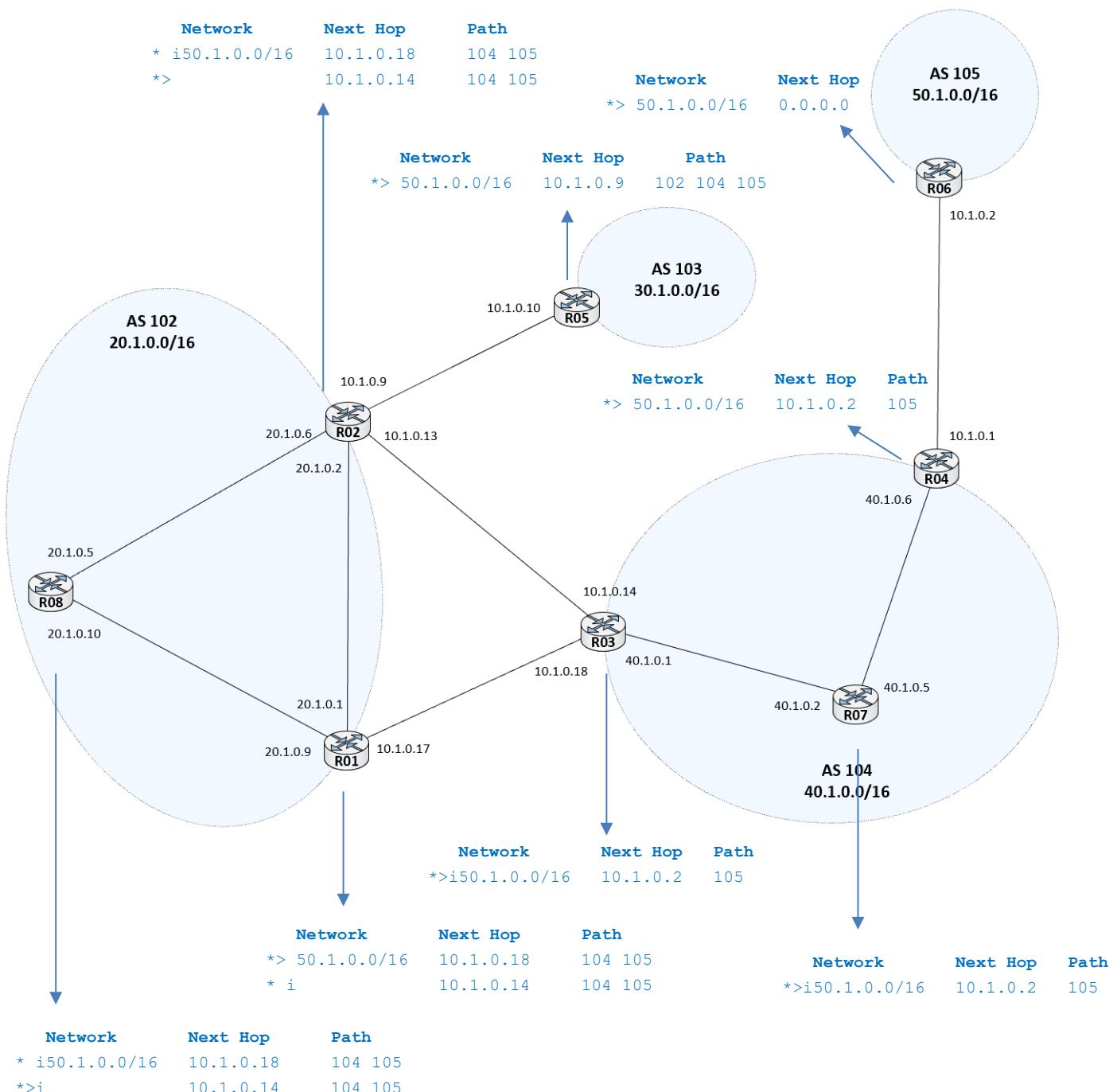
```
root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
Abans:
*->i20.1.0.0/16      10.1.0.13           0       100      0 102 i
*->i30.1.0.0/16      10.1.0.13           100     0 102 103 i
* i40.1.0.0/16        40.1.0.1            0       100      0 i
*->                   0.0.0.0             0           32768 i
Després:
*->i20.1.0.0/16      10.1.0.13           0       100      0 102 i
*->i30.1.0.0/16      10.1.0.13           100     0 102 103 i
* i40.1.0.0/16        40.1.0.1            0       100      0 i
*->                   0.0.0.0             0           32768 i
*-> 50.1.0.0/16        10.1.0.2            0           0 105 i

root@api-mv:~# lxc-attach -n R06 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
Abans:
*-> 50.1.0.0/16        0.0.0.0             0           32768 i
Després:
*-> 20.1.0.0/16        10.1.0.1            0       104 102 i
*-> 30.1.0.0/16        10.1.0.1            0       104 102 103 i
*-> 40.1.0.0/16        10.1.0.1            0       104 i
*-> 50.1.0.0/16        0.0.0.0             0           32768 i
```

Observeu la taula BGP de la resta de routers de l'escenari (`show ip bgp`) i verifiqueu que han après a arribar al prefix anunciat per R06.

El diagrama següent mostra la part de la taula BGP de cadascun dels routers on surt el prefix 50.1.0.0/16 (especificant només si és una ruta apresa via iBGP o eBGP i el valor dels atributs NEXT_HOP i AS_PATH). Com es pot comprovar, tots els routers han après a arribar al prefix de l'AS 105. Fixeu-vos que quan s'envia un prefix via iBGP no es modifica el next_hop ni l'AS_PATH. En canvi, quan s'envia un prefix via eBGP s'inclou el número d'AS del router que envia el paquet al principi de l'AS_PATH i es modifica el next_hop.

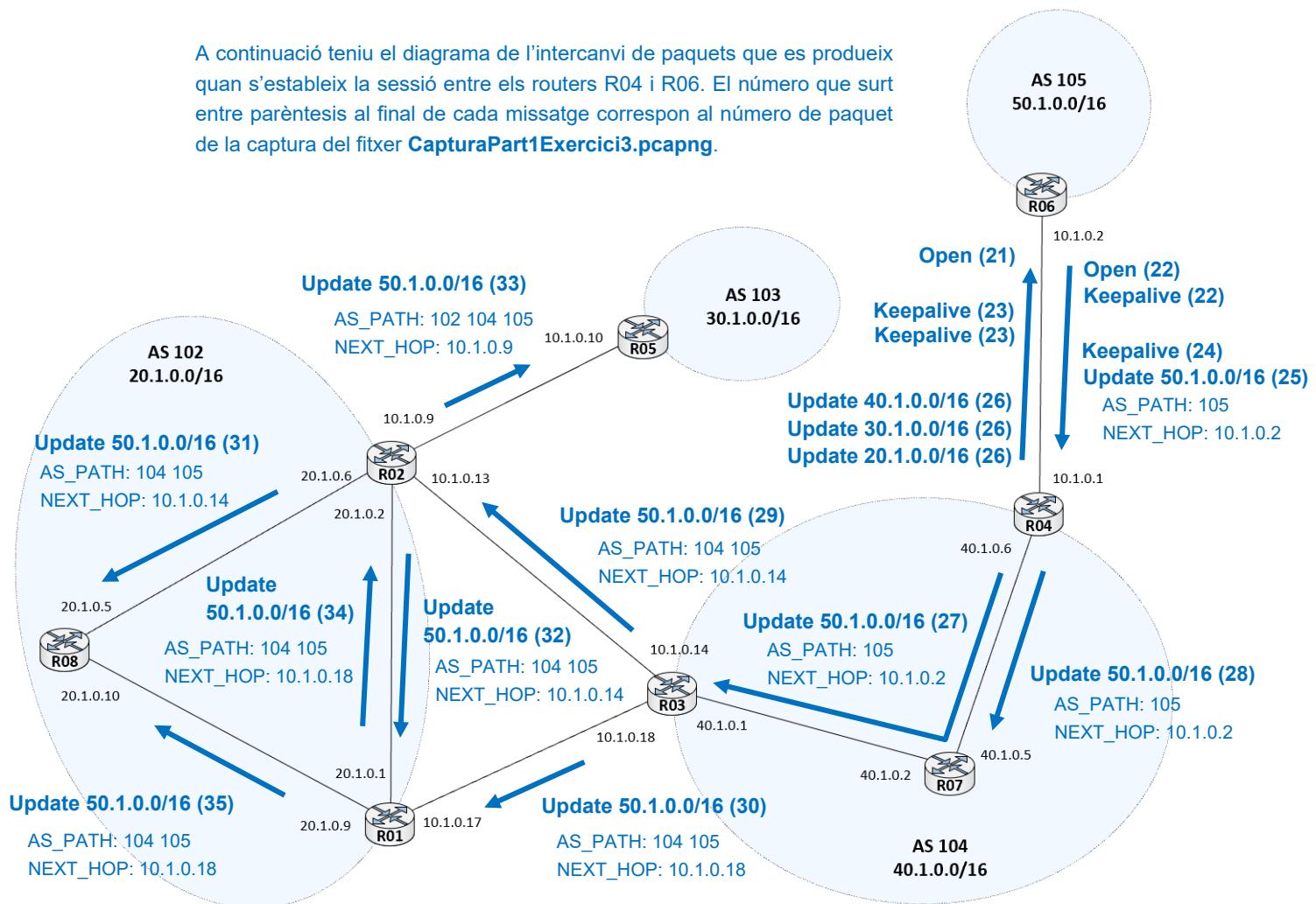
Fixeu-vos que els dos routers frontera de l'AS 102 (R01 i R02), han après dues rutes cap al prefix 50.1.0.0/16 que tenen la mateixa longitud d'AS_PATH. Com que una és eBGP i l'altra és iBGP, marquen com a best la ruta apresa via eBGP. El router intern de l'AS 102 (R08) també té dues options (són les best dels seus veïns). En aquest cas les dues les ha après via iBGP (el criteri 8 no li permet desempatar), el cost OSPF al nexthop BGP és 20 en ambdós casos (el criteri 9 no li permet desempatar) i no són eBGP (el criteri 10 no aplica). Fa servir el criteri 11 per desempatar, és a dir, tria la ruta apresa del veí amb menor identificador BGP. Si comproveu la configuració, l'ID BGP de R02 és 10.1.0.9 i l'ID BGP de R01 és 10.1.0.17, per tant, es queda amb la ruta amb next-hop 10.1.0.10 que ha après de R02.



Atureu el Wireshark i busqueu a les captures els paquets que permeten a cada router aprendre rutes per anar al prefix de l'AS 105.

- f. Comenteu el contingut d'aquests paquets. És a dir, per cada router fixeu-vos amb els atributs que associa al prefix de l'AS 105 quan l'anuncia a cadascun dels seus veïns.

A continuació teniu el diagrama de l'intercanvi de paquets que es produeix quan s'estableix la sessió entre els routers R04 i R06. El número que surt entre parèntesis al final de cada missatge correspon al número de paquet de la captura del fitxer [CapturaPart1Exercici3.pcapng](#).



Després de l'establiment de la sessió BGP (paquets 21 a 24), els dos routers s'han d'enviar totes les best de la seva taula BGP (paquets 25 i 26).

Com que R04 aprèn un prefix nou (50.1.0.0/16), l'ha d'anunciar a tots els seus veïns BGP, excepte el veí de qui ha après la ruta, o sigui, R06. Per això R04 envia un Update a R03 (paquet 27) i un Update a R07 (paquet 28). Com que tots dos són veïns iBGP, R04 no modifica el NEXT_HOP ni l'AS_PATH de la llista d'atributs del prefix.

R03 aprèn el prefix nou (50.1.0.0/16) i l'anuncia a tots els seus veïns BGP excepte a R4 → Envia un Update a R02 del prefix 50.1.0.0/16 (paquet 29) i Update del mateix prefix a R01 (paquet 30). Els dos veïns són eBGP i, per tant, modifica el NEXT_HOP i l'AS_PATH de la llista d'atributs del prefix.

R02 aprèn el prefix nou (50.1.0.0/16) i l'anuncia a tots els seus veïns BGP excepte a R3 → Update a R08 (paquet 31), Update a R01 (paquet 32) i Update a R05 (paquet 33).

R01 aprèn el prefix nou (50.1.0.0/16) i l'anuncia a tots els seus veïns BGP excepte a R3 → Update a R02 (paquet 34) i Update a R08 (paquet 35). A més a més, per mantenir la relació de veïnatge s'envien missatges Keepalive periòdicament (cada 1/3 del temps de hold time negociat a l'establir la sessió BGP).

R07 i R08 no envien cap paquet perquè les rutes apreses via iBGP no s'envien als veïns iBGP.

Exercici 4. Canvis en la topologia de l'escenari

Observeu i apunteu-vos (o guardeu-vos en un fitxer) la taula BGP de R01, R02, R03, R04 i R05.

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* i20.1.0.0/16    20.1.0.2          0     100      0 i
*>               0.0.0.0           0         32768 i
*>i30.1.0.0/16   10.1.0.10         0     100      0 103 i
*> 40.1.0.0/16   10.1.0.18         0         0 104 i
* i               10.1.0.14         0     100      0 104 i
* i50.1.0.0/16   10.1.0.14         100    0 104 105 i
*>               10.1.0.18         0     100      0 104 105 i

root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* i20.1.0.0/16    20.1.0.1          0     100      0 i
*>               0.0.0.0           0         32768 i
*> 30.1.0.0/16   10.1.0.10         0         0 103 i
* i40.1.0.0/16   10.1.0.18         0     100      0 104 i
*>               10.1.0.14         0         0 104 i
* i50.1.0.0/16   10.1.0.18         100    0 104 105 i
*>               10.1.0.14         0         0 104 105 i

root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* 20.1.0.0/16    10.1.0.17         0         0 102 i
*>               10.1.0.13         0         0 102 i
* 30.1.0.0/16   10.1.0.17         0         0 102 103 i
*>               10.1.0.13         0         0 102 103 i
* i40.1.0.0/16   40.1.0.6          0     100      0 i
*>               0.0.0.0           0         32768 i
*>i50.1.0.0/16   10.1.0.2          0     100      0 105 i

root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
*>i20.1.0.0/16   10.1.0.13         0     100      0 102 i
*>i30.1.0.0/16   10.1.0.13         100    0 102 103 i
* i40.1.0.0/16   40.1.0.1          0     100      0 i
*>               0.0.0.0           0         32768 i
*> 50.1.0.0/16   10.1.0.2          0         0 105 i

root@api-mv:~# lxc-attach -n R05 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
*> 20.1.0.0/16   10.1.0.9          0         0 102 i
*> 30.1.0.0/16   0.0.0.0           0         32768 i
*> 40.1.0.0/16   10.1.0.9          0         0 102 104 i
*> 50.1.0.0/16   10.1.0.9          0         0 102 104 105 i
```

Captureu amb el Wireshark els paquets als bridges br03, br08, br01, br02, br04 i br12.

La captura de paquets la teniu al fitxer **CapturaPart1Exercici4.pcapng**

Connecteu la interfície eth2 de R05 al br04 amb la comanda: `plug-if-br R05-eth2 br04`

Configureu R05 i R03 per tal que siguin veïns BGP.

```
root@api-mv:~# lxc-attach -n R03 -- vtysh
R03# sh run
(...)
router bgp 104
bgp router-id 10.1.0.18
network 40.1.0.0/16
neighbor 10.1.0.13 remote-as 102
neighbor 10.1.0.17 remote-as 102
neighbor 40.1.0.2 remote-as 104
neighbor 40.1.0.6 remote-as 104
(...)
R03# conf term
R03(config)# router bgp 104
R03(config-router)# neighbor 10.1.0.6 remote-as 103
R03(config-router)# ^Z
R03# sh run
(...)
router bgp 104
bgp router-id 10.1.0.18
network 40.1.0.0/16
neighbor 10.1.0.6 remote-as 103
neighbor 10.1.0.13 remote-as 102
neighbor 10.1.0.17 remote-as 102
neighbor 40.1.0.2 remote-as 104
neighbor 40.1.0.6 remote-as 104
(....)
```

R03# sh ip bgp summary										
Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd	
10.1.0.6	4	103	0	3	0	0	0	never	Active	
10.1.0.13	4	102	22	23	0	0	0	00:17:07	2	
10.1.0.17	4	102	22	25	0	0	0	00:17:06	2	
40.1.0.2	4	104	19	25	0	0	0	00:17:07	0	
40.1.0.6	4	104	21	25	0	0	0	00:17:07	2	

```

root@api-mv:~# lxc-attach -n R05 -- vtysh

R05# sh run
(...)
router bgp 103
bgp router-id 10.1.0.10
network 30.1.0.0/16
neighbor 10.1.0.9 remote-as 102
(...)
R05# conf term
R05(config)# router bgp 103
R05(config-router)# neighbor 10.1.0.5 remote-as 104
R05(config-router)# ^Z
R05# sh run
(...)
router bgp 103
bgp router-id 10.1.0.10
network 30.1.0.0/16
neighbor 10.1.0.5 remote-as 104
neighbor 10.1.0.9 remote-as 102
(...)

```

Verifiqueu que s'estableix la sessió BGP entre els dos *routers*.

```

R05# sh ip bgp summary
BGP router identifier 10.1.0.10, local AS number 103
RIB entries 7, using 784 bytes of memory
Peers 2, using 18 KiB of memory

Neighbor      V   AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.0.5      4   104     8       9          0       0     0 00:00:21      3
10.1.0.9      4   102    23      26          0       0     0 00:18:43      3

Total number of neighbors 2
Total num. Established sessions 2
Total num. of routes received      6

```

```

root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip bgp summary'
BGP router identifier 10.1.0.18, local AS number 104
RIB entries 7, using 784 bytes of memory
Peers 5, using 44 KiB of memory

```

```

Neighbor      V   AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.0.6      4   103     7       28         0       0     0 00:01:00      2
10.1.0.13     4   102    24      26          0       0     0 00:19:23      2
10.1.0.17     4   102    24      28          0       0     0 00:19:22      2
40.1.0.2      4   104    21      28          0       0     0 00:19:23      0
40.1.0.6      4   104    23      28          0       0     0 00:19:23      2

Total number of neighbors 5
Total num. Established sessions 5
Total num. of routes received      8

```

Mireu les taules BGP dels routers R01, R02, R03, R04 i R05. En aquest exercici analitzareu els canvis que hi ha hagut a les taules comparant-les amb les que tenien el routers abans d'establir el nou peer.

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* i20.1.0.0/16    20.1.0.2          0     100     0 i
* >              0.0.0.0          0           32768 i
*  30.1.0.0/16    10.1.0.18         0           0 104 103 i
* >i             10.1.0.10         0     100     0 103 i
* > 40.1.0.0/16   10.1.0.18         0           0 104 i
* i               10.1.0.14         0     100     0 104 i
* i50.1.0.0/16   10.1.0.14         100          0 104 105 i
* >              10.1.0.18         0     100     0 104 105 i

root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
* i20.1.0.0/16    20.1.0.1          0     100     0 i
* >              0.0.0.0          0           32768 i
*  30.1.0.0/16    10.1.0.14         0           0 104 103 i
* >              10.1.0.10         0           0 103 i
*  40.1.0.0/16    10.1.0.10         0           0 103 104 i
* i               10.1.0.18         0     100     0 104 i
* >              10.1.0.14         0           0 104 i
* i50.1.0.0/16   10.1.0.10         100          0 103 104 105 i
* i               10.1.0.18         100          0 104 105 i
* >              10.1.0.14         0           0 104 105 i

root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
*  20.1.0.0/16    10.1.0.6          0           0 103 102 i
* >              10.1.0.17         0           0 102 i
* >              10.1.0.13         0           0 102 i
* > 30.1.0.0/16   10.1.0.6          0           0 103 i
* >              10.1.0.17         0           0 102 103 i
* >              10.1.0.13         0           0 102 103 i
* i40.1.0.0/16   40.1.0.6          0     100     0 i
* >              0.0.0.0          0           32768 i
*>i50.1.0.0/16  10.1.0.2          0     100     0 105 i

root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
*>i20.1.0.0/16  10.1.0.13         0     100     0 102 i
*>i30.1.0.0/16  10.1.0.6          0     100     0 103 i
* i40.1.0.0/16   40.1.0.1          0     100     0 i
* >              0.0.0.0          0           32768 i
*> 50.1.0.0/16   10.1.0.2          0           0 105 i

root@api-mv:~# lxc-attach -n R05 -- vtysh -c 'show ip bgp'
      Network      Next Hop          Metric LocPrf Weight Path
*  20.1.0.0/16    10.1.0.5          0           0 104 102 i
* >              10.1.0.9          0           0 102 i
*> 30.1.0.0/16   0.0.0.0          0           32768 i
*> 40.1.0.0/16   10.1.0.5          0           0 104 i
* >              10.1.0.9          0           0 102 104 i
*> 50.1.0.0/16   10.1.0.5          0           0 104 105 i
* >              10.1.0.9          0           0 102 104 105 i
```

- a. Quina ruta o rutes tenia R03 per anar al prefix de l'AS 103 abans de connectar el cable? Quina era la best? Quines rutes té després de connectar el cable? Quina és ara la best? Per què? Raoneu la resposta. Busqueu a les captures quins paquets BGP i amb quina informació ha enviat R03 a cadascun del seus veïns BGP per informar del canvi.

Els routers BGP només envien la best de cada prefix als seus veïns. Per tant, si un router té N veïns, com a màxim només podrà aprendre N opcions per anar a cada prefix.

Abans d'establir la sessió entre R05 i R03, R03 té quatre veïns: R01 (10.1.0.17), R02 (10.1.0.13), R07 (40.1.0.2) i R04 (40.1.0.6). R01 i R02 són veïns eBGP de R03. R07 i R04 són veïns iBGP de R03. Recordeu que, en les configuracions BGP que es consideren en aquesta pràctica, és obligatori que tots els routers BGP d'un mateix AS siguin veïns (encara que no estiguin directament connectats) i en aquest escenari tots els routers de la figura són BGP (inclòs R07). Per tant, R03 pot tenir, com a màxim, quatre rutes possibles per arribar al prefix 30.1.0.0/16.

Les rutes que R03 té a la taula BGP per arribar al prefix 30.1.0.0/16 **abans de connectar R03 i R05** són:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 30.1.0.0/16	10.1.0.17	0	102	103	i
*>	10.1.0.13	0	102	103	i

Com s'observa a la taula, R03 té dues rutes possibles per arribar al prefix 30.1.0.0/16: la que té next-hop 10.1.0.17 (R02) i la que té next-hop 10.1.0.13 (R01). Això vol dir (com veurem més endavant) que la best per al prefix 30.1.0.0/16 de R07 i R04 és la que els ha comunicat R02.

Recordeu que per escollir la best cal comparar els atributs de les diferents rutes seguint els criteris següents (de moment només els de color blau):

1. No s'admet una ruta si el router no té cap camí cap al NEXT_HOP del prefix.
2. S'escull la ruta amb **major WEIGHT**. (Criteri local al router).
3. S'escull la ruta amb **major LOCAL_PREFERENCE**. (Criteri global dins un sistema autònom).
4. Es prefereixen les rutes originades de forma local pel router.
5. S'escull la ruta que travessa un menor número d'AS (AS_PATH **més curt**)
6. S'escull la ruta amb un codi ORIGIN **menor**: IGP < EGP < Incomplete.
7. S'escull la ruta amb **menor MED** (Multi-Exit Discriminator).
8. Es prefereix una ruta apresa via eBGP sobre una apresa via iBGP.
9. Es tria la ruta amb menor mètrica (segons el protocol d'encaminament interior) cap al NEXT_HOP.
10. Quan les rutes que es comparen s'han rebut via eBGP, es queda la ruta que ha arribat primer.
11. Es tria la ruta apresa del veí amb menor identificador BGP (que sol ser una de les IPs del router).

Les dues opcions tenen un next-hop vàlid (criteri 1), la mateixa longitud d'AS_PATH (criteri 5), el mateix ORIGIN (criteri 6), són totes dues eBGP (i, per tant, els criteris 8 i 9 no es poden fer servir) però **la que té next-hop 10.1.0.13 s'ha après abans i, per tant, s'escull com a best (criteri 10)**.

Quan es connecta el cable i es configuren R03 i R05 per ser veïns BGP, s'estableix la sessió BGP entre els dos routers (paquets 2 a 5 de la captura **CapturaPart1Exercici4.pcapng** que s'adjunta).

Després d'establir la sessió BGP, els routers s'intercanvien totes les seves best (paquets 6 i 7). Dins el paquet 7, el router R05 envia un Update del prefix 30.1.0.0/16 amb els atributs següents:

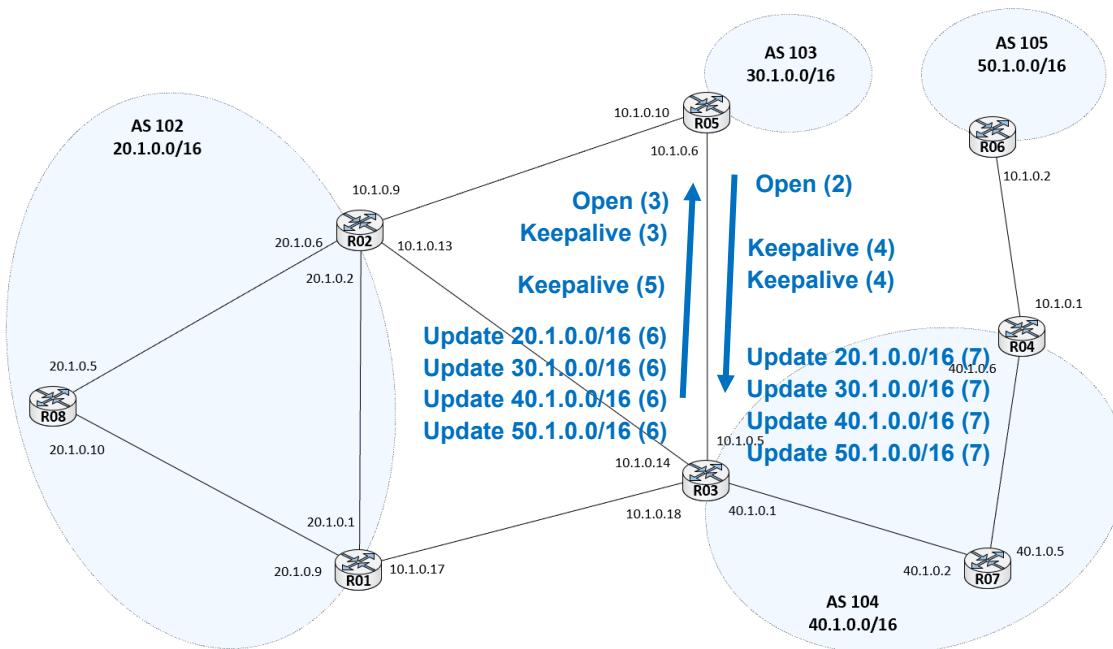
Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 103
NEXT_HOP: 10.1.0.6
MULTI_EXIT_DISC: 0

Quan R03 rep aquest Update, inclou aquesta informació a la seva taula BGP i, com que l'AS_PATH d'aquesta ruta té una longitud menor que la ruta amb next-hop 10.1.0.13 (que és la que tenia R03 com a best per anar al prefix 30.1.0.0/16), R03 canvia la best per anar al prefix 30.1.0.0/16.

Així doncs, les rutes que té R03 a la taula BGP per anar al prefix 30.1.0.0/16 **després de connectar R03 i R05** són:

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 30.1.0.0/16	10.1.0.6	0	0	103 i	
*	10.1.0.17	0	102	103 i	
*	10.1.0.13	0	102	103 i	

El diagrama mostra els missatges que envien R03 i R05 per establir la sessió BGP i per enviar-se les rutes best de la taula un cop establerta la sessió.



Paquet 6: 10.1.0.5 (R03) → 10.1.0.6 (R05)

Prefix: 20.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 102
NEXT_HOP: 10.1.0.5

Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 102 103
NEXT_HOP: 10.1.0.5

Prefix: 40.1.0.0/16
ORIGIN: IGP
AS_PATH: 104
NEXT_HOP: 10.1.0.5
MULTI_EXIT_DISC: 0

Prefix: 50.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 105
NEXT_HOP: 10.1.0.5
MULTI_EXIT_DISC: 0

Paquet 7: 10.1.0.6 (R05) → 10.1.0.5 (R03)

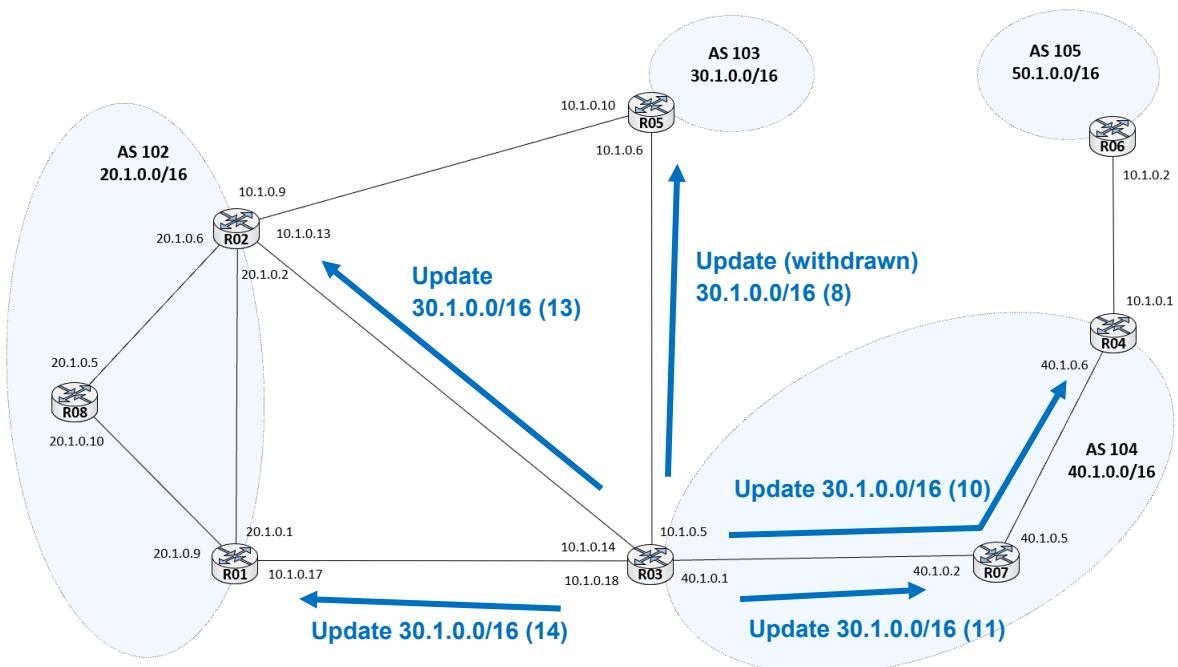
Prefix: 20.1.0.0/16
ORIGIN: IGP
AS_PATH: 103 102
NEXT_HOP: 10.1.0.6

Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 103
NEXT_HOP: 10.1.0.6
MULTI_EXIT_DISC: 0

Prefix: 40.1.0.0/16
ORIGIN: IGP
AS_PATH: 103 102 104
NEXT_HOP: 10.1.0.6

Prefix: 50.1.0.0/16
ORIGIN: IGP
AS_PATH: 103 102 104 105
NEXT_HOP: 10.1.0.6

Com que R03 ha modificat la best per anar al prefix 30.1.0.0/16, ha de comunicar els canvis als seus veïns.



Paquet 14:

10.1.0.18 (R03) → 10.1.0.17 (R01)

Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 103
NEXT_HOP: 10.1.0.18

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip bgp'
      Network      Next Hop     Metric LocPrf Weight Path
Abans:
*>i30.1.0.0/16  10.1.0.10          0    100      0 103 i
Després:
* 30.1.0.0/16  10.1.0.18          0    104      0 103 i
*>i                  10.1.0.10          0    100      0 103 i
```

El paquet 14 conté un Update del prefix 30.1.0.0/16 que R03 envia a R01. Fixeu-vos que, com que R01 és un veí eBGP de R03, quan R03 envia el prefix 30.1.0.0/16 canvia el NEXT_HOP (posa la seva adreça IP = 10.1.0.18) i afegeix el seu número d'AS al principi de l'AS_PATH (AS_PATH = 104 103). R01 instal·la la nova ruta apresa a la taula BGP però, com que no modifica la seva best, no envia cap paquet als seus veïns BGP.

Paquet 13:

10.1.0.14 (R03) → 10.1.0.13 (R02)

Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 103
NEXT_HOP: 10.1.0.14

```
root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp'
      Network      Next Hop     Metric LocPrf Weight Path
Abans:
*> 30.1.0.0/16  10.1.0.10          0    103 i
Després:
* 30.1.0.0/16  10.1.0.14          0    104      0 103 i
*>                  10.1.0.10          0    103 i
```

El paquet 13 conté un Update del prefix 30.1.0.0/16 que R03 envia a R02. Fixeu-vos que, com que R02 és un veí eBGP de R03, quan R03 envia el prefix 30.1.0.0/16 canvia el NEXT_HOP (posa la seva adreça IP = 10.1.0.14) i afegeix el seu número d'AS al principi de l'AS_PATH (AS_PATH = 104 103). R02 instal·la la nova ruta apresa a la taula BGP però, com que no modifica la seva best, no envia cap paquet als seus veïns BGP.

Paquet 10:

40.1.0.1 (R03) → 40.1.0.6 (R04)	<code>root@api-mv:~# lxc-attach -n R04 -- vtysh -c 'show ip bgp'</code>
Prefix: 30.1.0.0/16	Network Next Hop Metric LocPrf Weight Path
ORIGIN: IGP	Abans:
AS_PATH: 103	*>i30.1.0.0/16 10.1.0.13 100 0 102 103 i
NEXT_HOP: 10.1.0.6	Després:
MULTI_EXIT_DISC: 0	*>i30.1.0.0/16 10.1.0.6 0 100 0 103 i
LOCAL_PREF: 100	

El paquet 10 conté un Update del prefix 30.1.0.0/16 que R03 envia a R04. Fixeu-vos que, com que R04 és un veí iBGP de R03, quan R03 envia el prefix 30.1.0.0/16 no modifica el NEXT_HOP ni l'AS_PATH. En canvi, s'afegeix l'atribut LOCAL_PREF que és obligatori enviar en sessions iBGP. Abans de la sessió BGP entre R03 i R05, la best de R04 ja era una ruta apresa via iBGP de R03, però amb NEXT_HOP = 10.1.0.13 i AS_PATH = 102 103. Com que la nova ruta s'aprèn del mateix router que havia anunciat la best anterior, R04 ha de modificar la llista d'atributs de la ruta (sobreescrivir la informació). A més a més, tot i que no es veu a la captura (perquè no s'han capturat els paquets al br05) R04 enviarà un Update a R06 sobre el prefix 30.1.0.0/16. Al veí R07 no li enviarà cap Update perquè les rutes apreses via iBGP no s'envien als veïns iBGP (per això tots els routers BGP del mateix AS han de ser veïns).

Paquet 11:

40.1.0.1 (R03) → 40.1.0.2 (R07)	<code>root@api-mv:~# lxc-attach -n R07 -- vtysh -c 'show ip bgp'</code>
Prefix: 30.1.0.0/16	Network Next Hop Metric LocPrf Weight Path
ORIGIN: IGP	Abans:
AS_PATH: 103	*>i30.1.0.0/16 10.1.0.13 100 0 102 103 i
NEXT_HOP: 10.1.0.6	Després:
MULTI_EXIT_DISC: 0	*>i30.1.0.0/16 10.1.0.6 0 100 0 103 i
LOCAL_PREF: 100	

El paquet 11 conté un Update del prefix 30.1.0.0/16 que R03 envia a R07 via iBGP. Igual que fa R04, el que fa R07 és actualitzar la informació de la ruta del prefix 30.1.0.0/16 perquè la ruta que tenia i la nova ruta les aprèn del mateix veí, R02. Com a conseqüència de l'actualització, R07 no envia cap paquet (perquè només té veïns del mateix AS i les rutes apreses via iBGP no s'anuncien via iBGP).

Paquet 6:

10.1.0.5 (R03) → 10.1.0.6 (R05)

(…)
Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 104 102 103
NEXT_HOP: 10.1.0.5
(…)

Paquet 7:

10.1.0.6 (R05) → 10.1.0.5 (R03)

(…)
Prefix: 30.1.0.0/16
ORIGIN: IGP
AS_PATH: 103
NEXT_HOP: 10.1.0.6
MULTI_EXIT_DISC: 0
(…)

Paquet 8:

10.1.0.5 (R03) → 10.1.0.6 (R05)

Withdrawn prefix: 20.1.0.0/16

El paquet 8 és un Update que R03 envia a R05. Però aquest Update no conté un prefix i la seva llista d'atributs, sinó que és un Withdrawn de la ruta 30.1.0.0/16. Just després d'establir la sessió BGP, R03 i R05 estan obligats a enviar-se totes les rutes best de la seva taula BGP. Al paquet 6 de la captura, R03 envia l'Update sobre el prefix 30.1.0.0/16 amb els atributs que corresponen a la best que té a la seva taula abans de la sessió BGP amb R05. Quan rep el paquet 7, R03 aprèn una ruta millor per arribar al prefix 30.1.0.0/16 i modifica la best d'aquest prefix a la taula. Com que R05 és el seu nou next-hop de la ruta best, R03 li ha de dir que la informació sobre el prefix 30.1.0.0/16 que li havia enviat al paquet 6 ja no és vàlida. Per això li envia l'Update withdrawn d'aquest prefix. En general, els Updates withdrawn d'una ruta s'envien al nou next-hop quan es canvia la best d'un prefix.

- b. Per cada **router** i cada **prefix BGP** analitzeu els canvis que s'hagin produït a la taula BGP. Comproveu si s'han après noves rutes, si s'han modificat les que es tenien o si s'han eliminat rutes i raoneu el per què. Els **routers** que tenen ara més d'una ruta per anar a un prefix, quina han triat com a millor i per què?

Prefixes 40.1.0.0/16 i 50.1.0.0/16:

Després de l'establiment de la sessió BGP entre R03 i R05, R03 envia les seves rutes best a R05 (paquet 6).

Paquet 6: 10.1.0.5 (R03) → 10.1.0.6 (R05)

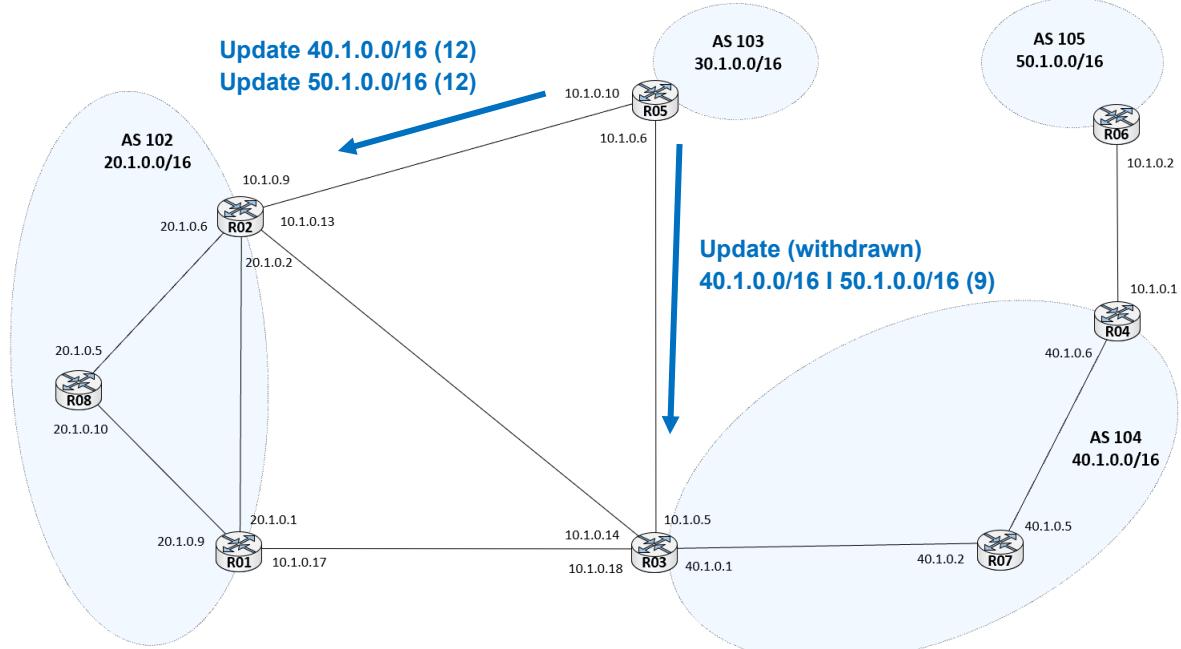
Prefix: 20.1.0.0/16	Prefix: 30.1.0.0/16	Prefix: 40.1.0.0/16	Prefix: 50.1.0.0/16
ORIGIN: IGP	ORIGIN: IGP	ORIGIN: IGP	ORIGIN: IGP
AS_PATH: 104 102	AS_PATH: 104 102 103	AS_PATH: 104	AS_PATH: 104 105
NEXT_HOP: 10.1.0.5	NEXT_HOP: 10.1.0.5	NEXT_HOP: 10.1.0.5	NEXT_HOP: 10.1.0.5
		MULTI_EXIT_DISC: 0	MULTI_EXIT_DISC: 0

```
root@api-mv:~# lxc-attach -n R05 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
Abans:
* > 20.1.0.0/16        10.1.0.9           0          0 102 i
* > 30.1.0.0/16        0.0.0.0           0          32768 i
* > 40.1.0.0/16        10.1.0.9           0          0 102 104 i
* > 50.1.0.0/16        10.1.0.9           0          0 102 104 105 i

Després:
* 20.1.0.0/16          10.1.0.5           0          0 104 102 i
*>                      10.1.0.9           0          0 102 i
* > 30.1.0.0/16        0.0.0.0           0          32768 i
* > 40.1.0.0/16          10.1.0.5           0          0 104 i
*                      10.1.0.9           0          0 102 104 i
* > 50.1.0.0/16          10.1.0.5           0          0 104 105 i
*                      10.1.0.9           0          0 102 104 105 i
```

La ruta al prefix 30.1.0.0/16 que R03 anuncia a R05 al paquet 6, no s'instal·la a la taula de R05 perquè a l'AS_PATH de la ruta hi ha el número d'AS de R05.

R05 canvia la best del prefix 40.1.0.0/16 (la que li anuncia R03 té un AS_PATH més curt). El mateix passa amb el prefix 50.1.0.0/16. Com que R05 modifica la best dels prefixes 40.1.0.0/16 i 50.1.0.0/16, el router R05 ha d'anunciar els canvis als seus veïns:



Paquet 12: 10.1.0.10 (R05) → 10.1.0.9 (R02)	<code>root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp'</code>
Prefix: 40.1.0.0/16	Network Next Hop Metric LocPrf Weight Path
ORIGIN: IGP	Abans:
AS_PATH: 103 104	* i40.1.0.0/16 10.1.0.18 0 100 0 104 i
NEXT_HOP: 10.1.0.10	*> 10.1.0.14 0 104 i
Prefix: 50.1.0.0/16	* i50.1.0.0/16 10.1.0.18 100 0 104 105 i
ORIGIN: IGP	Después:
AS_PATH: 103 104 105	* 40.1.0.0/16 10.1.0.10 0 103 104 i
NEXT_HOP: 10.1.0.10	* i 10.1.0.18 0 100 0 104 i
	*> 10.1.0.14 0 104 i
	* 50.1.0.0/16 10.1.0.10 0 103 104 105 i
	* i 10.1.0.18 100 0 104 105 i
	*> 10.1.0.14 0 104 105 i

El paquet 12 conté un Update del prefix 40.1.0.0/16 i un Update del prefix 50.1.0.0/16 que R05 envia a R02. Fixeu-vos que, com que R02 és un veí eBGP de R05, quan R05 li envia els prefixes, canvia el NEXT_HOP (posa la seva adreça IP = 10.1.0.14) i afegeix el seu número d'AS al principi de l'AS_PATH (AS_PATH = 103 104 pel prefix 40.1.0.0/16 i AS_PATH = 103 104 105 pel prefix 50.1.0.0/16). R02 instal·la les dues rutes apreses a la taula BGP però cap de les dues modifica la best del prefix corresponent i, per tant, R02 no envia cap paquet als seus veïns BGP.

El paquet 9 és un Update que R05 envia a R03 i que conté el withdrawn de les rutes 40.1.0.0/16 i 50.1.0.0/16. Les rutes d'aquests dos prefixes que R05 aprèn de R03 (paquet 6) milloren la best que tenia R05 (paquet 7). Com que el nou next-hop és R03, el router R05 envia a R03 un Update amb el withdrawn d'aquests dos prefixes perquè R03 esborri les opcions que tenia amb next-hop R05 per aquests prefixes. (En aquest escenari, R03 no havia instal·lat aquelles rutes a la seva taula perquè el seu número d'AS apareixia a l'AS_PATH però, en general, quan es rep el withdrawn s'esborra la ruta del prefix indicat que s'havia après del veí que envia el withdrawn).

Paquet 6: 10.1.0.5 (R03) → 10.1.0.6 (R05)	Paquet 7: 10.1.0.6 (R05) → 10.1.0.5 (R03)	Paquet 9: 10.1.0.6 (R05) → 10.1.0.5 (R03)
Prefix: 40.1.0.0/16 ORIGIN: IGP AS_PATH: 104 NEXT_HOP: 10.1.0.5 MULTI_EXIT_DISC: 0	Prefix: 40.1.0.0/16 ORIGIN: IGP AS_PATH: 103 102 104 NEXT_HOP: 10.1.0.6	Withdrawn prefix: 40.1.0.0/16 Withdrawn prefix: 50.1.0.0/16
Prefix: 50.1.0.0/16 ORIGIN: IGP AS_PATH: 104 105 NEXT_HOP: 10.1.0.5 MULTI_EXIT_DISC: 0	Prefix: 50.1.0.0/16 ORIGIN: IGP AS_PATH: 103 102 104 105 NEXT_HOP: 10.1.0.6	

Prefixes 20.1.0.0/16:

Després de l'establiment de la sessió BGP entre R03 i R05, R05 envia les seves rutes best a R03:

Paquet 7: 10.1.0.6 (R05) → 10.1.0.5 (R03)

Prefix: 20.1.0.0/16 ORIGIN: IGP AS_PATH: 103 102 NEXT_HOP: 10.1.0.6	Prefix: 30.1.0.0/16 ORIGIN: IGP AS_PATH: 103 NEXT_HOP: 10.1.0.6	Prefix: 40.1.0.0/16 ORIGIN: IGP AS_PATH: 103 102 104 NEXT_HOP: 10.1.0.6	Prefix: 50.1.0.0/16 ORIGIN: IGP AS_PATH: 103 102 104 105 NEXT_HOP: 10.1.0.6
		MULTI_EXIT_DISC: 0	

```
root@api-mv:~# lxc-attach -n R03 -- vtysh -c 'show ip bgp'
      Network          Next Hop            Metric LocPrf Weight Path
Abans:
* 20.1.0.0/16        10.1.0.17           0          0 102 i
*>                  10.1.0.13           0          0 102 i
* 30.1.0.0/16        10.1.0.17           0          0 102 103 i
*>                  10.1.0.13           0          0 102 103 i
* i40.1.0.0/16       40.1.0.6            0         100 0 i
*>                  0.0.0.0             0          32768 i
*>i50.1.0.0/16       10.1.0.2            0         100 0 105 i

Després:
* 20.1.0.0/16        10.1.0.6           0          0 103 102 i
*                  10.1.0.17           0          0 102 i
*>                  10.1.0.13           0          0 102 i
*> 30.1.0.0/16       10.1.0.6           0          0 103 i
*                  10.1.0.17           0          0 102 103 i
*                  10.1.0.13           0          0 102 103 i
* i40.1.0.0/16       40.1.0.6            0         100 0 i
*>                  0.0.0.0             0          32768 i
*>i50.1.0.0/16       10.1.0.2            0         100 0 105 i
```

La ruta als prefixes 40.1.0.0/16 i 50.1.0.0/16 que R03 anuncia a R03 no s'instal·la a la taula de R03 perquè hi ha el número de l'AS de R03 (104) a l'AS_PATH de la ruta anunciada. R03 canvia la best del prefix 40.1.0.0/16 (la que li anuncia R03 té un AS_PATH més curt). El mateix passa amb el prefix 50.1.0.0/16.

La ruta que aprèn per al prefix 20.1.0.0/16 no millora la best que ja té a la taula. S'instal·la a la taula BGP però no s'envia cap paquet als veïns.

El cas del prefix 30.1.0.0/16 és el que s'ha analitzat a l'apartat a) d'aquest mateix exercici.

- c. Analitzeu amb el Wireshark els paquets capturats als diferents *routers*. Quins paquets i amb quina informació s'ha enviat per propagar noves rutes? Quins paquets i amb quina informació s'han enviat per eliminar rutes?

Els missatges Update s'utilitzen tant per propagar noves rutes com per esborrar rutes que ja no són vàlides.

Quan es vol anunciar un prefix (o conjunt de prefixes) s'indica el prefix (o prefixes) al camp *Network Layer Reachability Information* i la llista d'atributs associada al camp *Path attributes* del paquet.

Quan es vol esborrar un prefix (o conjunt de prefixes) s'indica el prefix (o prefixes) al camp *Withdrawn Routes*.

Atureu les captures.

Des d'un terminal del **PC** utilitzeu l'eina `tracepath_api` per saber per on passen els paquets que s'envien entre els terminals que hi ha a l'interior dels sistemes autònoms de l'escenari. Per exemple, per saber per on passen els paquets des del PC05 al PC01 (amb adreça IP 20.1.0.98) la comanda és:

```
tracepath_api PC05 20.1.0.98
```

- d. Comenteu si la sortida de la comanda `tracepath_api` coincideix amb el que observeu a les taules d'encaminament dels routers.

L'script `tracepath_api` fa ús de les comandes `ip route list match` i `ip route get` per representar els salts que fa un paquet que s'envia des d'un PC o router de l'escenari a una determinada adreça IP.

```
root@api-mv:~# tracepath_api 20.1.0.98
PC05  Matching FIB entries:
          40.1.0.129      eth0
Next Hop: 40.1.0.129

R04  Matching FIB entries:
          20.1.0.0/16      40.1.0.5      eth0
Next Hop: 40.1.0.5

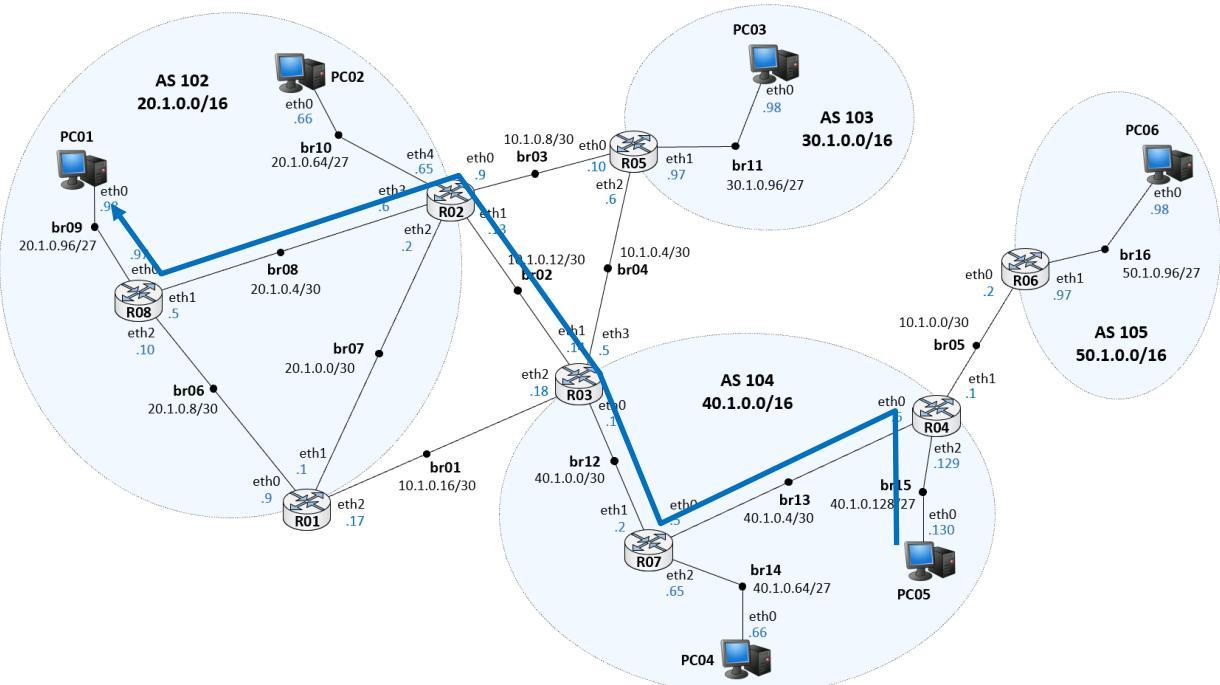
R07  Matching FIB entries:
          20.1.0.0/16      40.1.0.1      eth1
Next Hop: 40.1.0.1

R03  Matching FIB entries:
          20.1.0.0/16      10.1.0.13     eth1
Next Hop: 10.1.0.13

R02  Matching FIB entries:
          20.1.0.96/27     20.1.0.5      eth3
Next Hop: 20.1.0.5

R08  Matching FIB entries:
          20.1.0.0/16      0.0.0.0      directly connected
          20.1.0.96/27     0.0.0.0      directly connected
Next Hop: 0.0.0.0

*** PATH SUMMARY ***
PC05 --> R04(40.1.0.129) --> R07(40.1.0.5) --> R03(40.1.0.1) -->
R02(10.1.0.13) --> R08(20.1.0.5) --> 20.1.0.98 (Destination Address)
```



Atureu l'escenari de la primera part amb la comanda: P05-E01-stop

PART II. ATRIBUTS BGP

Escenari P05-E02

Executeu els scripts que s'indiquen a continuació per tal d'arrancar els contenidors i configurar les adreces IP dels *routers* i PCs de l'escenari.

En un terminal del **PC**, executeu la comanda: P05-E02-start-zebra

Triga una mica. Espereu a que acabi d'executar-se l'script i surti el prompt del terminal.

Executeu els scripts que s'indiquen a continuació per tal d'activar el dimoni del protocol d'encaminament interior que s'utilitza en cada *router* de l'escenari.

En un terminal del **PC**, executeu la comanda: P05-E02-start-int

Exercici 1. Configuració de l'escenari

En aquest escenari les adreces IP de les interfícies dels *routers* i dels PCs ja estan configurades, així com el protocol d'encaminament interior que s'utilitza en cada sistema autònom, tal i com teniu representat a la figura **Escenari part II** del final de la pràctica.

Comproveu la taula d'encaminament del PC03: lxc-attach -n PC03 -- vtysh -c 'show ip route'

Al PC03 hi ha configurada una ruta per defecte (0.0.0.0/0) amb next-hop 20.1.0.129 (que és l'adreça IP de R09 a la xarxa 20.1.0.128/27). Tots els paquets amb una adreça IP destí que no sigui de la xarxa 20.1.0.128/27, el PC03 els enviarà a R09.

```
lxc-attach -n PC03 -- vtysh -c 'show ip route'
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
       > - selected route, * - FIB route

S>* 0.0.0.0/0 [1/0] via 20.1.0.129, eth0
C>* 20.1.0.128/27 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
```

La resta de PCs de l'escenari també estan configurats amb una única ruta estàtica al prefix 0.0.0.0/0 (ruta per defecte) en la que el next-hop és l'adreça IP del router que tenen directament connectat.

Comproveu també les taules d'encaminament dels *routers* de l'AS 102.

```
root@api-mv:~# lxc-attach -n R01 -- vtysh -c 'show ip route'
O>* 10.1.0.8/30 [110/30] via 20.1.0.5, eth0, 00:00:21
O>* 10.1.0.12/30 [110/30] via 20.1.0.5, eth0, 00:00:21
C>* 10.1.0.16/30 is directly connected, eth1
O>* 10.1.0.20/30 [110/30] via 20.1.0.5, eth0, 00:00:26
O>* 20.1.0.0/30 [110/20] via 20.1.0.5, eth0, 00:00:31
C>* 20.1.0.4/30 is directly connected, eth0
O>* 20.1.0.8/30 [110/20] via 20.1.0.5, eth0, 00:00:31
O>* 20.1.0.128/27 [110/20] via 20.1.0.5, eth0, 00:00:31
O>* 20.1.0.192/27 [110/30] via 20.1.0.5, eth0, 00:00:21
```

```

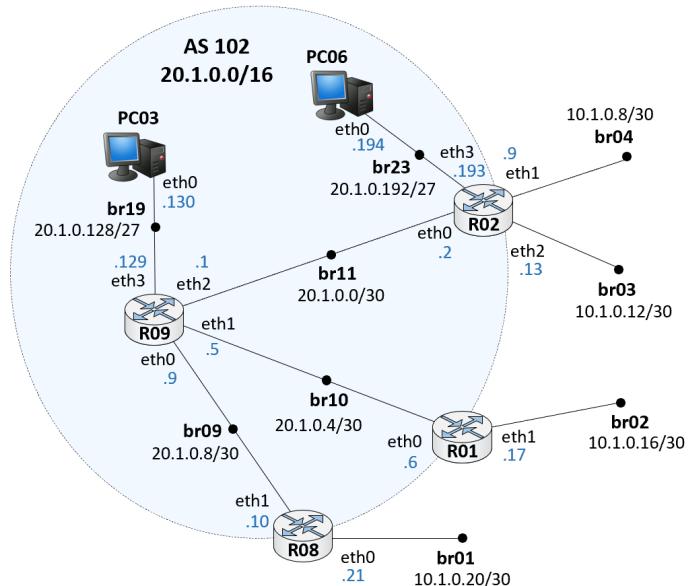
root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip route'
C>* 10.1.0.8/30 is directly connected, eth1
C>* 10.1.0.12/30 is directly connected, eth2
O>* 10.1.0.16/30 [110/30] via 20.1.0.1, eth0, 00:00:21
O>* 10.1.0.20/30 [110/30] via 20.1.0.1, eth0, 00:00:21
C>* 20.1.0.0/30 is directly connected, eth0
O>* 20.1.0.4/30 [110/20] via 20.1.0.1, eth0, 00:00:21
O>* 20.1.0.8/30 [110/20] via 20.1.0.1, eth0, 00:00:21
O>* 20.1.0.128/27 [110/20] via 20.1.0.1, eth0, 00:00:21
C>* 20.1.0.192/27 is directly connected, eth3

root@api-mv:~# lxc-attach -n R08 -- vtysh -c 'show ip route'
O>* 10.1.0.8/30 [110/30] via 20.1.0.9, eth1, 00:00:21
O>* 10.1.0.12/30 [110/30] via 20.1.0.9, eth1, 00:00:21
O>* 10.1.0.16/30 [110/30] via 20.1.0.9, eth1, 00:00:26
C>* 10.1.0.20/30 is directly connected, eth0
O>* 20.1.0.0/30 [110/20] via 20.1.0.9, eth1, 00:00:31
O>* 20.1.0.4/30 [110/20] via 20.1.0.9, eth1, 00:00:31
C>* 20.1.0.8/30 is directly connected, eth1
O>* 20.1.0.128/27 [110/20] via 20.1.0.9, eth1, 00:00:31
O>* 20.1.0.192/27 [110/30] via 20.1.0.9, eth1, 00:00:21

root@api-mv:~# lxc-attach -n R09 -- vtysh -c 'show ip route'
O>* 10.1.0.8/30 [110/20] via 20.1.0.2, eth2, 00:00:21
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth2, 00:00:21
O>* 10.1.0.16/30 [110/20] via 20.1.0.6, eth1, 00:00:26
O>* 10.1.0.20/30 [110/20] via 20.1.0.10, eth0, 00:00:26
C>* 20.1.0.0/30 is directly connected, eth2
C>* 20.1.0.4/30 is directly connected, eth1
C>* 20.1.0.8/30 is directly connected, eth0
C>* 20.1.0.128/27 is directly connected, eth3
O>* 20.1.0.192/27 [110/20] via 20.1.0.2, eth2, 00:00:21

```

A l'AS 102 s'utilitza OSPF com a protocol d'encaminament interior. A la taula d'encaminament dels quatre routers d'aquest AS hi ha una ruta per a cadascuna de les 9 xarxes que es veuen a la figura (les internes de l'AS 102 i les xarxes que connecten l'AS 102 amb els AS veïns).



- a. Amb quins PCs de l'escenari us podeu comunicar des del PC03? Raoneu la resposta i utilitzeu l'eina `tracepath_api` per verificar el camí que segueixen els paquet des del PC03 al PC o PCs als que pot arribar.

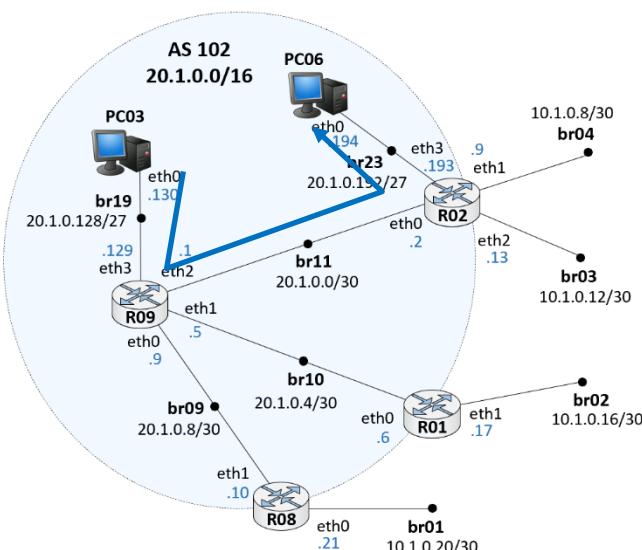
De moment només hi ha connectivitat entre els PCs de l'AS 102 (entre PC03 i PC06). Quan s'intenta enviar un paquet des del PC03 a qualsevol PC de fora de l'AS 102, el paquet arriba fins a R09 i és R09 el que no té cap ruta a la taula d'encaminament que li serveixi per enviar el paquet (per això surt el missatge *Network is unreachable*).

Amb la comanda `tracepath_api` es pot comprovar que hi ha connectivitat entre PC03 i PC06:

PC03 → PC06

```
root@api-mv:~# tracepath_api PC03 20.1.0.194
PC03  Matching FIB entries:
          20.1.0.129      eth0
          Next Hop: 20.1.0.129
R09   Matching FIB entries:
          20.1.0.192/27    20.1.0.2      eth2
          Next Hop: 20.1.0.2
R02   Matching FIB entries:
          20.1.0.192/27    0.0.0.0      directly connected
          Next Hop: 0.0.0.0

*** PATH SUMMARY ***
PC03 --> R09(20.1.0.129) --> R02(20.1.0.2) --> 20.1.0.194 (Destination Address)
```



Amb la comanda `tracepath_api` es pot comprovar que no hi ha connectivitat entre PC03 i els PCs de fora de l'AS 102

PC03 → PC01

```
root@api-mv:~# tracepath_api PC03 30.1.0.130
PC03  Matching FIB entries:
          20.1.0.129      eth0
          Next Hop: 20.1.0.129
R09   Network is unreachable
```

PC03 → PC05

```
root@api-mv:~# tracepath_api PC03 40.1.0.130
PC03  Matching FIB entries:
                  20.1.0.129          eth0
      Next Hop: 20.1.0.129
R09    Network is unreachable
```

PC03 → PC02

```
root@api-mv:~# tracepath_api PC03 50.1.0.130
PC03  Matching FIB entries:
                  20.1.0.129          eth0
      Next Hop: 20.1.0.129
R09    Network is unreachable
```

PC03 → PC04

```
root@api-mv:~# tracepath_api PC03 60.1.0.130
PC03  Matching FIB entries:
                  20.1.0.129          eth0
      Next Hop: 20.1.0.129
R09    Network is unreachable
```

Executeu els scripts que s'indiquen a continuació per tal d'activar el dimoni del protocol d'encaminament exterior (*bgpd*) a tots els *routers* de l'escenari.

En un terminal del **PC**, executeu la comanda: P05-E02-start-bgpd

Comproveu la configuració dels *routers* R02, R08 i R09 amb la comanda:

```
lxc-attach -n nom_router -- vtysh -c 'show run'
```

```
root@api-mv:~# lxc-attach -n R02 -- vtysh
R02# sh run
( . . . )
router bgp 102
  bgp router-id 10.1.0.9
  network 20.1.0.0/16
  neighbor 10.1.0.10 remote-as 103
  neighbor 10.1.0.14 remote-as 104
  neighbor 20.1.0.1 remote-as 102
  neighbor 20.1.0.6 remote-as 102
  neighbor 20.1.0.10 remote-as 102
( . . . )
```

```

root@api-mv:~# lxc-attach -n R08 -- vtysh

R08# sh run
(...)
router bgp 102
bgp router-id 10.1.0.21
network 20.1.0.0/16
neighbor 10.1.0.22 remote-as 106
neighbor 20.1.0.2 remote-as 102
neighbor 20.1.0.6 remote-as 102
neighbor 20.1.0.9 remote-as 102
(...)

root@api-mv:~# lxc-attach -n R09 -- vtysh

R09# sh run
(...)
router bgp 102
bgp router-id 20.1.0.1
neighbor 20.1.0.2 remote-as 102
neighbor 20.1.0.6 remote-as 102
neighbor 20.1.0.10 remote-as 102
(...)

```

b. Quants veïns BGP té cadascun d'aquests routers? Quins són i per què?

Com es pot comprovar mirant la seva configuració, a R02 s'han configurat 5 veïns, a R08 se n'han configurat 4 i a R09 se n'han configurat 3. A continuació teniu el detall de l'estat actual dels veïns de cadascun d'aquests routers:

```

root@api-mv:~# lxc-attach -n R02 -- vtysh -c 'show ip bgp summary'

BGP router identifier 10.1.0.9, local AS number 102
      Neighbor     V   AS MsgRcvd MsgSent TblVer  InQ OutQ Up/Down State/PfxRcd
10.1.0.10      4   103      6      10        0      0      0 00:01:25          3
10.1.0.14      4   104      6      9        0      0      0 00:01:26          3
20.1.0.1       4   102      3      10        0      0      0 00:01:25          0
20.1.0.6       4   102      0      0        0      0      0 never           Active
20.1.0.10      4   102      7      9        0      0      0 00:01:25          3

Total number of neighbors 5
Total num. Established sessions 4
Total num. of routes received      9

```

R02 és de l'AS 102 i té 5 peers configurats:

- 10.1.0.10 (R05) és de l'AS 103 i la sessió eBGP està en estat *Established*
- 10.1.0.14 (R03) és de l'AS 104 i la sessió eBGP està en estat *Established*
- 20.1.0.1 (R09) és de l'AS 102 i la sessió iBGP està en estat *Established*
- 20.1.0.6 (R01) és de l'AS 102 i la sessió iBGP està en estat *Active* (R01 encara no està configurat i no està enviant el missatge OPEN)
- 20.1.0.10 (R08) és de l'AS 102 i la sessió iBGP està en estat *Established*

```
root@api-mv:~# lxc-attach -n R08 -- vtysh -c 'show ip bgp summary'
```

BGP router identifier 10.1.0.21, local AS number 102

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
10.1.0.22	4	106	8	8	0	0	0	00:01:25	4
20.1.0.2	4	102	7	9	0	0	0	00:01:25	4
20.1.0.6	4	102	0	0	0	0	0	never	Active
20.1.0.9	4	102	3	9	0	0	0	00:01:25	0

Total number of neighbors 4
Total num. Established sessions 3
Total num. of routes received 8

R08 és de l'AS 102 i té 4 peers configurats:

- 10.1.0.22 (R07) és de l'AS 106 i la sessió eBGP està en estat *Established*
- 20.1.0.2 (R02) és de l'AS 102 i la sessió iBGP està en estat *Established*
- 20.1.0.6 (R01) és de l'AS 102 i la sessió iBGP està en estat *Active* (R01 encara no està configurat i no està enviant el missatge OPEN)
- 20.1.0.9 (R08) és de l'AS 102 i la sessió iBGP està en estat *Established*

```
root@api-mv:~# lxc-attach -n R09 -- vtysh -c 'show ip bgp summary'
```

BGP router identifier 20.1.0.1, local AS number 102

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
20.1.0.2	4	102	8	5	0	0	0	00:01:25	4
20.1.0.6	4	102	0	0	0	0	0	never	Active
20.1.0.10	4	102	7	5	0	0	0	00:01:25	3

Total number of neighbors 3

Total num. Established sessions 2

Total num. of routes received 7

R09 és de l'AS 102 i té 3 peers configurats:

- 20.1.0.2 (R02) és de l'AS 102 i la sessió iBGP està en estat *Established*
- 20.1.0.6 (R01) és de l'AS 102 i la sessió iBGP està en estat *Active* (R01 encara no està configurat i no està enviant el missatge OPEN)
- 20.1.0.10 (R08) és de l'AS 102 i la sessió iBGP està en estat *Established*

Tenint en compte la configuració de R02, R08 i R09, configureu el *router* R01 per tal que anunciï el prefix del seu sistema autònom (20.1.0.0/16) amb els atributs per defecte.

```
root@api-mv:~# lxc-attach -n R01 -- vtysh

R01# conf term
R01(config)# router bgp 102
R01(config-router)# bgp router-id 10.1.0.17
R01(config-router)# network 20.1.0.0/16
R01(config-router)# neighbor 10.1.0.18 remote-as 104
R01(config-router)# neighbor 20.1.0.2 remote-as 102
R01(config-router)# neighbor 20.1.0.5 remote-as 102
R01(config-router)# neighbor 20.1.0.10 remote-as 102
R01(config-router)# exot
R01(config)# exit
R01# sh run
(...)
router bgp 102
  bgp router-id 10.1.0.17
  network 20.1.0.0/16
  neighbor 10.1.0.18 remote-as 104
  neighbor 20.1.0.2 remote-as 102
  neighbor 20.1.0.5 remote-as 102
  neighbor 20.1.0.10 remote-as 102
(...)
```

→ R01 és un router de l'AS 102
 → S'assigna l'ID BGP 10.1.0.17
 → Es configura R01 per anunciar el prefix 20.1.0.0/16
 → Es defineix R03 de l'AS 104 com a veí eBGP
 → Es defineix R02 de l'AS 102 com a veí iBGP
 → Es defineix R09 de l'AS 102 com a veí iBGP
 → Es defineix R08 de l'AS 102 com a veí iBGP

Verifiqueu que s'estableixen les sessions BGP de R01 amb tots els seus veïns amb la comanda:

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp summary'
```

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp summary'
BGP router identifier 10.1.0.17, local AS number 102
RIB entries 9, using 1008 bytes of memory
Peers 4, using 36 KiB of memory
Neighbor      V   AS MsgRcvd MsgSent    TblVer  InQ OutQ Up/Down  State/PfxRcd
10.1.0.18      4   104     8      9        0       0     0  0 00:00:10      3
20.1.0.2       4   102     7      7        0       0     0  0 00:00:10      4
20.1.0.5       4   102     3      7        0       0     0  0 00:00:11      0
20.1.0.10      4   102     6      7        0       0     0  0 00:00:11      3
Total number of neighbors 4
Total num. Established sessions 4
Total num. of routes received      10
```

A R01 s'ha configurat 4 peers:

- 10.1.0.18 (R03) és de l'AS 104 i la sessió eBGP està en estat *Established*
- 20.1.0.2 (R02) és de l'AS 102 i la sessió iBGP està en estat *Established*
- 20.1.0.5 (R09) és de l'AS 102 i la sessió iBGP està en estat *Established*
- 20.1.0.10 (R08) és de l'AS 102 i la sessió iBGP està en estat *Established*

Executant la comanda 'show ip bgp summary' a la resta de routers de l'AS 102, es pot comprovar que ara tots tenen totes les sessions BGP en estat *Established*.

Exercici 2. Atribut MED

L'objectiu d'aquest exercici és entendre el funcionament de l'atribut Multi-Exit-Discriminator (MED o metric). El MED serveix per suggerir al sistema autònom veí per on es prefereix que es realitzi l'accés a l'AS. Per tant, afecta al trànsit entrant (inbound) a l'AS.

Per defecte, quan un router anuncia el prefix del seu propi AS, ho fa amb un valor de MED = 0 a tots els seus veïns (eBGP i iBGP). Si es rep via eBGP un prefix amb el MED a la llista d'atributs i s'escull aquesta ruta com la best al prefix, el valor de l'atribut MED es propaga als veïns iBGP però no als veïns eBGP.

En aquest exercici haureu d'observar les rutes cap al prefix 20.1.0.0/16 que té R03 i haureu de modificar el punt d'entrada a l'AS 102 que escull R03 per arribar al prefix 20.1.0.0/16 modificant l'atribut MED.

Abans de resoldre les preguntes de l'exercici, fixeu-vos com s'ha propagat el prefix 20.1.0.0/16 als diferents routers de l'escenari. A l'escenari d'aquesta segona part, tots els routers ja tenien la configuració BGP preparada excepte R01, que l'heu hagut de configurar a l'exercici 1. El que teniu a la figura següent és el resum dels paquets UPDATE amb el prefix 20.1.0.0/16 que s'han enviat els routers quan heu engegat el dimoni bgpd (sense R01) i, a la pàgina següent, teniu els paquets UPDATE amb el prefix 20.1.0.0/16 que s'envien quan configureu R01. Entre parèntesi teniu el número del paquet a la captura **CapturaPart2Exercici2Intro.pcapng** (utilitzeu aquest número per saber l'ordre cronològic d'enviament dels paquets).

Update 20.1.0.0/16 (4 i 9)

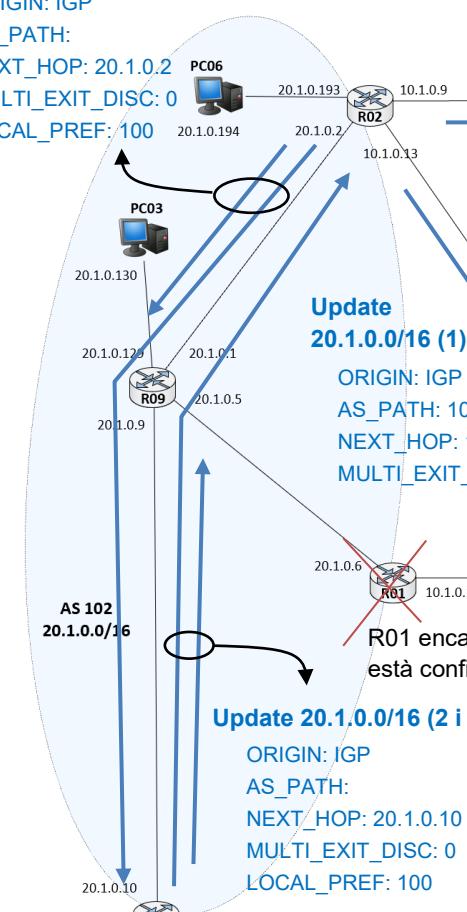
ORIGIN: IGP

AS_PATH:

NEXT_HOP: 20.1.0.2

MULTI_EXIT_DISC: 0

LOCAL_PREF: 100



Update 20.1.0.0/16 (7)

ORIGIN: IGP

AS_PATH: 102

NEXT_HOP: 10.1.0.9

MULTI_EXIT_DISC: 0

LOCAL_PREF: 100

Update 20.1.0.0/16 (13)

ORIGIN: IGP

AS_PATH: 103 102

NEXT_HOP: 10.1.0.6

Update 20.1.0.0/16 (5)

ORIGIN: IGP

AS_PATH: 104 102

NEXT_HOP: 10.1.0.5

AS 104
40.1.0.0/16

Update 20.1.0.0/16 (6)

ORIGIN: IGP

AS_PATH: 102

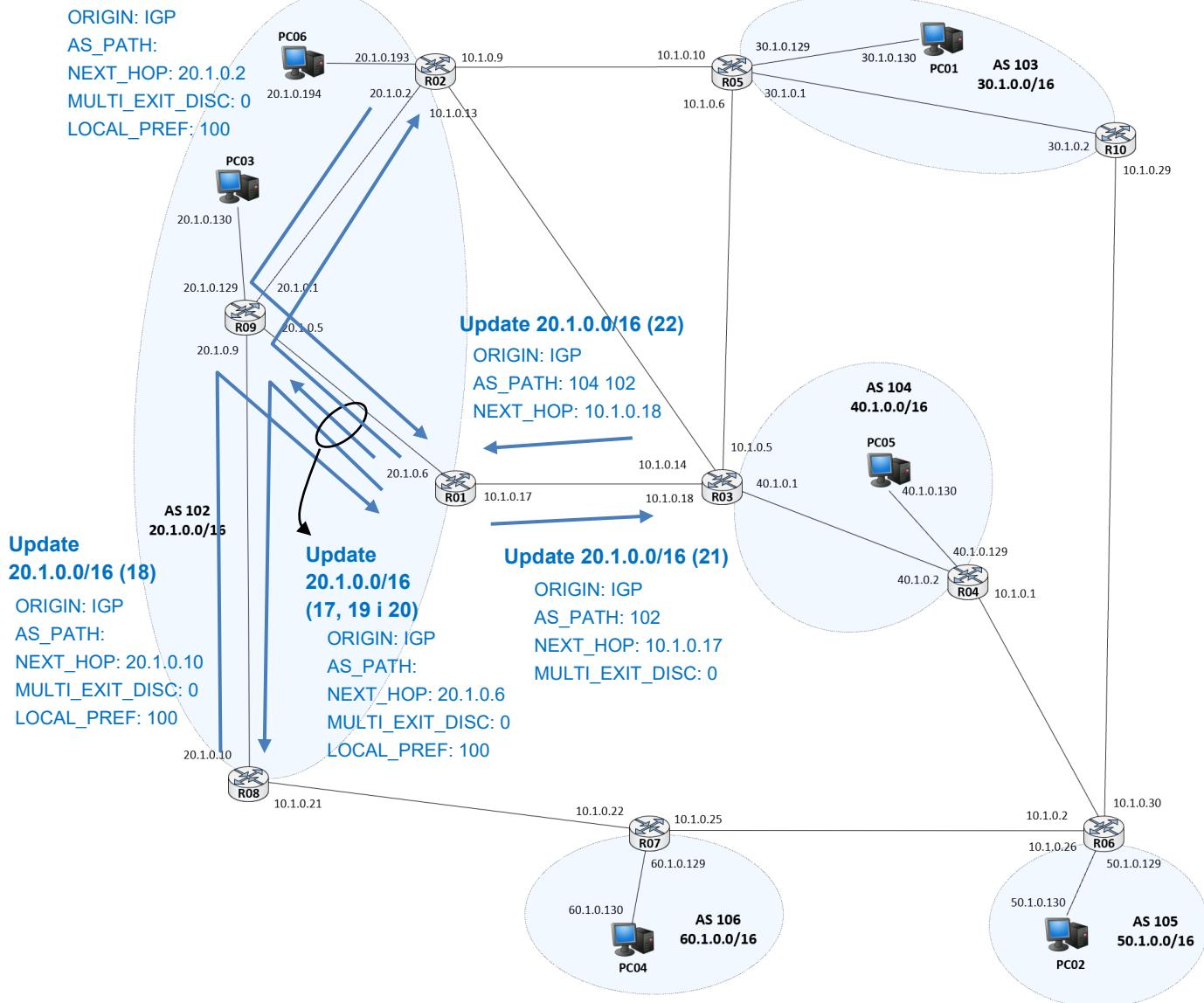
NEXT_HOP: 10.1.0.13

MULTI_EXIT_DISC: 0

LOCAL_PREF: 100

AS 104
40.1.0.0/16

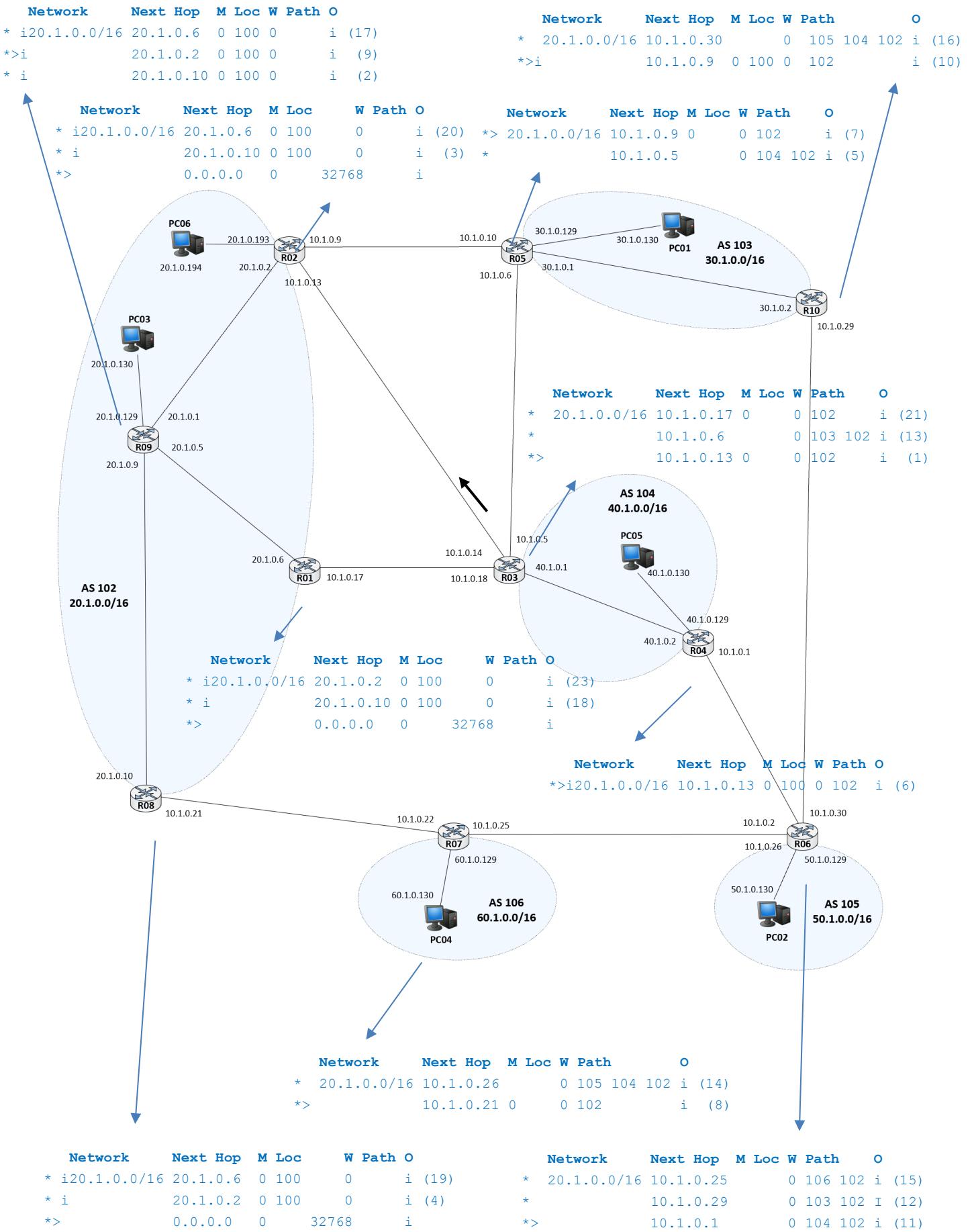
AS 104
40.1.0.0/16</

Update 20.1.0.0/16 (23)

Com s'observa a les figures, els UPDATE que envien R02, R08 i més tard R01 a tots els seus veïns (eBGP i iBGP) anunciant el prefix 20.1.0.0/16 del seu AS tenen l'atribut MULTI_EXIT_DISC (MED) = 0. De moment té el valor per defecte perquè encara no s'ha configurat cap filtre per modificar-lo.

La primera ruta al prefix 20.1.0.0/16 que rep R02 és la que li envia R01 (paquet 1). R03 instal·la la ruta a la taula BGP i, com que no en té cap més, la selecciona com a best i la propaga a tots els seus veïns excepte R02. A R04, que és del seu mateix AS (peer iBGP), li envia l'UPDATE sense modificar els atributs ORIGIN, AS_PATH, NEXT_HOP i MED i, a més a més, afegeix l'atribut LOCAL_PREFERENCE (que en aquest cas té el valor per defecte, 100) (paquet 6). A R05, que és d'un AS diferent (peer eBGP), li envia l'UPDATE sense modificar l'atribut ORIGIN, afegint el seu ASN a l'AS_PATH i modificant el NEXT_HOP (paquet 5). R04 instal·la la ruta al prefix 20.1.0.0/16 que rep de R03 a la taula BGP i, com que és la best (és l'única que té), la propaga al seu veí R06. Com que R06 és un veí eBGP, R04 envia el prefix 20.1.0.0/16 amb l'atribut ORIGIN sense modificar, afegint el seu ASN a l'AS_PATH i modificant el NEXT_HOP (però no propaga el MED a la llista d'atributs) (paquet 11).

A l'esquema de la pàgina següent teniu les rutes al prefix 20.1.0.0/16 que té cada router a la taula BGP com a conseqüència dels UPDATES rebuts dels diagrames anteriors (en aquest exemple, cada paquet es correspon a una ruta de l'entrada de la taula BGP d'un router excepte el paquet 22 perquè hi ha l'AS de R01 a l'AS_PATH i, per tant, no és una ruta vàlida). Al final de cada ruta teniu el número del paquet UPDATE de la captura que li ha permès al router aprendre la ruta. Fixeu-vos que, com més temps fa que s'ha après la ruta, més avall és a la taula BGP.



Activeu el Wireshark per capturar els paquets als bridges br03, br02, br08, br12 i br05.

La captura de paquets la teniu al fitxer **CapturaPart2Exercici2.pcapng**

Observeu la taula BGP actual de R03: `lxc-attach -n R03 -- vtysh -c 'show ip bgp'`

lxc-attach -n R03 -- vtysh -c 'show ip bgp'

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.17	0		0	102 i
*	10.1.0.6			0	103 102 i
*>	10.1.0.13	0		0	102 i
* 30.1.0.0/16	10.1.0.17			0	102 103 i
*	10.1.0.13			0	102 103 i
*>	10.1.0.6	0		0	103 i
* i40.1.0.0/16	40.1.0.2	0	100	0	i
*>	0.0.0.0	0		32768	i
* 50.1.0.0/16	10.1.0.6			0	103 105 i
*>i	10.1.0.2	0	100	0	105 i
* 60.1.0.0/16	10.1.0.17			0	102 106 i
*	10.1.0.6			0	103 102 106 i
* i	10.1.0.2		100	0	105 106 i
*>	10.1.0.13			0	102 106 i

a. Quantes rutes apareixen per al prefix 20.1.0.0/16? Quina està seleccionada com a best? Per què?

R03 ha après tres rutes pel prefix 20.1.0.0/16. Recordeu que, si no s'utilitza l'opció de multipath bgp, els routers trien una única best cap a cada prefix i és la que propaguen als seus veïns. Per tant, el nombre màxim de rutes que un router pot aprendre cap a un prefix és igual al nombre de veïns BGP del router. En aquest cas, R03 té quatre veïns i, per tant, com a màxim podria tenir quatre rutes per anar al prefix 20.1.0.0/16 a la taula BGP. Si en té menys de quatre, significa que algun dels seus veïns l'està fent servir com a next-hop (en aquest cas, si ho mireu al diagrama de la pàgina anterior, R04 fa servir la ruta rebuda de R03 per anar al prefix 20.1.0.0/16).

De les tres rutes que ha après per anar al prefix 20.1.0.0/16, R03 ha escollit la que té com a next-hop 10.1.0.13.

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.17	0		0	102 i
*	10.1.0.6			0	103 102 i
*>	10.1.0.13	0		0	102 i

Criteri 1. No s'admet una ruta si el router no té cap camí cap al NEXT_HOP del prefix

Les tres rutes tenen el símbol * al principi, la qual cosa significa que són rutes vàlides perquè el router sap arribar a l'adreça IP de la columna Next Hop.

Criteri 2. S'escull la ruta amb major pes (WEIGHT)

Les tres rutes tenen el mateix valor de WEIGHT.

Criteri 3. S'escull la ruta amb major LOCAL_PREFERENCE

Les tres rutes s'han après via eBGP i, com que l'atribut LOCAL_PREFERENCE no es propaga en sessions eBGP (i encara no s'ha configurat cap filtre que modifiqui aquest atribut), les tres tenen la columna LocPrf buida. És important tenir en compte que una ruta que no té cap valor al camp LOCAL_PREFERENCE es tracta com si tingués el valor per defecte, que és 100.

Criteri 4. Es prefereixen les rutes originades de forma local pel router

Cap de les tres rutes l'ha generat el router (a través de, per exemple, la comanda `network` o per redistribució d'un altre protocol) per tant aquest criteri no serveix en aquest cas per escollir la millor.

Criteri 5. S'escull la ruta que travessa un menor número d'AS (AS PATH més curt)

La segona ruta té un AS_PATH més llarg que les altres dues i, per tant, queda descartada com a possible best pel prefix 20.1.0.0/16

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.17	0		0	102 i
*	10.1.0.6			0	103 102 i
*>	10.1.0.13	0		0	102 i

Criteri 6. S'escull la ruta amb un codi ORIGIN menor: IGP (i) < EGP (e) < Incomplete (?)

Les dues que queden tenen el mateix ORIGIN (recordeu que aquest atribut és la darrera lletra que apareix a cada ruta i que en aquesta pràctica sempre serà “i”).

Criteri 7. S'escull la ruta amb **menor MED** (Multi-Exit Discriminator)

Les dues rutes tenen el mateix valor a la columna Metric (atribut MED).

Criteri 8. Es prefereix una ruta apresa via eBGP sobre una apresa via iBGP

Les dues rutes són eBGP.

Criteri 9. Es tria la ruta amb menor mètrica (segons el protocol d'encaminament interior) cap al NEXT HOP

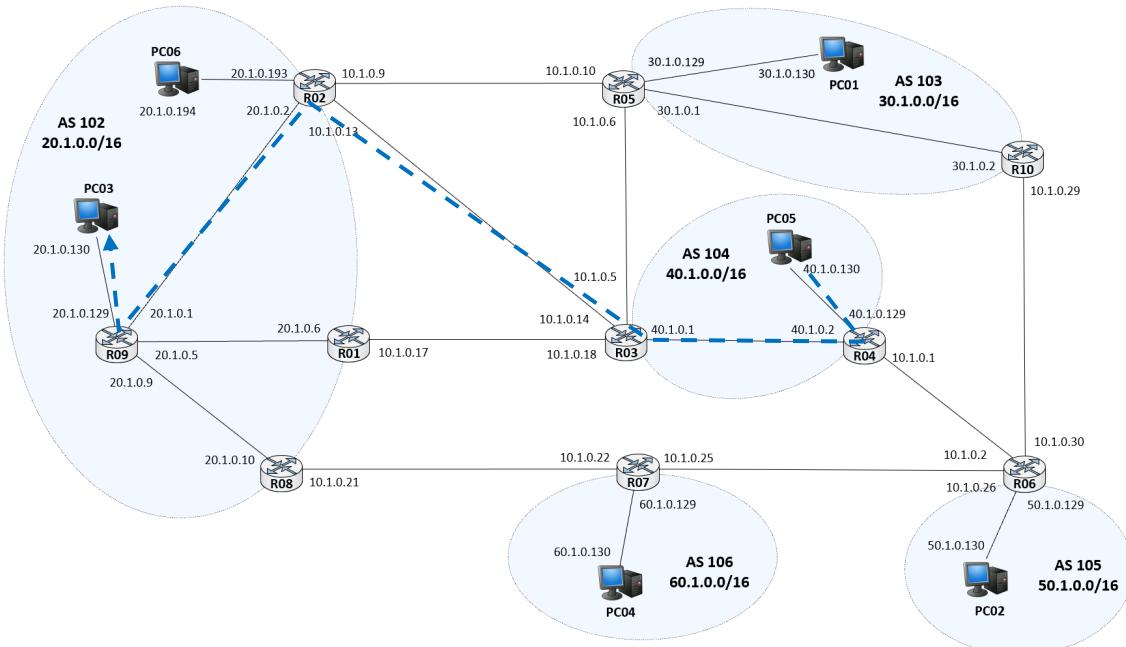
Com que les dues són eBGP, no es pot aplicar el criteri de la menor mètrica al next-hop.

Criteri 10. Quan les rutes que es comparen s'han rebut via eBGP, es queda la ruta que ha arribat primer

Les dues són eBGP i la primera que ha arribat és la que està més avall a la taula, és a dir, la que té next-hop 10.1.0.13. Aquesta és la que s'escull com a ruta best cap al prefix 20.1.0.0/16 i es marca amb el símbol > al principi de la ruta.

Des d'un terminal del **PC** utilitzeu l'eina `tracepath_api` per verificar per on passen els paquets que s'envien des del PC05 al PC03: `tracepath api PC05 20.1.0.130`

b. Apunteu-vos el camí que segueixen els paquets per anar del PC05 al PC03.



```

tracepath_api PC05 20.1.0.130
PC05  Matching FIB entries:
        40.1.0.129      eth0
    Next Hop: 40.1.0.129

R04  Matching FIB entries:
    20.1.0.0/16      40.1.0.1      eth0
    Next Hop: 40.1.0.1

R03  Matching FIB entries:
    20.1.0.0/16      10.1.0.13     eth2
    Next Hop: 10.1.0.13

R02  Matching FIB entries:
    20.1.0.128/27    20.1.0.1      eth0
    Next Hop: 20.1.0.1

R09  Matching FIB entries:
    20.1.0.0/16      0.0.0.0      directly connected
    20.1.0.128/27    0.0.0.0      directly connected
    Next Hop: 0.0.0.0

*** PATH SUMMARY ***
PC05 --> R04(40.1.0.129) --> R03(40.1.0.1) --> R02(10.1.0.13) --> R09(20.1.0.1) --
> 20.1.0.130 (Destination Address)

```

Comproveu també les taules d'encaminament BGP de la resta de *routers*, excepte els del sistema autònom 102, i apunteu-vos les rutes que tenen per arribar al prefix 20.1.0.0/16.

(Mireu el diagrama de la pàgina 66)

R03:

	Network	Next Hop	Metric	LocPrf	Weight	Path
*	20.1.0.0/16	10.1.0.17	0		0	102 i
*		10.1.0.6			0	103 102 i
*>		10.1.0.13	0		0	102 i

Criteri 10 → Entre les dues amb AS_PATH 102, la primera que ha arribat

R04:

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>i	20.1.0.0/16	10.1.0.13	0	100	0	102 i

Només té una ruta pel prefix 20.1.0.0/16

R05:

	Network	Next Hop	Metric	LocPrf	Weight	Path
*>	20.1.0.0/16	10.1.0.9	0		0	102 i
*		10.1.0.5			0	104 102 i

Criteri 5 → AS_PATH més curt

R10:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.30			0	105 104 102 i
*>i	10.1.0.9	0	100		0 102 i

Criteri 5 → AS_PATH més curt

R07:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.26			0	105 104 102 i
*>	10.1.0.21	0			0 102 i

Criteri 5 → AS_PATH més curt

R06:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 20.1.0.0/16	10.1.0.25			0	106 102 i
*	10.1.0.29			0	103 102 i
*>	10.1.0.1				0 104 102 i

Criteri 10 → La primera que ha arribat

R01:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i20.1.0.0/16	20.1.0.2	0	100	0	i
* i	20.1.0.10	0	100	0	i
*>	0.0.0.0	0			32768 i

Criteri 2 → Major weight

R02:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i20.1.0.0/16	20.1.0.6	0	100	0	i
* i	20.1.0.10	0	100	0	i
*>	0.0.0.0	0			32768 i

Criteri 2 → Major weight

R08:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i20.1.0.0/16	20.1.0.6	0	100	0	i
* i	20.1.0.2	0	100	0	i
*>	0.0.0.0	0			32768 i

Criteri 2 → Major weight

R09:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i20.1.0.0/16	20.1.0.6	0	100	0	i
*>i	20.1.0.2	0	100	0	i
* i	20.1.0.10	0	100	0	i

Criteri 11 → Les tres rutes empaten en tot i s'escull la ruta apresa del router amb ID menor

- ID BGP de 20.1.0.6 (R01) és 10.1.0.17
- ID BGP de 20.1.0.2 (R02) és **10.1.0.9**
- ID BGP de 20.1.0.10 (R08) és 10.1.0.21

(L'ID dels routers el podeu veure amb la comanda 'show run' o bé 'show ip bgp summary' a cada router)

Configureu els routers R01 i R02 per tal de suggerir a R03, modificant l'atribut MED, que canviï la seva millor ruta (best) per anar al prefix 20.1.0.0/16. És a dir, si ara està fent servir com a nexthop l'adreça 10.1.0.13, forceu a que esculli la 10.1.0.17. Si per contra està fent servir la 10.1.0.17, configureu el MED per a que esculli la 10.1.0.13.

Per fer-ho, acobleu un terminal a R01, un altre a R02 i un altre a R03. Entreu al vtysh i seguiu els passos següents:

- Configureu el valor de l'atribut MED que voleu que R01 i R02 anuncien per al prefix 20.1.0.0/16 al veí R03

```
R01# configure terminal
R01(config)# access-list P20 permit 20.1.0.0/16
R01(config)# route-map RMAP1 permit 10
R01(config-route-map)# match ip address P20
R01(config-route-map)# set metric _____ ← escolliu un valor
R01(config-route-map)# exit
R01(config)# route-map RMAP1 permit 20
R01(config-route-map)# exit
R01(config)# router bgp 102
R01(config-router)# neighbor 10.1.0.18 route-map RMAP1 out

R02# configure terminal
R02(config)# access-list P20 permit 20.1.0.0/16
R02(config)# route-map RMAP1 permit 10
R02(config-route-map)# match ip address P20
R02(config-route-map)# set metric _____ ← escolliu un valor
R02(config-route-map)# exit
R02(config)# route-map RMAP1 permit 20
R02(config-route-map)# exit
R02(config)# router bgp 102
R02(config-router)# neighbor 10.1.0.14 route-map RMAP1 out
```

- Configureu R03 perquè tingui en compte el MED de routers del mateix sistema autònom a l'hora d'escollir una ruta i forceu l'enviament de rutes (soft reset) de R01 i R02 cap a R03:

```
R03# configure terminal
R03(config)# router bgp 104
R03(config-router)# bgp deterministic-med
R03(config-router)# exit
R03(config)# exit
R03# clear ip bgp 10.1.0.13 in
R03# clear ip bgp 10.1.0.17 in
```

Nota: És imprescindible que entengueu les comandes que acabeu de configurar i la seva utilitat. (Consulteu la informació de les pàgines 17 i 18 per entendre les comandes i el seu significat).

Hi ha diferents maneres de configurar els routers BGP per tal que tinguin en compte l'atribut MED. En aquesta pràctica **només es considerarà el cas bgp-deterministic-med activat i always-compare-med desactivat**. Això significa que un router només compararà l'atribut MED de dues rutes en el cas que aquestes dues rutes s'hagin après del mateix AS. És a dir, només es compararà el MED si les dues rutes tenen el mateix ASN al principi de l'AS_PATH.

En la configuració actual, R03 ha escollit la ruta amb nexthop 10.1.0.13 per anar al prefix 20.1.0.0/16:

```
lxc-attach -n R03 -- vtysh -c 'show ip bgp'

      Network          Next Hop        Metric LocPrf Weight Path
*  20.1.0.0/16        10.1.0.17         0        0 102  i
*              10.1.0.6
*>            10.1.0.13         0        0 102  i
```

Fixeu-vos que, les rutes que ha après R03 dels veïns 10.1.0.17 (R01) i 10.1.0.13 (R02) provenen del mateix AS i tenen el valor MED per defecte (que és 0). Com que les dues rutes empiten en tot abans del criteri 8 (el del MED), es podria influenciar en la selecció de la ruta de R03 cap al prefix 20.1.0.0/16 fent que el veí 10.1.0.17 (R01) envii a R03 pel prefix 20.1.0.0/16 un valor de MED més petit del que envia el veí 10.1.0.13 (R02).

Per exemple:

```
R01# configure terminal
R01(config)# access-list P20 permit 20.1.0.0/16
R01(config)# route-map RMAP1 permit 10
R01(config-route-map)# match ip address P20
R01(config-route-map)# set metric 2
R01(config-route-map)# exit
R01(config)# route-map RMAP1 permit 20
R01(config-route-map)# exit
R01(config)# router bgp 102
R01(config-router)# neighbor 10.1.0.18 route-map RMAP1 out

R02# configure terminal
R02(config)# access-list P20 permit 20.1.0.0/16
R02(config)# route-map RMAP1 permit 10
R02(config-route-map)# match ip address P20
R02(config-route-map)# set metric 4
R02(config-route-map)# exit
R02(config)# route-map RMAP1 permit 20
R02(config-route-map)# exit
R02(config)# router bgp 102
R02(config-router)# neighbor 10.1.0.14 route-map RMAP1 out
```

En aquesta pràctica s'utilitzen route-maps per modificar els valors dels atributs associats a un determinat prefix.

Un route-map és una construcció condicional (similar a un esquema `if-then (-else)`) que permet avaluar certes condicions de concordança (`match`) i (opcionalment) modificar un o més valors. Cada route-map s'identifica amb un nom i pot contenir més d'una clàusula de concordança. Cada clàusula (entrada) té assignat un número de seqüència.

Començant per la primera entrada del route-map (la que té un número de seqüència més petit), s'avaluen les condicions de l'entrada del route-map.

- Si hi ha concordança (`match`), s'executen les accions que hi hagi especificades a l'entrada i (si no hi ha la comanda `Continue`) es surt del route-map acceptant (`permit`) o eliminant (`deny`) el paquet.
- Si no hi ha concordança (`match`), es salta a la següent entrada del route-map i es comprova si hi ha "match".
- Si s'acaben les entrades del route-map sense que hi hagi concordança amb les condicions de cap de les entrades, es descarta el paquet (*implicit deny*).

Per definir les condicions de concordança de les entrades del route-map, en aquesta pràctica es fan servir access-lists. En general, les llistes d'accés permeten fer filtrat de paquets i controlar el flux de paquets a la xarxa, però en el nostre cas serviran simplement per indicar a quins prefixes volem modificar els atributs.

Fixeu-vos, per exemple, en la configuració de R01 quan introduiu les comandes de l'enunciat:

```
R01# show run
( ... )
router bgp 102
bgp router-id 10.1.0.17
network 20.1.0.0/16
neighbor 10.1.0.18 remote-as 104
neighbor 10.1.0.18 route-map RMAP1 out      ----> [6]
neighbor 20.1.0.2 remote-as 102
neighbor 20.1.0.5 remote-as 102
neighbor 20.1.0.10 remote-as 102
( ... )
access-list P20 permit 20.1.0.0/16          ----> [1]
!
route-map RMAP1 permit 10                  ----> [2]
  match ip address P20                      ----> [3]
  set metric 2                            ----> [4]
!
route-map RMAP1 permit 20                  ----> [5]
( ... )
```

Es configura una llista d'accés (access-list) que es diu P20 associada al prefix 20.1.0.0/16 [1]

Es configura un route-map que es diu RMAP1 i que té dues entrades.

- La primera entrada del route-map RMAP1 [2] té número de seqüència 10 i la condició de concordança o match [3] és que el prefix que s'avalua sigui (o estigui contingut dins) el prefix definit per l'access-list P20 (és a dir, el prefix 20.1.0.0/16).
 - Si es compleix aquesta condició, s'assigna el valor 2 a l'atribut metric (MED) [4] i es surt del filtre acceptant el paquet (*permit*).
 - Si no es compleix la condició de l'entrada, es salta a l'entrada següent del route-map.
- La segona entrada del route-map RMAP1 [5] té número de seqüència 20 i està buida (no té cap condició de concordança (match) ni defineix cap acció). Qualsevol paquet que arribi a aquesta condició, s'accepta (*permit*). El propòsit d'aquesta entrada és evitar que cap prefix arribi al final del route-map i se li apliqui el *deny* per defecte.

Finalment s'indica que el route-map s'ha d'aplicar sobre els prefixes BGP que s'enviïn (out) al veí 10.1.0.18 [6]

A més a més de configurar R01 i R02, cal assegurar que R03 té l'opció **`bgp-deterministic-med`** activada per poder tenir en compte el valor del MED quan arribi al criteri 8 de la selecció de ruta. Per fer-ho, s'executen les comandes següents:

```
R03# configure terminal
R03(config)# router bgp 104
R03(config-router)# bgp deterministic-med
R03(config-router)# exit
R03(config)# exit
( . . . )
!
router bgp 104
  bgp router-id 10.1.0.18
  bgp deterministic-med
  network 40.1.0.0/16
  neighbor 10.1.0.6 remote-as 103
  neighbor 10.1.0.13 remote-as 102
  neighbor 10.1.0.17 remote-as 102
  neighbor 40.1.0.2 remote-as 104
!
( . . . )
```

Un cop els peers BGP estan en estat *Established* només s'envien rutes quan hi ha canvis a la taula BGP. Això vol dir que, per poder aplicar la política configurada, cal forçar d'alguna manera l'enviament de les rutes de R01 a R03 i de R02 a R03. Per tal de no haver de fer un hard reset de la connexió (que implicaria tancar la sessió BGP i tornar-la a establir) es va definir el missatge ROUTE-REFRESH (RFC 2918). Aquest missatge permet demanar a un veí que enviï les best de la seva taula BGP sense haver d'interrompre la sessió BGP.

Per tal que R03 sol·liciti l'enviament de les best a R01 i R02, executeu les comandes:

```
R03# clear ip bgp 10.1.0.13 in
R03# clear ip bgp 10.1.0.17 in
```

Observeu la taula BGP de R03: `show ip bgp`

- c. Quantes rutes apareixen per al prefix 20.1.0.0/16? Quina està seleccionada com a *best*? Per què?

<code>lxc-attach -n R03 -- vtysh -c 'show ip bgp'</code>					
Network	Next Hop	Metric	LocPrf	Weight	Path
*> 20.1.0.0/16	10.1.0.17	2	0	102	i
*	10.1.0.6		0	103	102 i
*	10.1.0.13	4	0	102	i

Hi ha tres rutes pel prefix 20.1.0.0/16 i s'escull la que té next-hop 10.1.0.17 (R01). La segona ruta es descarta com a best perquè té l'AS_PATH més llarg que les altres dues. La primera i l'última empaten en tots els criteris fins arribar al criteri 8. R03 té configurada l'opció deterministic-med i, com que les dues rutes s'han anunciat per veïns del mateix AS, compara el MED de les dues i escull la primera perquè té un valor de MED més petit.

Atureu les captures del Wireshark i observeu-les.

- d. Quin missatge envia R03 a R02 i a R01 per aplicar el canvi de política en la selecció de rutes?

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.1.0.14	10.1.0.13	BGP	91	ROUTE-REFRESH Message
2	0.000962189	10.1.0.18	10.1.0.17	BGP	91	ROUTE-REFRESH Message

> Frame 1: 91 bytes on wire (728 bits), 91 bytes captured (728 bits) on interface an
 > Linux cooked capture
 > Internet Protocol Version 4, Src: 10.1.0.14, Dst: 10.1.0.13
 > Transmission Control Protocol, Src Port: 44228, Dst Port: 179, Seq: 431817312, Ack
 > Border Gateway Protocol - ROUTE-REFRESH Message
 Marker: ffffffffffffffffffffff
 Length: 23
 Type: ROUTE-REFRESH Message (5)
 Address family identifier (AFI): IPv4 (1)
 Subtype: Normal route refresh request [RFC2918] with/without ORF [RFC5291] (0)
 Subsequent address family identifier (SAFI): Unicast (1)

El missatge que R03 envia a R01 i a R02 és un ROUTE-REFRESH, per sol·licitar que aquests routers li tornin a enviar les best de la seva taula.

- e. Propaga R03 l'atribut MED del prefix 20.1.0.0/16 al seu veí R05 (eBGP)? i al seu veí R04 (iBGP)?

A R05 (eBGP) no i a R04 (iBGP) sí. Mireu els diagrames de la pàgina següent.

- f. Propaga R04 l'atribut MED del prefix 20.1.0.0/16 al seu veí R06?

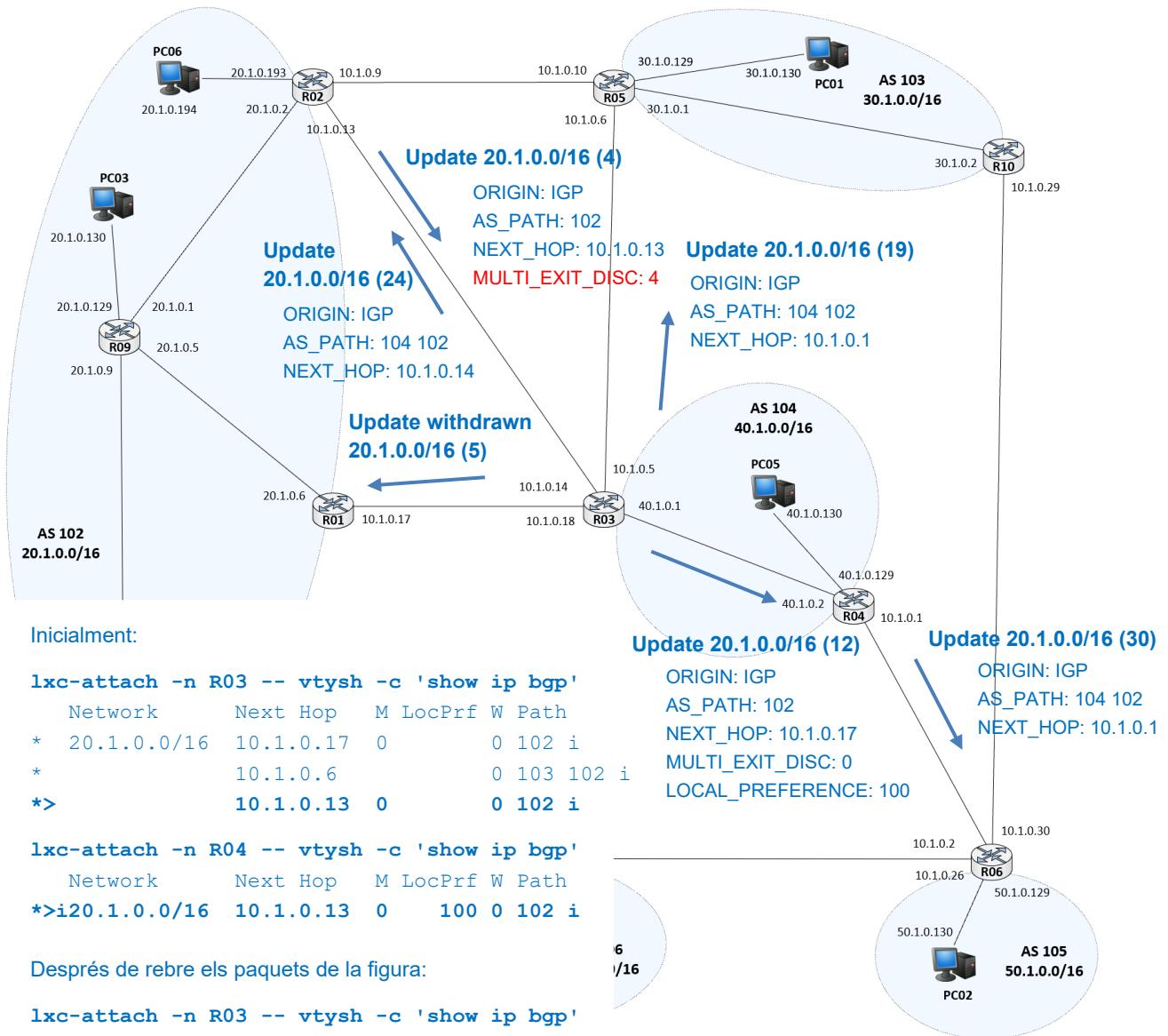
No. Mireu els diagrames de la pàgina següent.

Comproveu també les taules BGP de la resta de routers, excepte els de l'AS 102, i fixeu-vos amb el prefix 20.1.0.0/16.

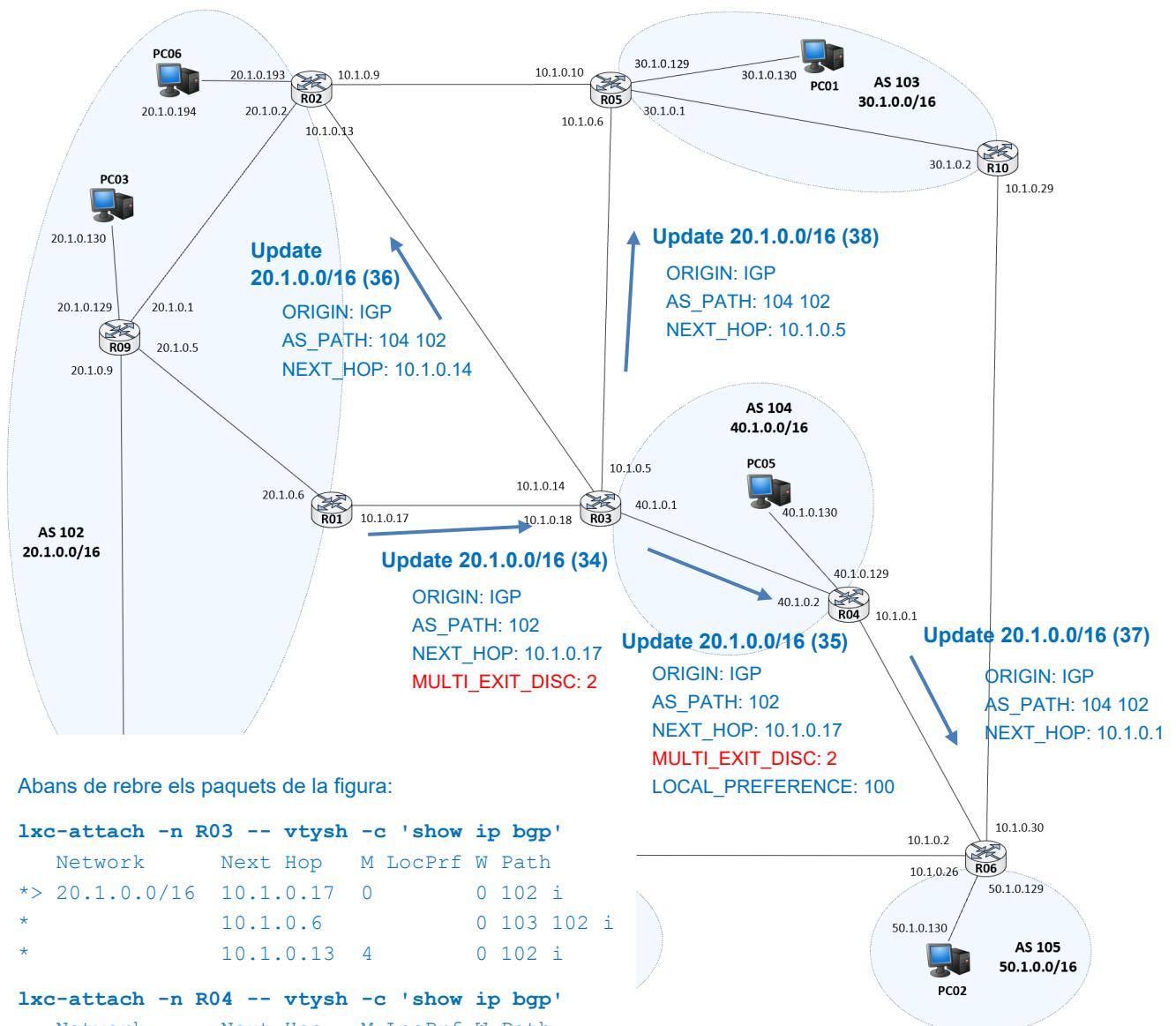
- g. Hi ha algun canvi en algun router sobre aquest prefix?

Només canvia R04. Mireu els diagrames de la pàgina següent.

A la captura **CapturaPart2Exercici2.pcapng** hi ha els missatges UPDATE que s'envien quan R03 envia el missatge ROUTE-REFRESH a R02 (paquet 1) i a R01 (paquet 2). El primer router que envia les best de la seva taula a R03 és R02 i quan envia l'Update del prefix 20.1.0.0/16 ho fa amb l'atribut MED = 4 (paquet 4). R03 instal·la aquesta informació a la seva taula BGP i, entre les tres opcions que té R03 pel prefix 20.1.0.0/16, tria la ruta amb nexthop 10.1.0.17 (recordeu que està configurat per tenir en compte l'atribut MED si les rutes que competeixen provenen del mateix AS). Com que canvia la best, R03 ha de propagar l'actualització als seus veïns. R01 és ara el seu nou nexthop i per això R03 li envia un Update amb un withdrawn del prefix 20.1.0.0/16 (paquet 5). A la resta de veïns, els ha d'enviar l'Update amb el prefix i la llista d'atributs. A R04 (veí iBGP) li envia el prefix amb l'atribut MED = 0 (que és el que té la best de la taula BGP de R03) (paquet 12). A R05 i a R02 (veïns eBGP) els hi envia el prefix sense l'atribut MED (paquets 19 i 24, respectivament). R04, quan rep el paquet 12 de R03, sobreescriu la informació de la seva best pel prefix 20.1.0.0/16 i envia un Update al seu veí R06, sense l'atribut MED (paquet 30).



Quan R01 rep el ROUTE-REFRESH de R03, li envia les best de la seva taula. R01 envia l'Update del prefix 20.1.0.0/16 amb MED = 2 (paquet 34). Quan R03 rep aquest paquet, sobreescriu la ruta best de la seva taula BGP i, com que hi ha hagut canvis en la best de R03, propaga la informació a tots els seus veïns, excepte R01. A R04 (veí iBGP) li envia l'Update del prefix 20.1.0.0/16 amb el MED = 2 (paquet 35). A R02 i R05 (veïns eBGP) els hi envia sense el MED (paquets 36 i 38, respectivament). R04 sobreescrivirà la informació de la seva best al prefix 20.1.0.0/16 i propaga la informació (sense el MED) al seu veí R06 (paquet 37).



Abans de rebre els paquets de la figura:

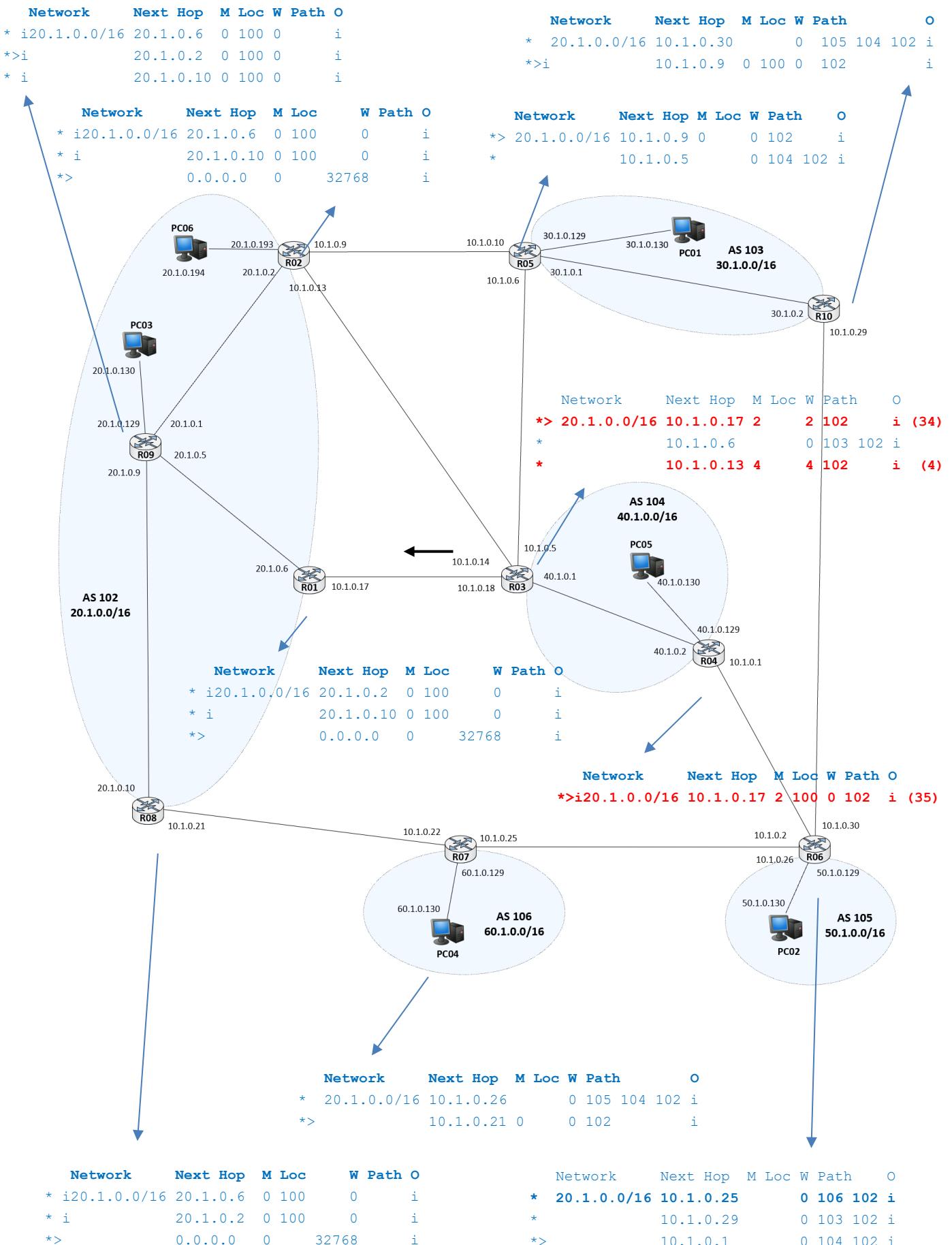
```
lxc-attach -n R03 -- vtysh -c 'show ip bgp'
  Network      Next Hop     M LocPrf W Path
*> 20.1.0.0/16 10.1.0.17  0      0 102 i
*          10.1.0.6       0 103 102 i
*          10.1.0.13      4      0 102 i

lxc-attach -n R04 -- vtysh -c 'show ip bgp'
  Network      Next Hop     M LocPrf W Path
*>i20.1.0.0/16 10.1.0.17  0      100 0 102 i
```

Després de rebre els paquets de la figura:

```
lxc-attach -n R03 -- vtysh -c 'show ip bgp'
  Network      Next Hop     M LocPrf W Path
*> 20.1.0.0/16 10.1.0.17  2      0 102 i      (paquet 34)
*          10.1.0.6       0 103 102 i
*          10.1.0.13      4      0 102 i

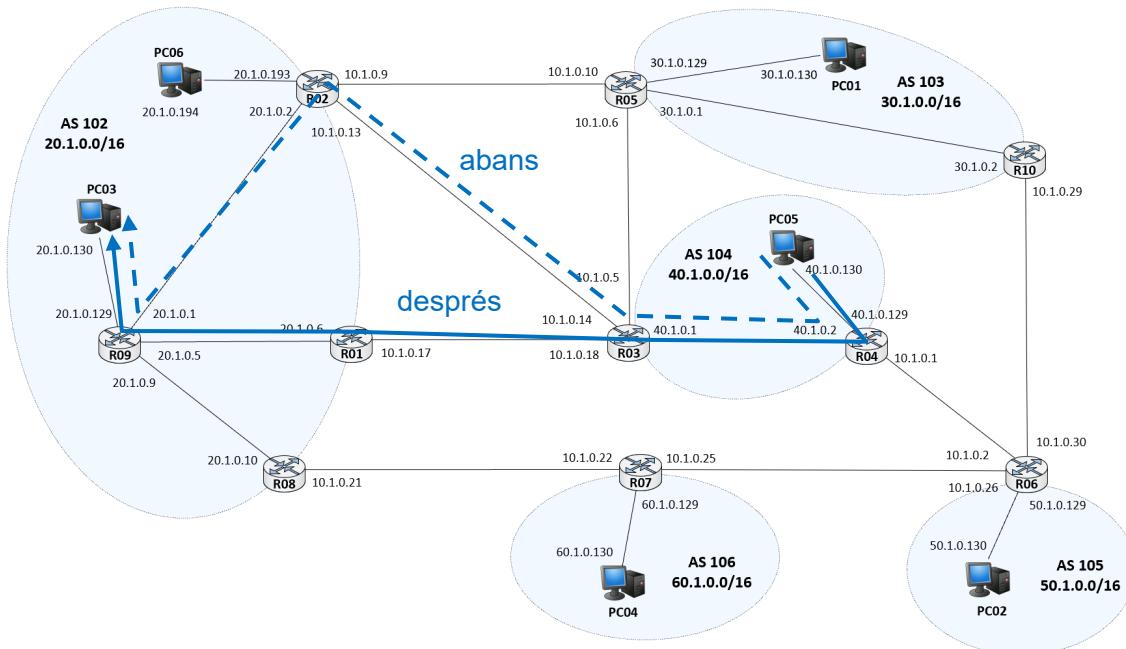
lxc-attach -n R04 -- vtysh -c 'show ip bgp'
  Network      Next Hop     M LocPrf W Path
*>i20.1.0.0/16 10.1.0.17  2      100 0 102 i      (paquet 35)
```



Des d'un terminal del **PC** utilitzeu l'eina `tracepath_api` per saber per on passen els paquets que s'envien des del PC05 al PC03: `tracepath_api PC05 20.1.0.130`

h. Compareu el resultat amb el que heu obtingut a l'apartat b.

Els paquets ara entren a l'AS 102 per R01 (abans entraven per R02).



`tracepath_api PC05 20.1.0.130`

```

PC05  Matching FIB entries:
          40.1.0.129      eth0
    Next Hop: 40.1.0.129

R04  Matching FIB entries:
          20.1.0.0/16      40.1.0.1      eth0
    Next Hop: 40.1.0.1

R03  Matching FIB entries:
          20.1.0.0/16      10.1.0.17     eth2
    Next Hop: 10.1.0.17

R01  Matching FIB entries:
          20.1.0.128/27    20.1.0.5      eth0
    Next Hop: 20.1.0.5

R09  Matching FIB entries:
          20.1.0.0/16      0.0.0.0      directly connected
          20.1.0.128/27    0.0.0.0      directly connected
    Next Hop: 0.0.0.0

*** PATH SUMMARY ***
PC05 --> R04(40.1.0.129) --> R03(40.1.0.1) --> R01(10.1.0.17) --> R09(20.1.0.5) --
> 20.1.0.130 (Destination Address)

```

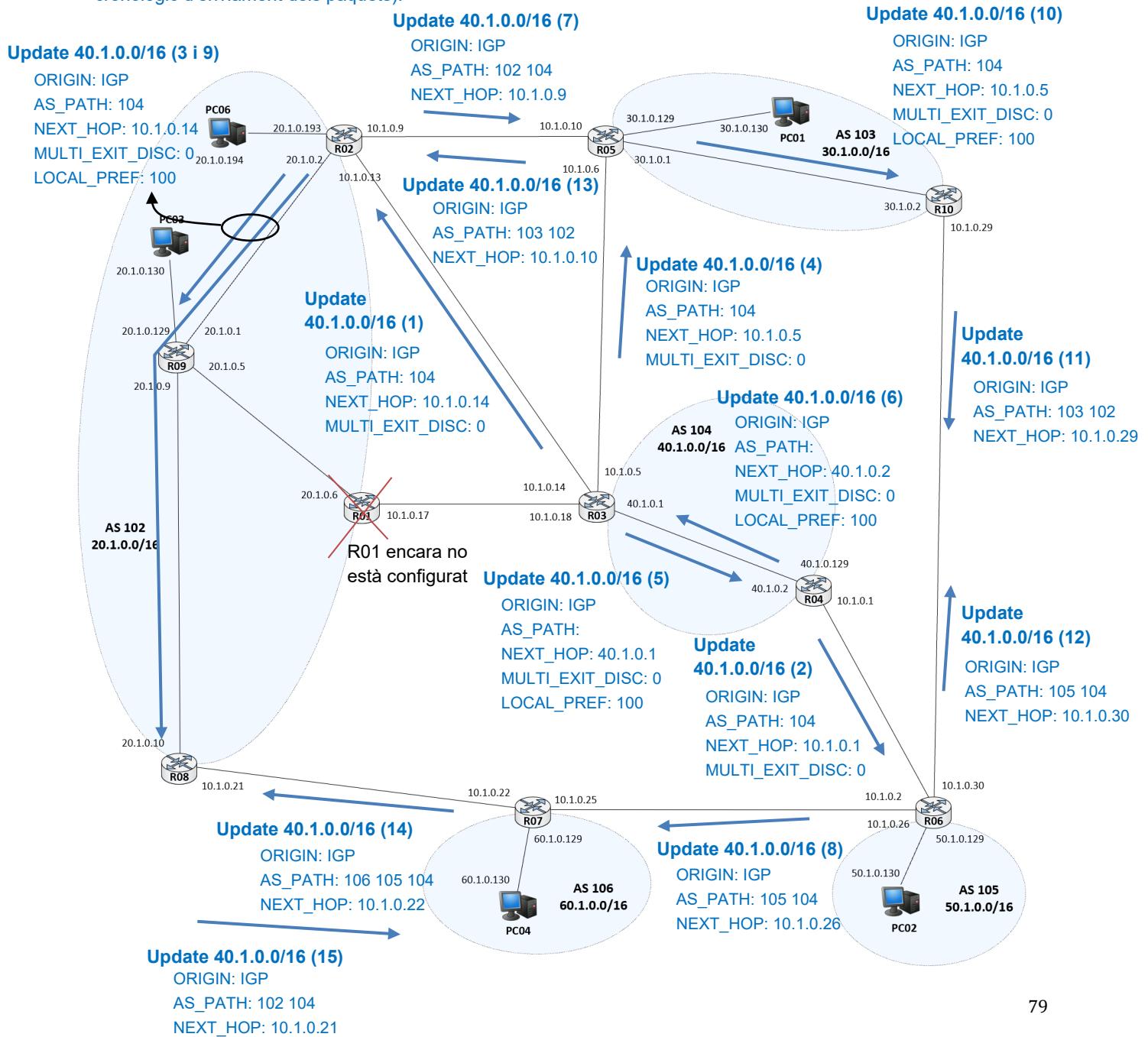
Exercici 3. Atribut LOCAL-PREFERENCE

L'objectiu d'aquest exercici és entendre el funcionament de l'atribut LOCAL-PREFERENCE. El LOCAL-PREFERENCE serveix per escollir el punt de sortida del sistema autònom per anar a un determinat prefix. Per tant, afecta al trànsit sortint (outbound) de l'AS.

Quan s'envia un prefix dins un UPDATE a un veí del mateix AS (peer iBGP) és obligatori incloure el LOCAL-PREFERENCE a la llista d'atributs. El valor per defecte és 100. Per contra, no s'envia mai l'atribut LOCAL-PREFERENCE als veïns d'un altre AS (peers eBGP).

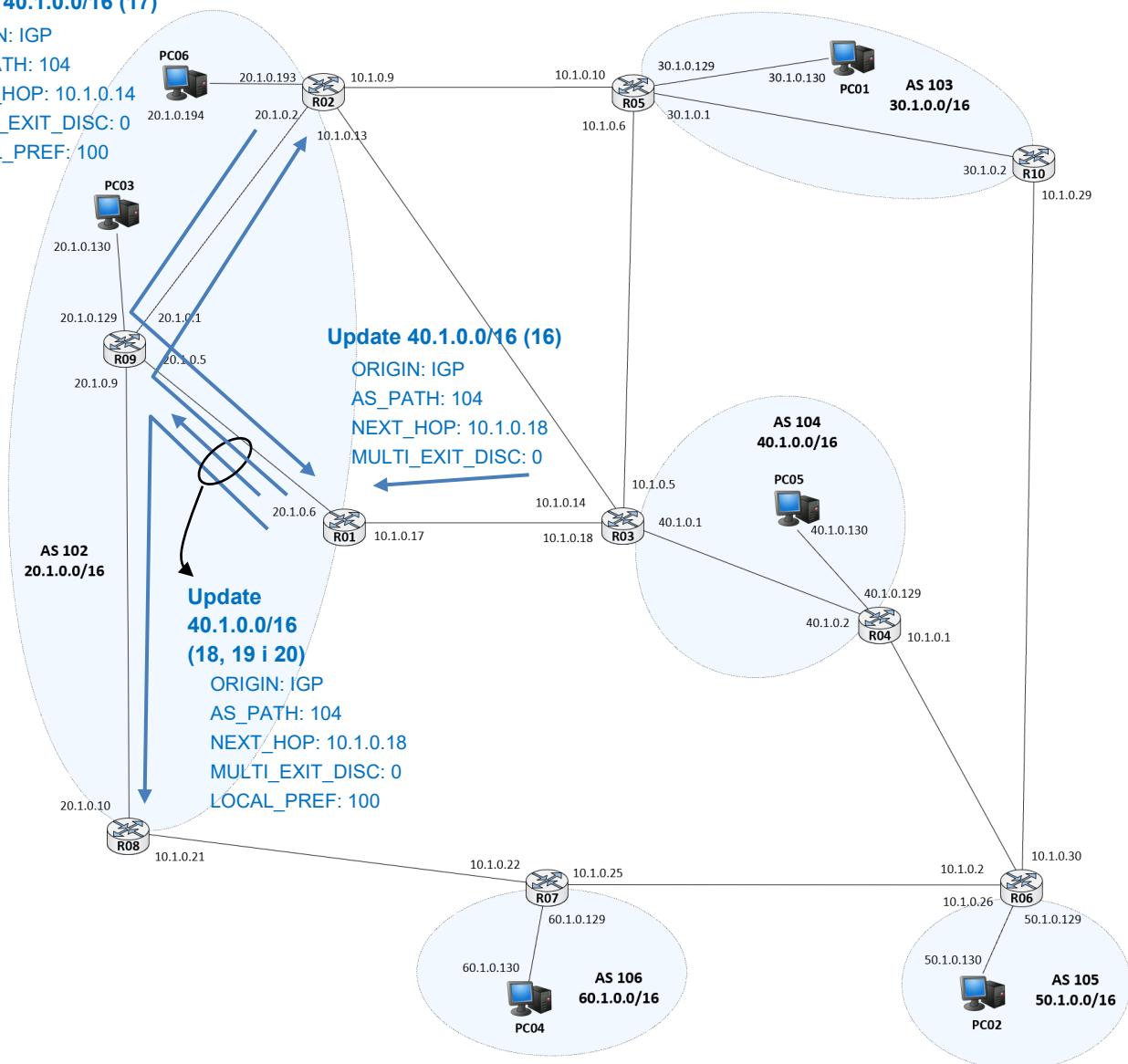
En aquest exercici haureu d'observar les rutes cap al prefix 40.1.0.0/16 que tenen els routers de l'AS 102 (R01, R02, R08 i R09) i haureu de modificar el punt de sortida de l'AS 102 que escullen aquests routers per arribar al prefix 40.1.0.0/16 modificant l'atribut LOCAL-PREFERENCE.

Abans de resoldre les preguntes de l'exercici, fixeu-vos com s'ha propagat el prefix 40.1.0.0/16 als diferents routers de l'escenari. A l'escenari d'aquesta segona part, tots els routers ja tenen la configuració BGP preparada excepte R01, que l'heu hagut de configurar a l'exercici 1. El que teniu a la figura següent és el resum dels paquets UPDATE amb el prefix 40.1.0.0/16 que s'han enviat els routers quan heu engegat el dimoni bgpd (sense R01) i, a la pàgina següent, teniu els paquets UPDATE amb el prefix 40.1.0.0/16 que s'envien quan configureu R01. Entre parèntesi tenui el número del paquet a la captura **CapturaPart2Exercici3Intro.pcapng** (utilitzeu aquest número per saber l'ordre cronològic d'enviament dels paquets).



Update 40.1.0.0/16 (17)

ORIGIN: IGP
 AS_PATH: 104
 NEXT_HOP: 10.1.0.14
 MULTI_EXIT_DISC: 0
 LOCAL_PREF: 100

**Update 40.1.0.0/16 (16)**

ORIGIN: IGP
 AS_PATH: 104
 NEXT_HOP: 10.1.0.18
 MULTI_EXIT_DISC: 0

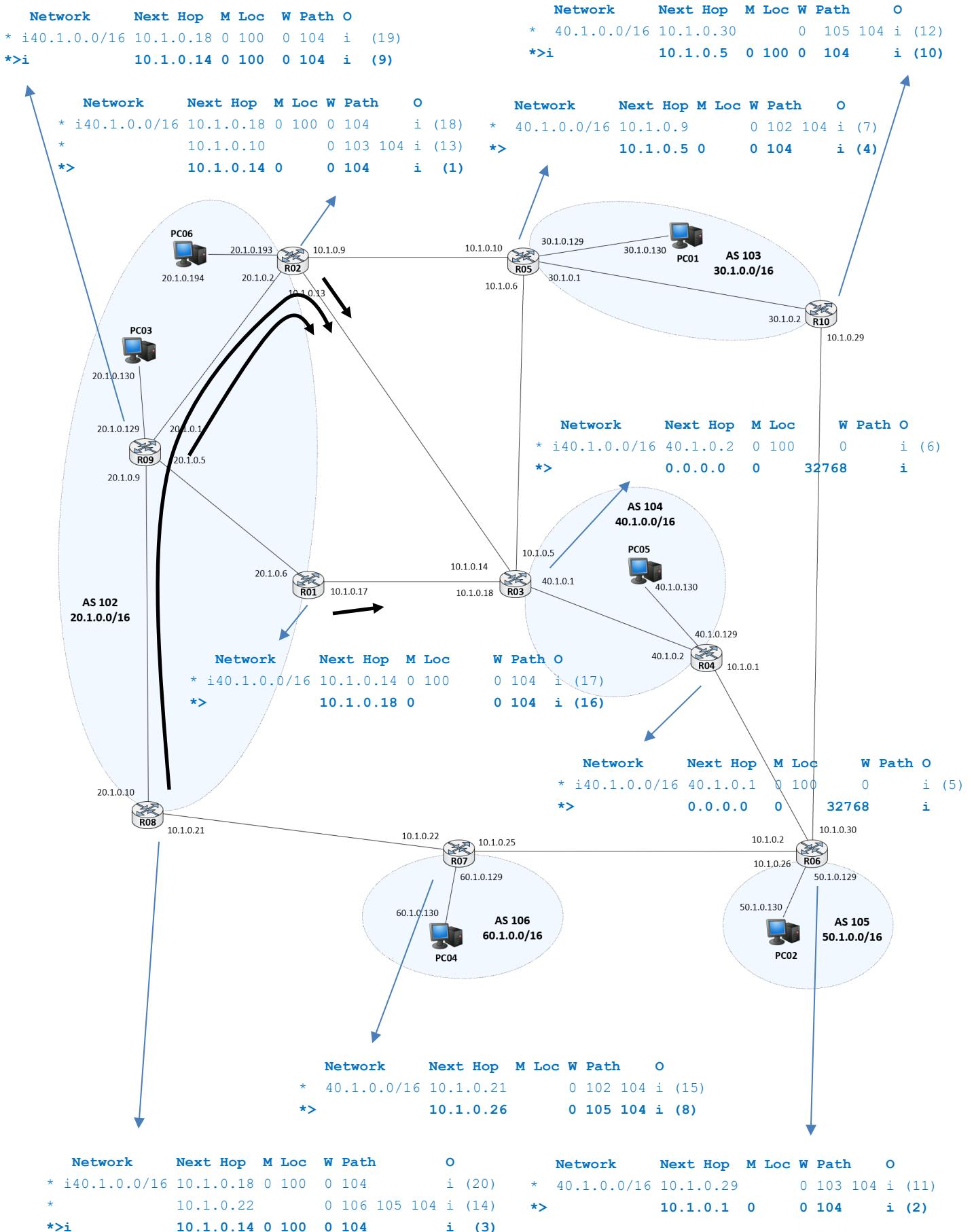
**Update
40.1.0.0/16
(18, 19 i 20)**

ORIGIN: IGP
 AS_PATH: 104
 NEXT_HOP: 10.1.0.18
 MULTI_EXIT_DISC: 0
 LOCAL_PREF: 100

Com s'observa a les figures, només els UPDATE que s'envien en sessions entre routers del mateix AS tenen l'atribut LOCAL_PREFERENCE a la llista d'atributs.

A l'esquema de la pàgina següent teniu les rutes al prefix 40.1.0.0/16 que té cada router com a conseqüència dels UPDATES rebuts dels diagrames anteriors (en aquest exemple, cada paquet rebut es correspon a una ruta de l'entrada de la taula BGP d'un router).

Al final de cada ruta teniu el paquet UPDATE que li ha permès al router aprendre la ruta. Fixeu-vos que, com més temps fa que s'ha après la ruta, més avall és a la taula.



Activeu el Wireshark a totes les interfícies de R01, R02 i R08.

La captura de paquets la teniu al fitxer [CapturaPart2Exercici3.pcapng](#)

Observeu la taula BGP de R01, R02 i R08 (show ip bgp).

- a. Quantes rutes apareixen per al prefix 40.1.0.0/16 en cadascun d'aquests routers? Per què hi ha aquestes rutes? Quina està seleccionada com a best? Per què?

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp'
      Network          Next Hop          Metric LocPrf Weight Path
* i40.1.0.0/16      10.1.0.14        0       100      0 104 i
* >                 10.1.0.18        0           0 104 i
```

Criteri 8 → Es prefereix una ruta eBGP sobre una ruta iBGP

```
lxc-attach -n R02 -- vtysh -c 'show ip bgp'
      Network          Next Hop          Metric LocPrf Weight Path
* i40.1.0.0/16      10.1.0.18        0       100      0 104 i
*                   10.1.0.10        0       103 104 i
* >                 10.1.0.14        0           0 104 i
```

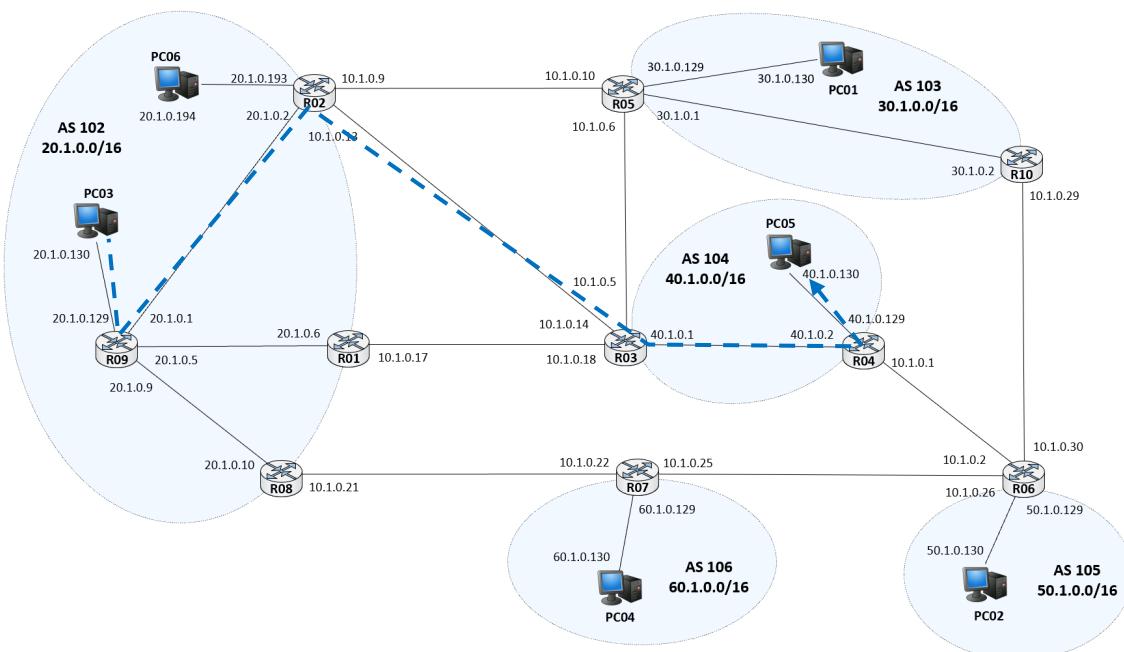
Criteri 8 → Entre les dues rutes amb AS_PATH 104, es prefereix la ruta eBGP sobre la ruta iBGP

```
lxc-attach -n R08 -- vtysh -c 'show ip bgp'
      Network          Next Hop          Metric LocPrf Weight Path
* i40.1.0.0/16      10.1.0.18        0       100      0 104 i
*                   10.1.0.22        0       106 105 104 i
* >i                10.1.0.14        0       100      0 104 i
```

Criteri 11 → Entre les dues rutes amb AS_PATH 104, com que són iBGP, s'escull la que té un ID BGP menor (la 10.1.0.14 s'apren de R02 que té ID 10.1.0.9 i la 10.1.0.18 s'apren de R01 que té ID 10.1.0.17)

Des d'un terminal del **PC** utilitzeu l'eina `tracepath_api` per saber per on passen els paquets que s'envien des del PC03 al PC05: `tracepath_api PC03 40.1.0.130`

- b. Apunteu-vos el camí que segueixen els paquets per anar del PC03 al PC05.



```

tracepath_api PC03 40.1.0.130

PC03  Matching FIB entries:
        20.1.0.129      eth0
    Next Hop: 20.1.0.129

R09  Matching FIB entries:
        40.1.0.0/16      20.1.0.2      eth2
    Next Hop: 20.1.0.2

R02  Matching FIB entries:
        40.1.0.0/16      10.1.0.14     eth2
    Next Hop: 10.1.0.14

R03  Matching FIB entries:
        40.1.0.128/27    40.1.0.2      eth1
    Next Hop: 40.1.0.2

R04  Matching FIB entries:
        40.1.0.128/27    0.0.0.0      directly connected
    Next Hop: 0.0.0.0

*** PATH SUMMARY ***
PC03 --> R09(20.1.0.129) --> R02(20.1.0.2) --> R03(10.1.0.14)
--> R04(40.1.0.2) --> 40.1.0.130 (Destination Address)

```

Comproveu també les taules d'encaminament BGP de la resta de *routers*, excepte els del sistema autònom 104, i apunteu-vos les rutes que tenen per arribar al prefix 40.1.0.0/16.

(Mireu el diagrama de la pàgina 80)

R03:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	40.1.0.2	0	100	0	i
*>	0.0.0.0	0		32768	i

Criteri 2 → Major weight

R04:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	40.1.0.1	0	100	0	i
*>	0.0.0.0	0		32768	i

Criteri 2 → Major weight

R05:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 40.1.0.0/16	10.1.0.9			0	102 104 i
*>	10.1.0.5	0		0 104	i

Criteri 5 → AS_PATH més curt

R10:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 40.1.0.0/16	10.1.0.30			0	105 104 i
*>i	10.1.0.5	0	100		0 104 i

Criteri 5 → AS_PATH més curt

R07:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 40.1.0.0/16	10.1.0.21			0	102 104 i
*>	10.1.0.26			0 105 104 i	

Criteri 10 → La primera que ha arribat

R06:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 40.1.0.0/16	10.1.0.29			0	103 104 i
*>	10.1.0.1	0			0 104 i

Criteri 5 → AS_PATH més curt

R01:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.14	0	100	0	104 i
*>	10.1.0.18	0			0 104 i

Criteri 8 → es prefereix la ruta eBGP sobre la iBGP

R02:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
*	10.1.0.10			0	103 104 i
*>	10.1.0.14	0			0 104 i

Criteri 8 → Entre les dues amb menor AS_PATH, es prefereix la ruta eBGP sobre la iBGP

R08:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
*	10.1.0.22			0	106 105 104 i
*>i	10.1.0.14	0	100		0 104 i

Criteri 11 → Entre les dues rutes amb menor AS_PATH es tria com a best la ruta apresa del router amb ID menor

- La ruta amb nexthop 10.1.0.18 s'ha après de R01 que té ID BGP 10.1.0.17
- La ruta amb nexthop 10.1.0.14 s'ha après de R02 que té ID BGP **10.1.0.9**

R09:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
*>i	10.1.0.14	0	100		0 104 i

Criteri 11 → Les rutes empaten en tot i s'escull la ruta apresa del router amb ID menor

- La ruta amb nexthop 10.1.0.18 s'ha après de R01 que té ID BGP 10.1.0.17
- La ruta amb nexthop 10.1.0.14 s'ha après de R02 que té ID BGP **10.1.0.9**

(L'ID dels routers el podeu veure amb la comanda 'show run' o bé 'show ip bgp summary' a cada router)

Configureu el *router* R01 per tal d'aconseguir que R01 sigui el punt de sortida del sistema autònom 102 per anar al prefix 40.0.1.0/16. Per fer-ho seguiu els passos següents:

- Configureu el valor de l'atribut LOCAL PREFERENCE que voleu que R01 anunciï als seu peers iBGP per al prefix 40.1.0.0/16 i forceu l'enviament de rutes (soft reset) de R03 a R01 per poder aplicar la política configurada:

```
R01# configure terminal
R01(config)# access-list P40 permit 40.1.0.0/16
R01(config)# route-map RMAP2 permit 10
R01(config-route-map)# match ip address P40
R01(config-route-map)# set local-preference _____ ← escolliu un valor
R01(config-route-map)# exit
R01(config)# route-map RMAP2 permit 20
R01(config-route-map)# exit
R01(config)# router bgp 102
R01(config-router)# neighbor 10.1.0.18 route-map RMAP2 in
R01(config-router)# exit
R01(config)# exit
R01# clear ip bgp 10.1.0.18 in
```

Nota: És imprescindible que entengueu les comandes que acabeu de configurar i la seva utilitat. (Consulteu la informació de les pàgines 17 i 18 per entendre les comandes i el seu significat).

Per aconseguir que R01 sigui el punt de sortida de l'AS 102 per anar al prefix 40.1.0.0/16 utilitzant l'atribut LOCAL-PREFERENCE cal fer que, quan R01 aprengui aquest prefix d'algun veí eBGP, l'anunciï als seus veïns iBGP amb un valor més alt de LOCAL-PREFERENCE del que assignen els altres routers BGP frontera de l'AS 102, per exemple 150:

```
R01# configure terminal
R01(config)# access-list P40 permit 40.1.0.0/16
R01(config)# route-map RMAP2 permit 10
R01(config-route-map)# match ip address P40
R01(config-route-map)# set local-preference 150
R01(config-route-map)# exit
R01(config)# route-map RMAP2 permit 20
R01(config-route-map)# exit
R01(config)# router bgp 102
R01(config-router)# neighbor 10.1.0.18 route-map RMAP2 in
R01(config-router)# exit
R01(config)# exit
```

En aquest cas s'ha configurat un route-map de nom **RMAP2** amb dues entrades. A la primera entrada, si arriba el prefix definit per la llista d'accés **P40** (que és el prefix 40.1.0.0/16) es fixa el valor de l'atribut local-preference a 150. Per a la resta de prefixes, la segona entrada del route-map permet que passin sense modificacions. Aquest route-map s'aplica a tots els paquets que arriben (in) del veí 10.1.0.18.

Per tal que es pugui aplicar la nova política, cal forçar que el veí 10.1.0.18 envii novament les seves best a R01 i per fer-ho enviant un missatge ROUTE-REFRESH cal executar la comanda següent:

```
R01# clear ip bgp 10.1.0.18 in
```

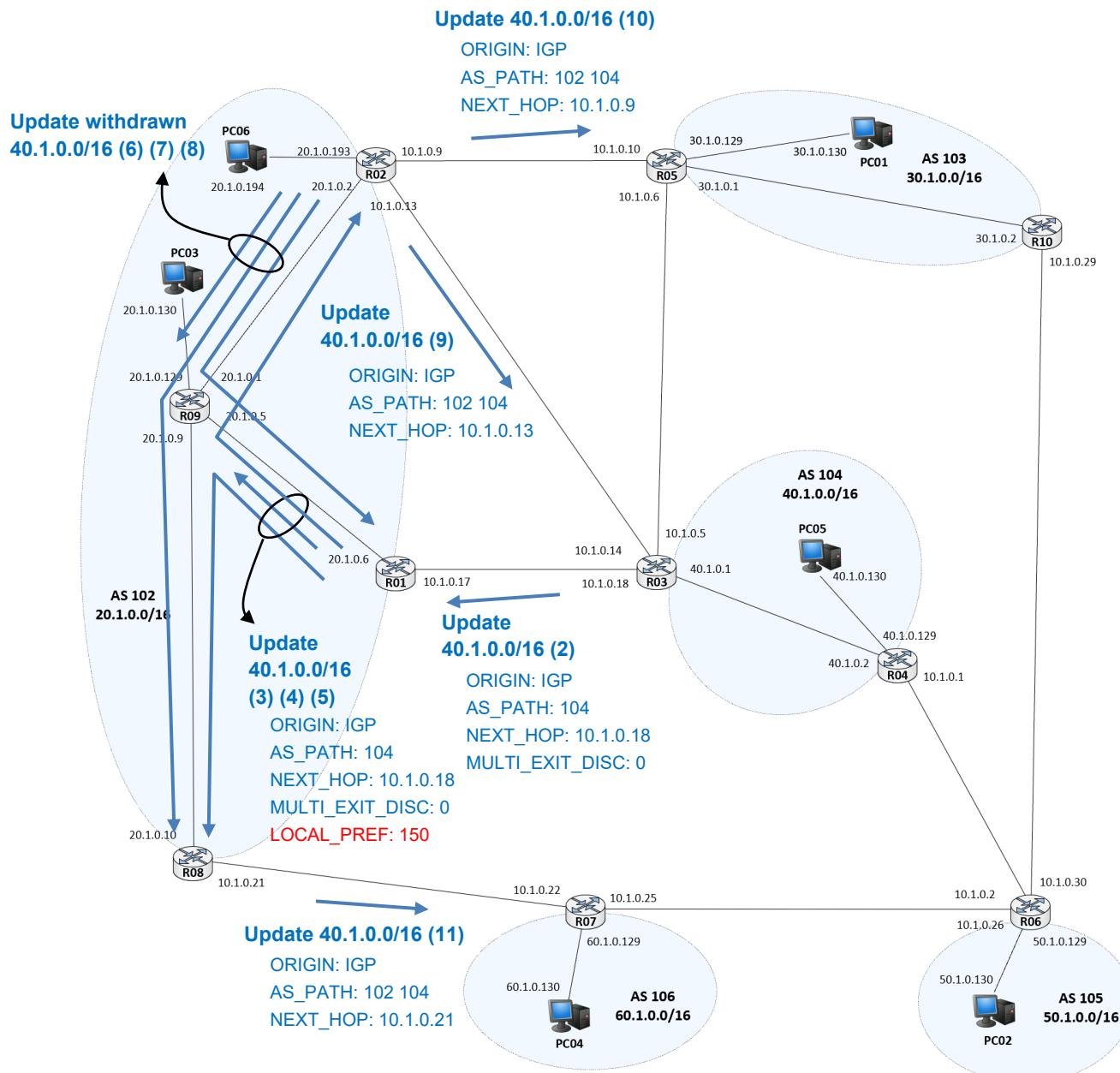
Com podeu comprovar a la captura **CapturaPart2Exercici3.pcapng**, després que R01 envii un missatge ROUTE-REFRESH a R03 (paquet 1), R03 envia les best de la seva taula BGP a R01. Quan arriba a R01 l'Update del prefix 40.1.0.0/16 (paquet 2), R01 li aplica el route-map RMAP2 i li assigna un valor de LOCAL-PREFERENCE = 150. R01 envia el prefix 40.1.0.0/16 amb aquest valor de LOCAL-PREFERENCE als seus tres veïns iBGP: R09 (paquet 3), R08 (paquet 4) i R02 (paquet 5).

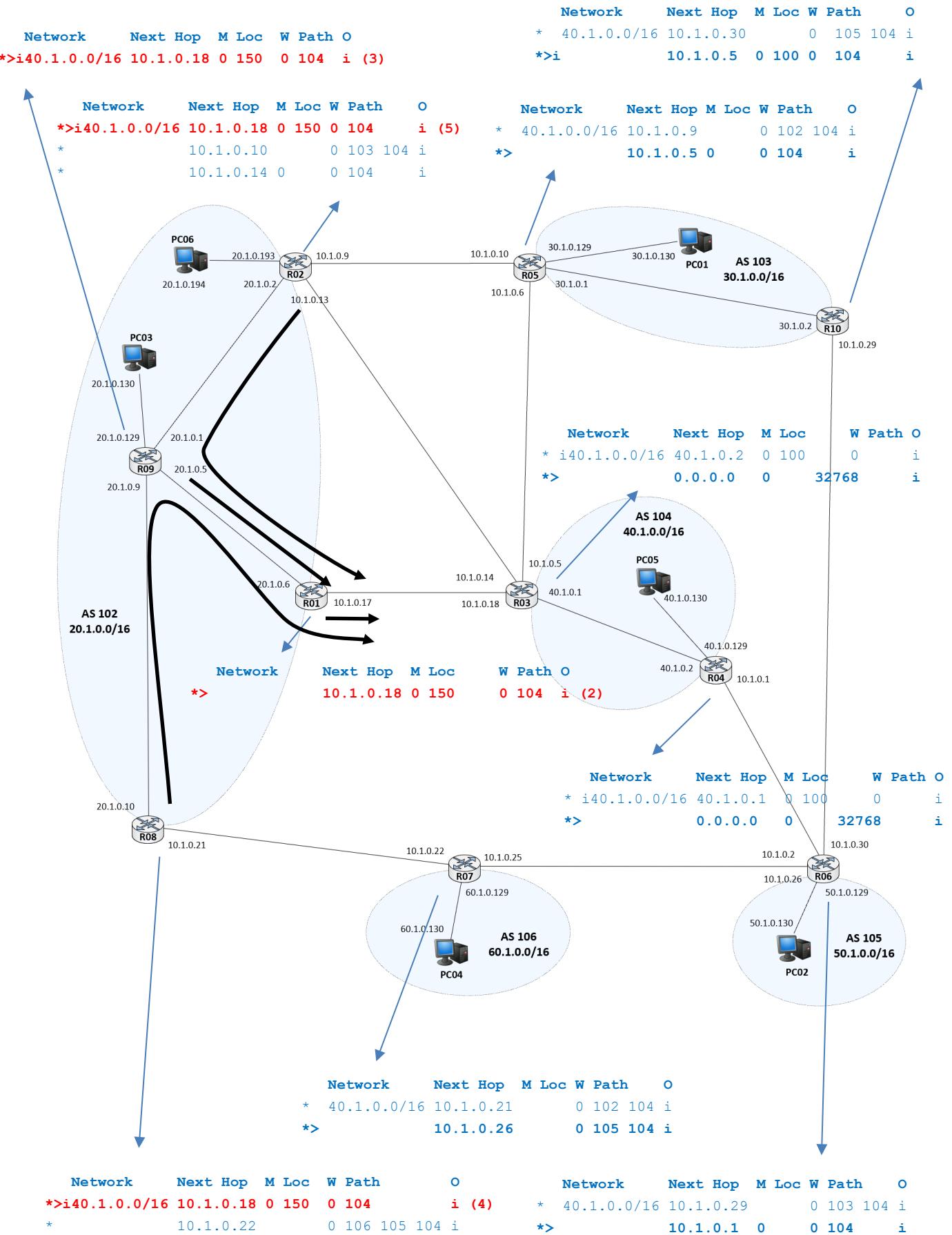
R02 modifica la seva best i ho notifica als seus veïns.

- Envia un Update withdrawn a tots els seus veïns iBGP (paquets 6, 7 i 8) perquè ara la seva best pel prefix 40.1.0.0/16 és la ruta interna al seu AS, sortint per R01.
- Envia un Update del prefix 40.1.0.0/16 (sense l'atribut LOCAL-PREFERENCE) als seus veïns eBGP, que són R03 (paquet 9) i R05 (paquet 10), per notificar-los el canvi de best.

La best de R08 també es modifica (abans sortia per R02 i ara surt per R01) i, per tant, ho notifica al seu veí R07 (paquet 11).

R09 també canvia la seva best però com que la nova best l'ha après via iBGP i només té veïns iBGP, no envia cap missatge anunciant el canvi.





Observeu la taula BGP de R01, R02 i R08: show ip bgp

- a. Quantes rutes apareixen ara per al prefix 40.1.0.0/16? Quina està seleccionada com a **best**? Per què?

```
lxc-attach -n R01 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.14	0	100	0	104 i
*>	10.1.0.18	0		0	104 i

Després:

*> 40.1.0.0/16	10.1.0.18	0	150	0	104 i
----------------	-----------	---	-----	---	-------

Només té una ruta cap al prefix 40.1.0.0/16. La ruta que tenia nexthop 10.1.0.14 l'ha esborrat perquè R02 li ha enviat el withdrawn.

```
lxc-attach -n R02 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
*	10.1.0.10			0	103 104 i
*>	10.1.0.14	0		0	104 i

Després:

*>i40.1.0.0/16	10.1.0.18	0	150	0	104 i
*	10.1.0.10			0	103 104 i
*	10.1.0.14	0		0	104 i

Criteri 3 → Escull la ruta amb LOCAL-PREFERENCE més alt. Si no hi ha res al camp local-preference, es considera el valor per defecte, que és 100.

```
lxc-attach -n R08 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
*	10.1.0.22			0	106 105 104 i
*>i	10.1.0.14	0	100	0	104 i

Després:

*>i40.1.0.0/16	10.1.0.18	0	150	0	104 i
*	10.1.0.22			0	106 105 104 i

Criteri 3 → Escull la ruta amb LOCAL-PREFERENCE més alt. Si el camp està buit, es considera el valor per defecte, que és 100. La ruta que tenia nexthop 10.1.0.14 l'ha esborrat perquè R02 li ha enviat el withdrawn.

```
lxc-attach -n R09 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i40.1.0.0/16	10.1.0.18	0	100	0	104 i
*>i	10.1.0.14	0	100	0	104 i

Després:

*>i40.1.0.0/16	10.1.0.18	0	150	0	104 i
----------------	-----------	---	-----	---	-------

Només té una ruta cap al prefix 40.1.0.0/16. La ruta que tenia nexthop 10.1.0.14 l'ha esborrat perquè R02 li ha enviat el withdrawn.

Atureu les captures del Wireshark i observeu-les.

- b. Propaga R01 l'atribut LOCAL PREFERENCE per al prefix 40.1.0.0/16 al seu veí R02? Per què?

Sí perquè l'atribut LOCAL-PREFERENCE es transmet en sessions iBGP. Mireu el diagrama de la pàgina 86.

- c. Propaga R02 l'atribut LOCAL PREFERENCE per al prefix 40.1.0.0/16 al seu veí R05? Per què?

No perquè l'atribut LOCAL-PREFERENCE no s'envia a veïns eBGP. Mireu el diagrama de la pàgina 86.

- d. Notifica R02 el seu canvi de ruta a R08? Per què?

No. Les rutes apreses via iBGP no s'anuncien als veïns iBGP. Mireu el diagrama de la pàgina 86.

Comproveu també les taules d'encaminament BGP de la resta de routers, excepte els de l'AS 104, i fixeu-vos amb el prefix 40.1.0.0/16.

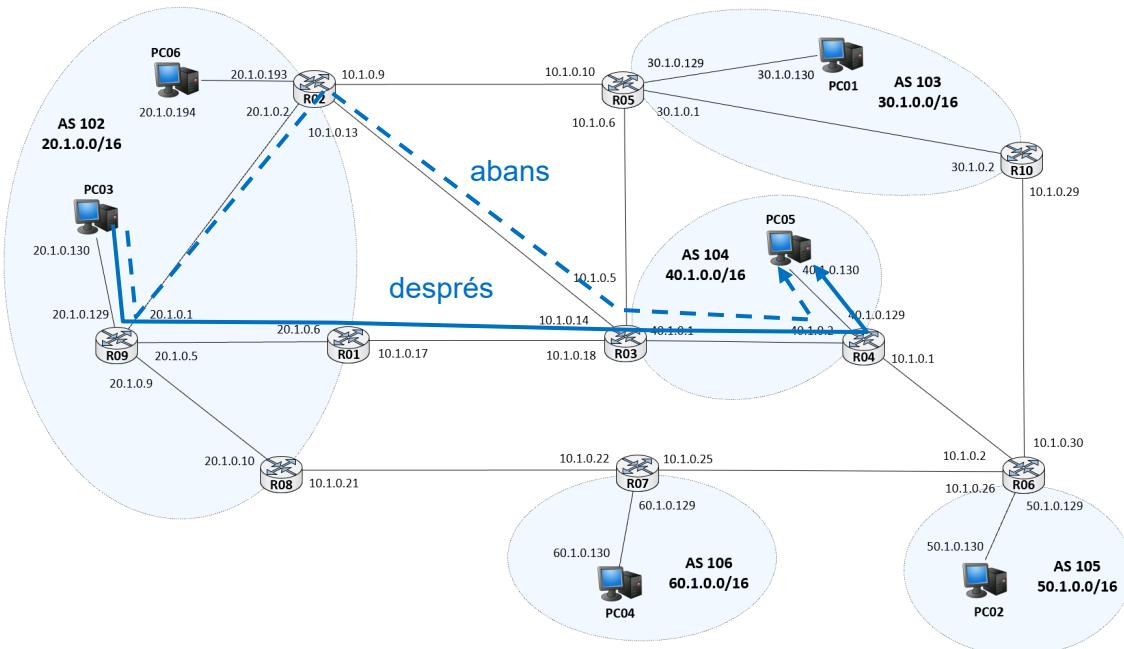
- e. Hi ha algun canvi en algun router?

No. Mireu el diagrama de la pàgina 87.

Des d'un terminal del **PC** utilitzeu l'eina `tracepath_api` per saber per on passen els paquets que s'envien des del PC03 al PC05: `tracepath api PC03 40.1.0.130`

- f. Compareu el resultat amb el que heu obtingut a l'apartat b.

Abans el trànsit per anar al prefix 40.1.0.0/16 sortia de l'AS 102 per R02 i ara surt per R01.



tracepath_api PC03 40.1.0.130

(· · ·)

*** PATH SUMMARY ***

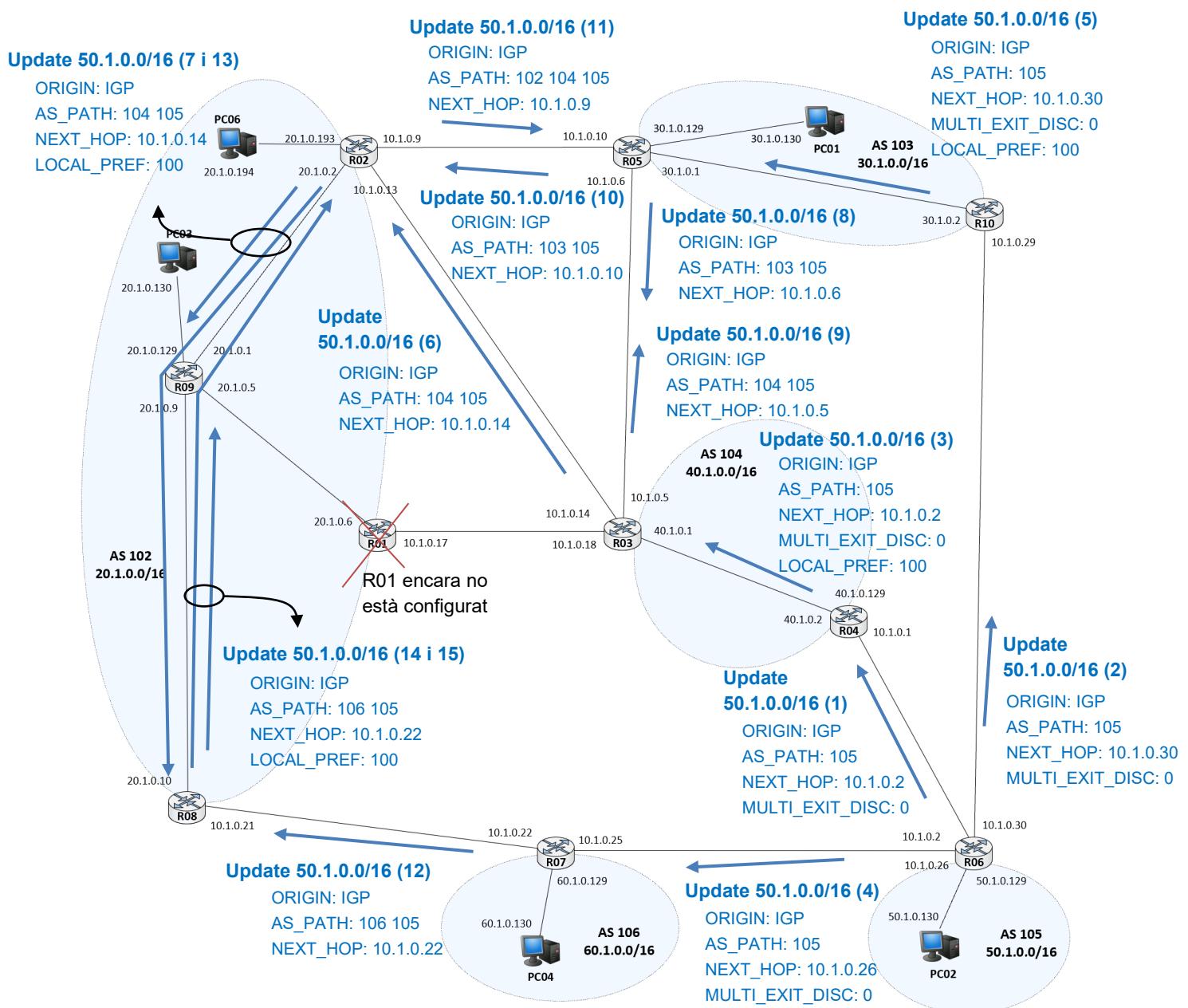
PC03 --> R09(20.1.0.129) --> R01(20.1.0.6) --> R03(10.1.0.18) --> R04(40.1.0.2)
--> 40.1.0.130 (Destination Address)

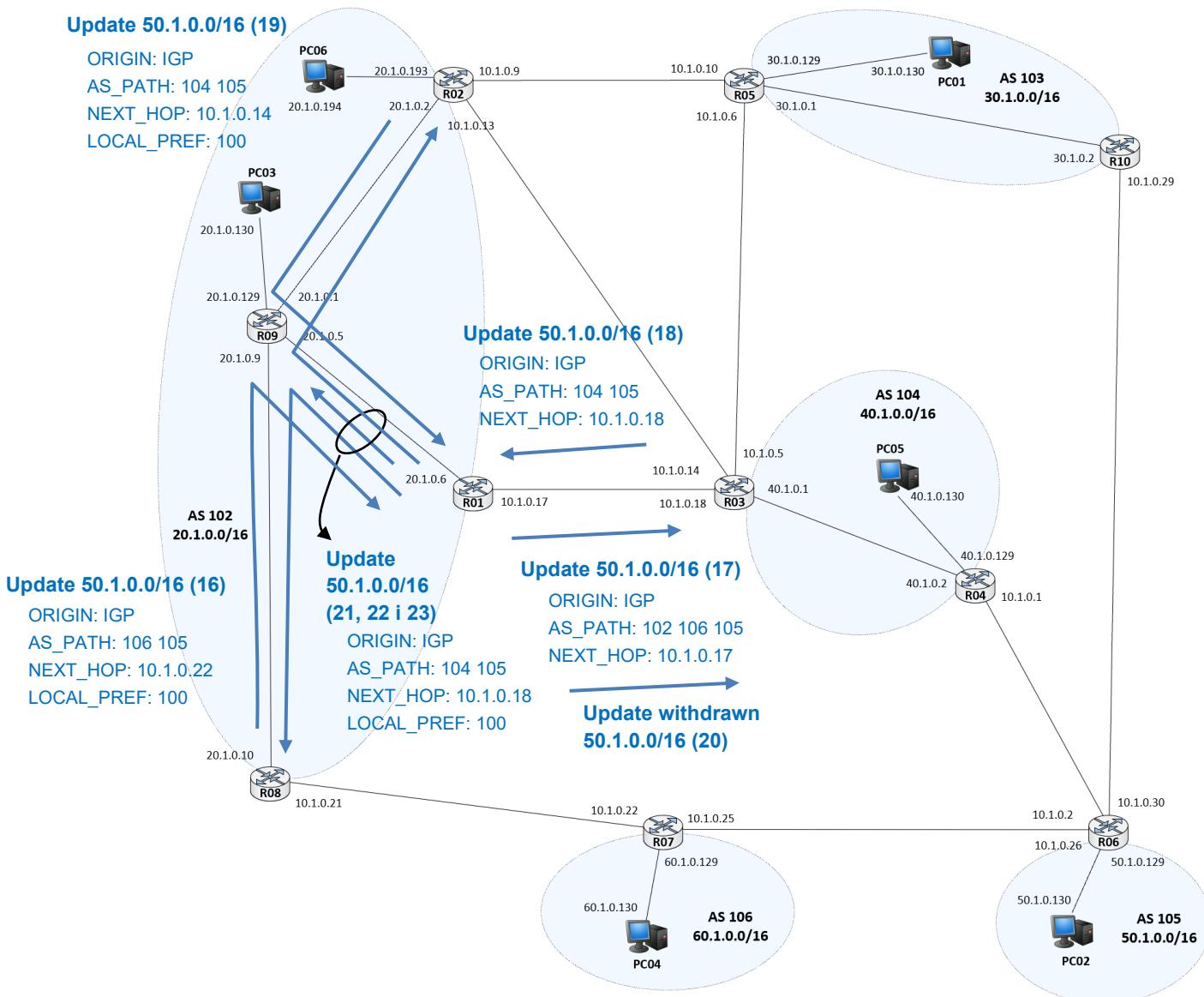
Exercici 4. WEIGHT

L'objectiu d'aquest exercici és entendre el funcionament del paràmetre WEIGHT. Aquest paràmetre serveix perquè el router pugui escollir, de forma local, per on enviar els paquets dirigits a un determinat prefix. Aquest paràmetre no s'envia mai als routers veïns.

En aquest exercici haureu d'observar les rutes cap al prefix 50.1.0.0/16 que té R02 i haureu de modificar la ruta que escull R02 per arribar al prefix 50.1.0.0/16 modificant el paràmetre WEIGHT.

Abans de resoldre les preguntes de l'exercici, fixeu-vos com s'ha propagat el prefix 50.1.0.0/16 als diferents routers de l'escenari. A l'escenari d'aquesta segona part, tots els routers ja tenien la configuració BGP preparada excepte R01, que l'heu hagut de configurar a l'exercici 1. El que teniu a la figura següent és el resum dels paquets UPDATE amb el prefix 50.1.0.0/16 que s'han enviat els routers quan heu engegat el dimoni bgpd (sense R01) i, a la pàgina següent, teniu els paquets UPDATE amb el prefix 50.1.0.0/16 que s'envien quan configureu R01. Entre parèntesi teniu el número del paquet a la captura **CapturaIntroExercici4Part2.pcapng** (utilitzeu aquest número per saber l'ordre cronològic d'enviament dels paquets)

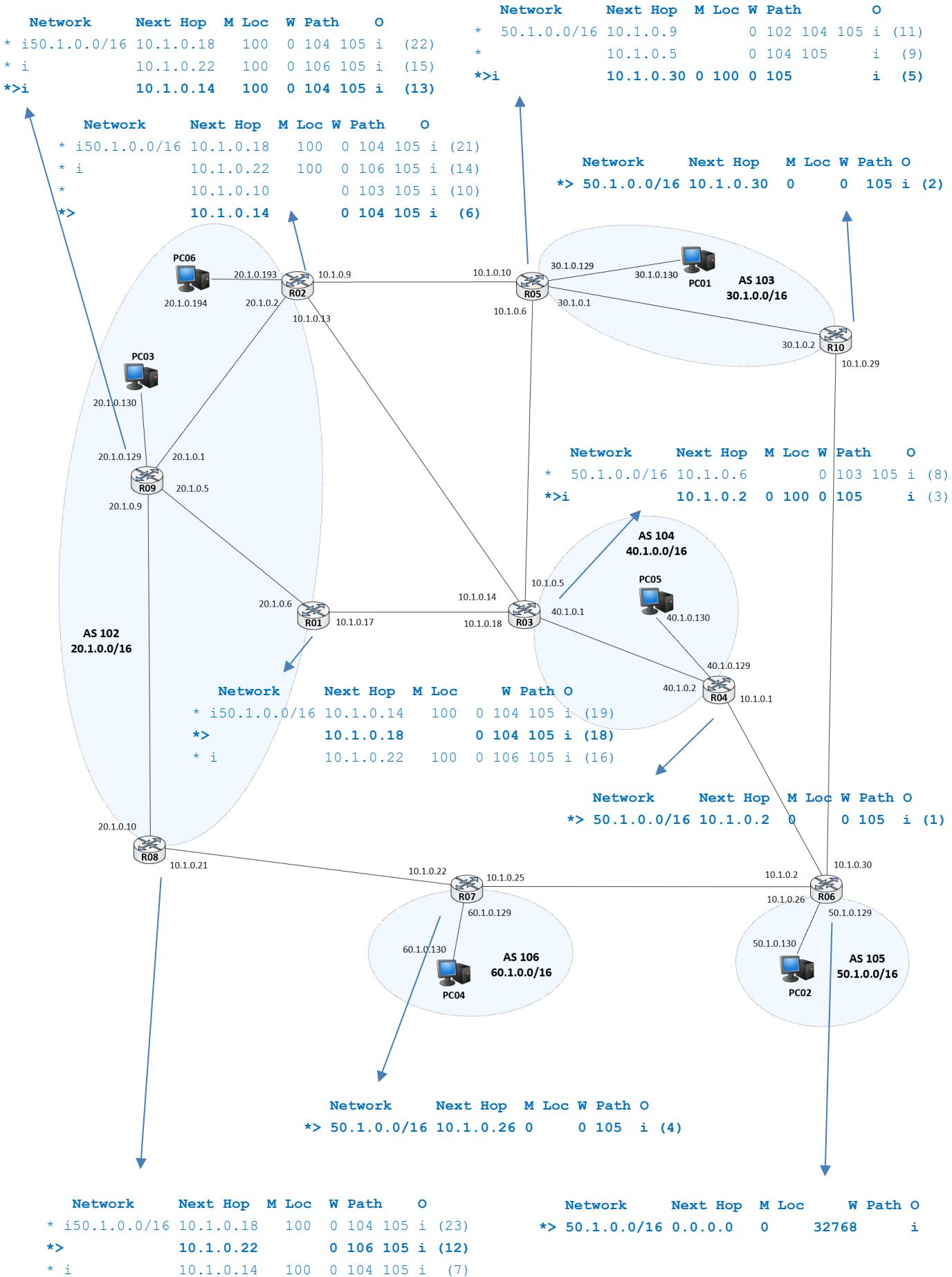




Com s'observa a les figures, el paràmetre WEIGHT no s'envia a cap veí BGP.

A l'esquema de la pàgina següent teniu les rutes al prefix 50.1.0.0/16 que té cada router com a conseqüència dels UPDATES rebuts dels diagrames anteriors (en aquest exemple, cada paquet es correspon a una ruta de l'entrada de la taula BGP d'un router, excepte el paquet 17 perquè R01 escull la ruta amb nexthop 10.1.0.18 (R03) per anar al prefix 50.1.0.0/16 i, com que R03 és el seu nou nexthop, R01 ha de demanar a R03 que esborri la ruta que li havia anunciat al paquet 17).

Al final de cada ruta teniu el paquet UPDATE que li ha permès al router aprendre la ruta. Fixeu-vos que, com més temps fa que s'ha après la ruta, més avall és a la taula.



Activeu el Wireshark per a que capturi paquets per totes les interfícies de R02.

La captura de paquets la teniu al fitxer [CapturaPart2Exercici3.pcapng](#)

Observeu la taula BGP de R02: `lxc-attach -n R02 -- vtysh -c 'show ip bgp'`

- a. Quantes rutes apareixen per al prefix **50.1.0.0/16**? Quina està seleccionada com a **best**? Per què?

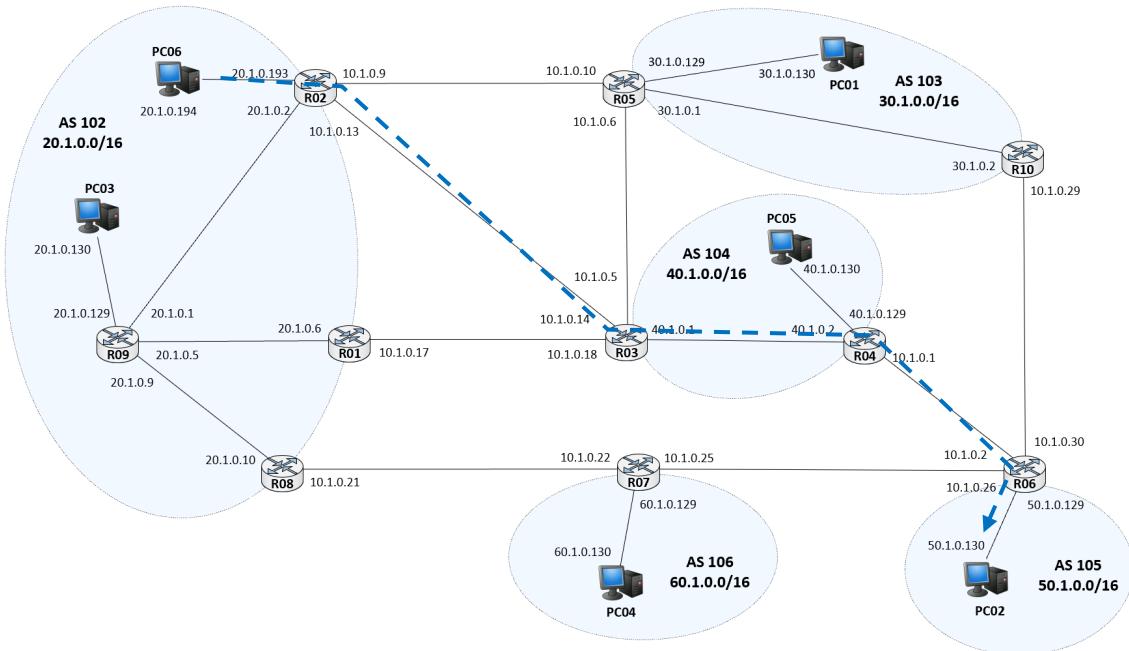
```
lxc-attach -n R02 -- vtysh -c 'show ip bgp'
```

Network	Next Hop	Metric	LocPrf	Weight	Path
* 50.1.0.0/16	10.1.0.18	100	0	104	105 i
* i	10.1.0.22	100	0	106	105 i
*	10.1.0.10		0	103	105 i
*>	10.1.0.14		0	104	105 i

Criteri 10 → De les dues eBGP, es queda amb la que ha arribat primer.

Des d'un terminal del **PC** utilitzeu l'eina `tracepath_api` per saber per on passen els paquets que s'envien des del PC06 al PC02: `tracepath_api PC06 50.1.0.130`

- b. Apunteu-vos el camí que segueixen els paquets per anar del PC06 al PC02.



```
tracepath_api PC06 50.1.0.130
```

```
( . . . )
```

```
*** PATH SUMMARY ***
PC06 --> R02(20.1.0.193) --> R03(10.1.0.14) --> R04(40.1.0.2) --> R06(10.1.0.2)
--> 50.1.0.130 (Destination Address)
```

Comproveu també les taules d'encaminament BGP de la resta de *routers*, excepte els del sistema autònom 105, i apunteu-vos les rutes que tenen per arribar al prefix 50.1.0.0/16.

(Mireu el diagrama de la pàgina 92)

R03:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 50.1.0.0/16	10.1.0.6			0	103 105 i
*>i	10.1.0.2	0	100		0 105 i

Criteri 5 → AS_PATH més curt

R04:

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 50.1.0.0/16	10.1.0.2	0			0 105 i

Només té una ruta

R05:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 50.1.0.0/16	10.1.0.9			0	102 104 105 i
*	10.1.0.5				0 104 105 i
*>i	10.1.0.30	0	100		0 105 i

Criteri 5 → AS_PATH més curt

R10:

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 50.1.0.0/16	10.1.0.30	0			0 105 i

Només té una ruta

R07:

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 50.1.0.0/16	10.1.0.26	0			0 105 i

Només té una ruta

R06:

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 50.1.0.0/16	0.0.0.0	0		32768	i

Només té una ruta

R01:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i50.1.0.0/16	10.1.0.14	100		0	104 105 i
*>	10.1.0.18				0 104 105 i

Criteri 8 → es prefereix la ruta eBGP sobre la iBGP

R02:

Network	Next Hop	Metric	LocPrf	Weight	Path
* i50.1.0.0/16	10.1.0.18	100		0	104 105 i
* i	10.1.0.22	100		0	106 105 i
*	10.1.0.10			0	103 105 i
*>	10.1.0.14				0 104 105 i

Criteri 10 → De les dues eBGP, es queda amb la que ha arribat primer.

R08:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 150.1.0.0/16	10.1.0.18	100	0	104	105 i
*>	10.1.0.22		0	106	105 i
* i	10.1.0.14	100	0	104	105 i

Criteri 8 → es prefereix la ruta eBGP sobre la iBGP

R09:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 150.1.0.0/16	10.1.0.18	100	0	104	105 i
* i	10.1.0.22	100	0	106	105 i
*>i	10.1.0.14	100	0	104	105 i

Criteri 11 → Les rutes empaten en tot i s'escull la ruta apresa del router amb ID menor

- La ruta amb nexthop 10.1.0.18 s'ha après de R01 que té ID BGP 10.1.0.17
- La ruta amb nexthop 10.1.0.22 s'ha après de R08 que té ID BGP 10.1.0.21
- La ruta amb nexthop 10.1.0.14 s'ha après de R02 que té ID BGP **10.1.0.9**

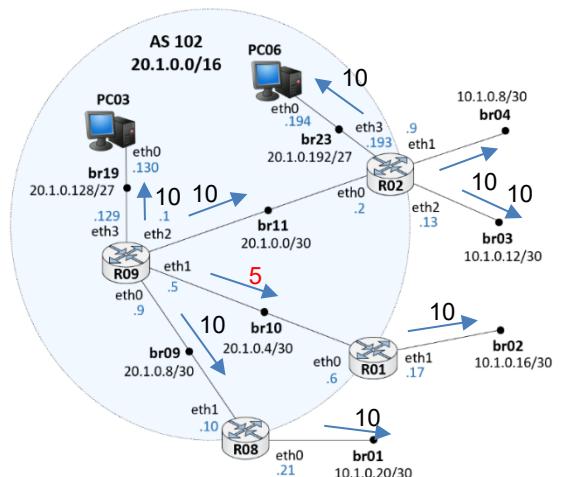
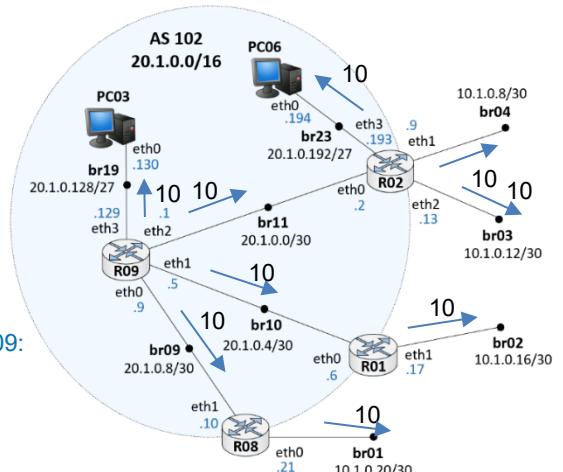
(L'ID dels routers el podeu veure amb la comanda 'show run' o bé 'show ip bgp summary' a cada router)

Fixeu-vos que, en aquest escenari, tots els enllaços de l'AS 102 tenen cost 10 en mètrica OSPF, per tant, des del punt de vista de R09, els tres next-hop estan a distància 20 i per això quan es mira el criteri 9 (que diu que es tria la ruta amb menor mètrica cap al next-hop segons el protocol d'encaminament interior) les tres rutes empaten.

```
lxc-attach -n R09 -- vtysh -c 'show ip route'
( ... )
O>* 10.1.0.8/30 [110/20] via 20.1.0.2, eth2
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth2
O>* 10.1.0.16/30 [110/20] via 20.1.0.6, eth1
O>* 10.1.0.20/30 [110/20] via 20.1.0.10, eth0
( ... )
```

Suposeu que es modifica la mètrica OSPF de la interfície eth1 de R09:

```
root@api-mv:~# lxc-attach -n R09 -- vtysh -c
R09# conf term
R09(config)# interface eth1
R09(config-if)# ip ospf cost 5
R09(config-if)# exit
R09(config)# exit
R09# show run
( ... )
interface eth1
  ip address 20.1.0.5/30
  ip ospf cost 5
( ... )
O>* 10.1.0.8/30 [110/20] via 20.1.0.2, eth2
O>* 10.1.0.12/30 [110/20] via 20.1.0.2, eth2
O>* 10.1.0.16/30 [110/15] via 20.1.0.6, eth1
O>* 10.1.0.20/30 [110/20] via 20.1.0.10, eth0
( ... )
```



Si fos així, la best que escolliria R09 per al prefix 50.1.0.0/16 seria la que té nexthop 10.1.1.14

Sense modificar la mètrica:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 50.1.0.0/16	10.1.0.18	100	0	104	105 i
* i	10.1.0.22	100	0	106	105 i
*>i	10.1.0.14	100	0	104	105 i

Criteri 11 → Les rutes empaten en tot i s'escull la ruta apresa del router amb ID menor

Modificar la mètrica:

Network	Next Hop	Metric	LocPrf	Weight	Path
*>50.1.0.0/16	10.1.0.18	100	0	104	105 i
* i	10.1.0.22	100	0	106	105 i
* i	10.1.0.14	100	0	104	105 i

Criteri 9 → S'escull la ruta amb menor mètrica al next hop segons el protocol d'encaminament interno

Configureu el *router* R02 per tal d'aconseguir que R02 canviï la seva millor ruta (best) per anar al prefix 50.1.0.0/16. (És a dir, si ara està fent servir com a *nexthop* l'adreça 10.1.0.10, forceu a que esculli la 10.1.0.14. Si per contra està fent servir la 10.1.0.14, configureu el WEIGHT per a que esculli la 10.1.0.10).

- Definiu dos *route-map* amb WEIGHT diferent per al prefix 50.1.0.0/16 i associeu-ne un a les rutes rebudes de cada veí eBGP de R02:

```
R02# configure terminal
R02(config)# access-list P50 permit 50.1.0.0/16
R02(config)# route-map RMAP2 permit 10
R02(config-route-map)# match ip address P50
R02(config-route-map)# set weight _____ ← escolliu un valor
R02(config-route-map)# exit
R02(config)# route-map RMAP2 permit 20
R02(config-route-map)# exit
R02(config)# route-map RMAP3 permit 10
R02(config-route-map)# match ip address P50
R02(config-route-map)# set weight _____ ← escolliu un valor
R02(config-route-map)# exit
R02(config)# route-map RMAP3 permit 20
R02(config-route-map)# exit
R02(config)# router bgp 102
R02(config-router)# neighbor 10.1.0.10 route-map RMAP2 in
R02(config-router)# neighbor 10.1.0.14 route-map RMAP3 in
R02(config-router)# exit
R02(config)# exit
R02# clear ip bgp 10.1.0.10 in
R02# clear ip bgp 10.1.0.14 in
```

Nota: És imprescindible que entengueu les comandes que acabeu de configurar i la seva utilitat. (Consulteu la informació de les pàgines 17 i 18 per entendre les comandes i el seu significat).

R02 ha escollit la ruta amb nexthop 10.1.0.14 per anar al prefix 50.1.0.0/16. Per tal d'aconseguir que R02 esculli la ruta amb nexthop 10.1.0.10 modificant el weight assignat a aquesta ruta cal assignar un pes més alt a la ruta que anuncia el veí 10.1.0.10. En aquest cas s'han configurat dos route-maps per assignar un pes 500 al prefix 50.1.0.0/16 anunciat pel veí 10.1.0.10 i un pes 200 al prefix 50.1.0.0/16 anunciat pel veí 10.1.0.14.

```

lxc-attach -n R02 -- vtysh
R02# configure terminal
R02(config)# access-list P50 permit 50.1.0.0/16
R02(config)# route-map RMAP2 permit 10
R02(config-route-map)# match ip address P50
R02(config-route-map)# set weight 500
R02(config-route-map)# exit
R02(config)# route-map RMAP2 permit 20
R02(config-route-map)# exit
R02(config)# route-map RMAP3 permit 10
R02(config-route-map)# match ip address P50
R02(config-route-map)# set weight 200
R02(config-route-map)# exit
R02(config)# route-map RMAP3 permit 20
R02(config-route-map)# exit
R02(config)# router bgp 102
R02(config-router)# neighbor 10.1.0.10 route-map RMAP2 in
R02(config-router)# neighbor 10.1.0.14 route-map RMAP3 in
R02(config-router)# exit
R02(config)# exit
R02# show run
(...)
router bgp 102
bgp router-id 10.1.0.9
network 20.1.0.0/16
neighbor 10.1.0.10 remote-as 103
neighbor 10.1.0.10 route-map RMAP2 in
neighbor 10.1.0.14 remote-as 104
neighbor 10.1.0.14 route-map RMAP3 in
neighbor 10.1.0.14 route-map RMAP1 out
neighbor 20.1.0.1 remote-as 102
neighbor 20.1.0.6 remote-as 102
neighbor 20.1.0.10 remote-as 102
!
access-list P50 permit 50.1.0.0/16
(...)
route-map RMAP2 permit 10
match ip address P50
set weight 500
!
route-map RMAP2 permit 20
!
route-map RMAP3 permit 10
match ip address P50
set weight 200
!
route-map RMAP3 permit 20
(...)

```

Per tal que es pugui aplicar la nova política, cal forçar que els veïns 10.1.0.10 i 10.1.0.14 tornin a enviar les seves best a R02. Per fer-ho enviant un missatge ROUTE-REFRESH, es proposa utilitzar les comandes següents:

```

R02# clear ip bgp 10.1.0.10 in
R02# clear ip bgp 10.1.0.14 in

```

Com es pot observar a la captura **CapturaPart2Exercici4.pcapng**, R02 envia un missatge ROUTE-REFRESH a R05 (paquet 1) i a R03 (paquet 2) per forçar que aquest veïns li enviïn les best de la seva taula BGP. El primer router que les envia és R05. Quan arriba l'Update del prefix 50.1.0.0/16 que envia R05 (paquet 3), R02 li aplica el route-map RMAP2 i li assigna un valor de WEIGHT = 500 i, com que aquesta ruta té un pes més alt que la que té del veí 10.1.0.14 (que de moment és 0), R02 canvia la best i ho anuncia als seus veïns. Com que R05 és el seu nou next-hop per al prefix 50.1.0.0/16, R02 li envia un Update amb un withdrawn del prefix 50.1.0.0/16 (paquet 4). Als veïns iBGP de l'AS 102 els hi envia un Update sense modificar el next-hop ni l'AS_PATH (paquets 5, 6 i 7). Al veí R03 li envia un Update modificant el next-hop i l'AS_PATH (paquet 9). En cap cas s'envia el weight a cap veí. A continuació R02 rep les best de R03 (paquet 8) i les fa passar pel route-map RMAP3 i al prefix 50.1.0.0/16 rebut del veí 10.1.0.14 li assigna un weight 200. Com a conseqüència d'aquest canvi, no s'envia cap paquet.

Update 50.1.0.0/16 (5) (6) (7)

ORIGIN: IGP
AS_PATH: 103 105
NEXT_HOP: 10.1.0.10
MULTI_EXIT_DISC: 0
LOCAL_PREF: 100

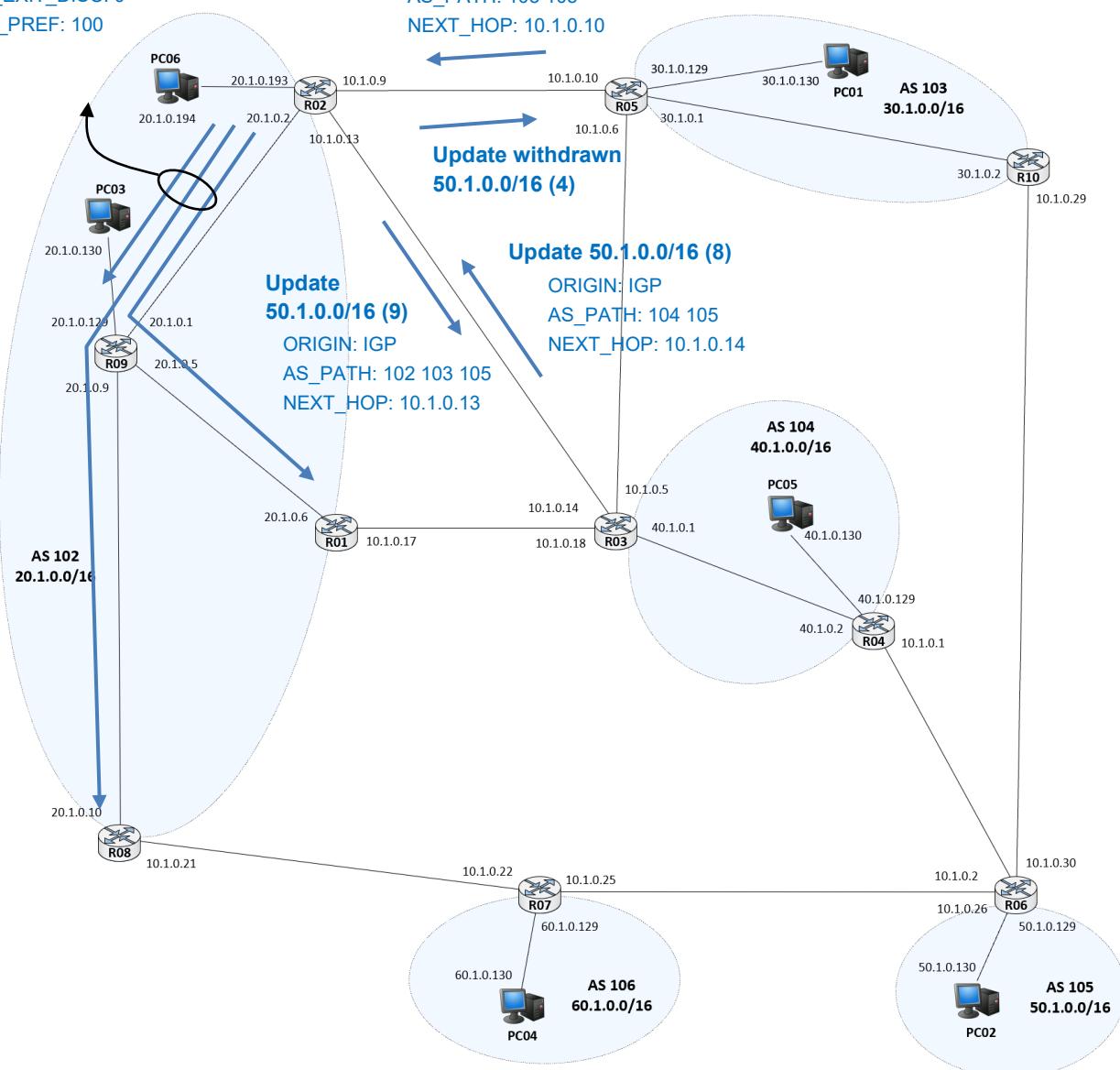
Update 50.1.0.0/16 (3)

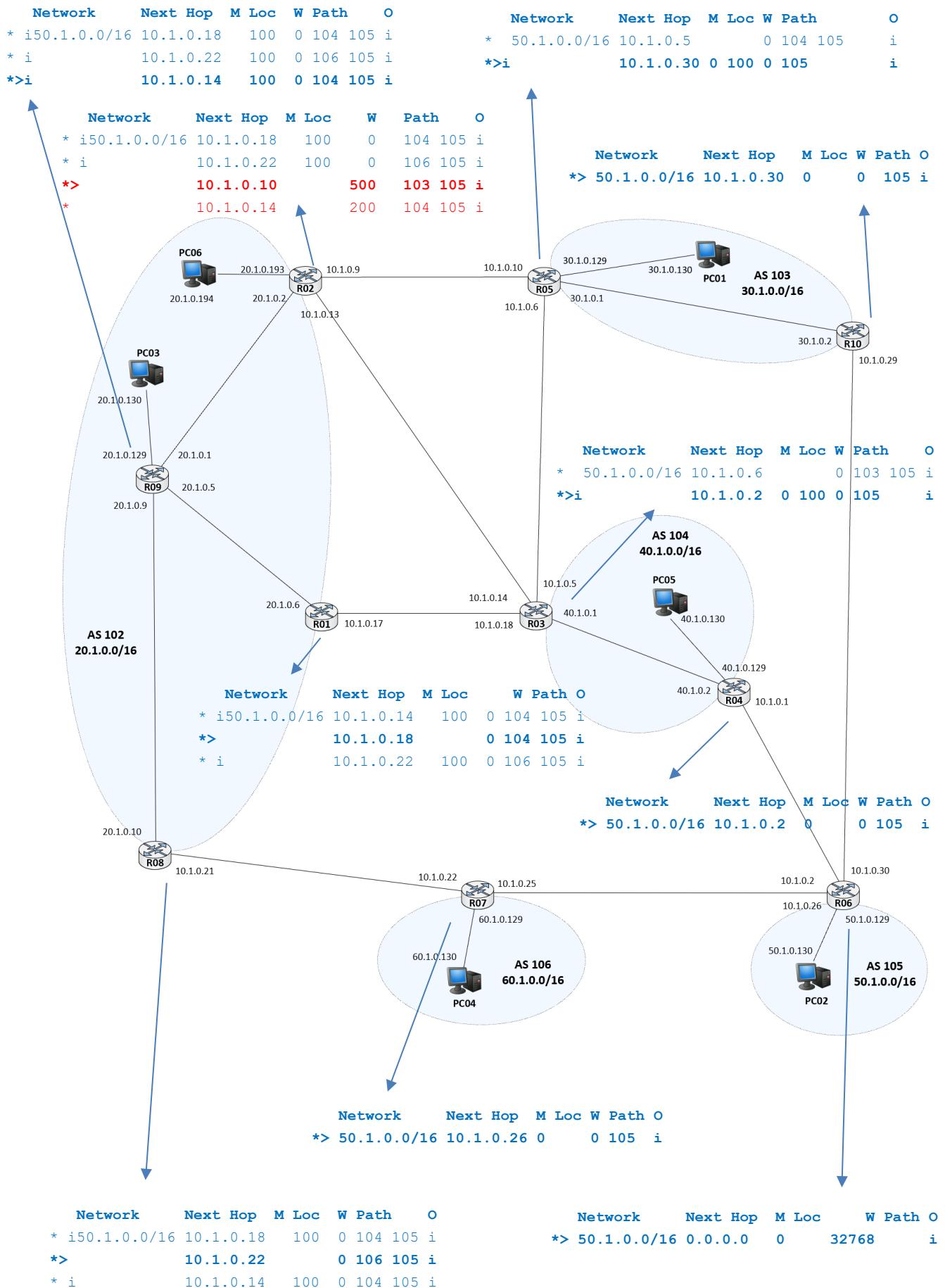
ORIGIN: IGP
AS_PATH: 103 105
NEXT_HOP: 10.1.0.10
MULTI_EXIT_DISC: 0
LOCAL_PREF: 100

Update withdrawn 50.1.0.0/16 (4)

Update 50.1.0.0/16 (8)

ORIGIN: IGP
AS_PATH: 104 105
NEXT_HOP: 10.1.0.14





Observeu la taula BGP de R02: `show ip bgp`

- a. Quantes rutes apareixen per al prefix 50.1.0.0/16? Quina està seleccionada com a millor? Per què?

```
lxc-attach -n R02 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 50.1.0.0/16	10.1.0.18	100	0	104 105 i	
* i	10.1.0.22	100	0	106 105 i	
*	10.1.0.10		0	103 105 i	
*>	10.1.0.14		0	104 105 i	

Després:

* 50.1.0.0/16	10.1.0.18	100	0	104 105 i
* i	10.1.0.22	100	0	106 105 i
*>	10.1.0.10		500	103 105 i
*	10.1.0.14		200	104 105 i

Criteri 2 → S'escull la ruta amb major WEIGHT.

Atureu les captures del Wireshark i observeu-les.

- b. R02 propaga el WEIGHT als seus veïns (eBGP i/o iBGP)?

No, el paràmetre WEIGHT no s'envia mai a cap veí. Mireu el diagrama de la pàgina 98.

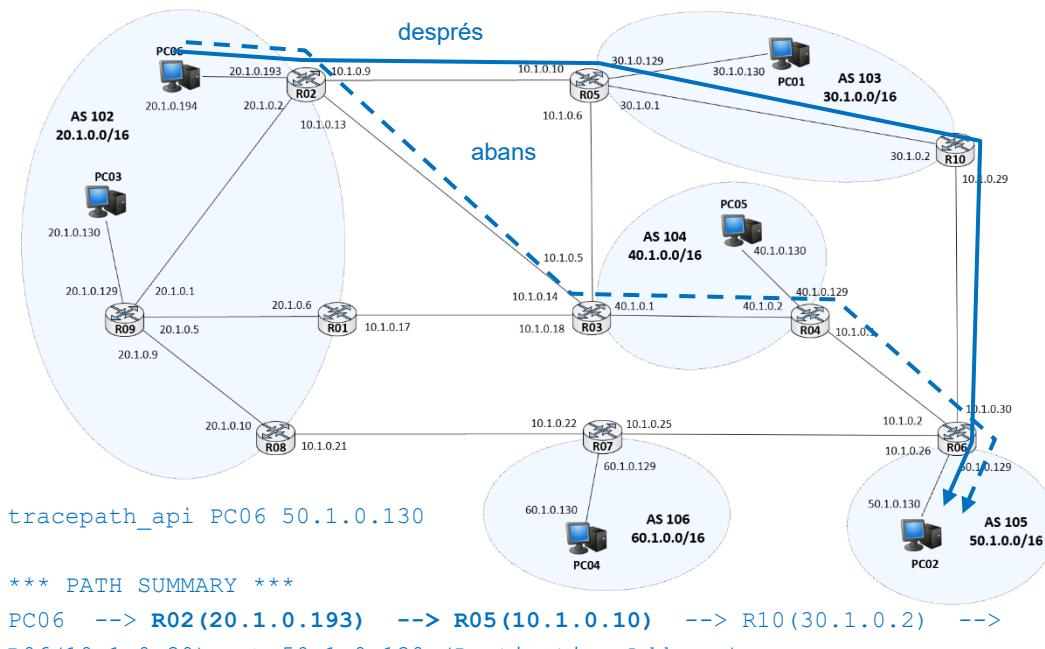
Comproveu també les taules d'encaminament BGP de la resta de routers, excepte els de l'AS 105, i fixeu-vos amb el prefix 50.1.0.0/16.

- c. Hi ha algun canvi en algun router?

No. Mireu el diagrama de la pàgina 99.

Des d'un terminal del PC utilitzeu l'eina `tracepath_api` per saber per on passen els paquets que s'envien des del PC06 al PC02: `tracepath_api PC03 50.1.0.130`

- d. Compareu el resultat amb el que heu obtingut a l'apartat b.



Exercici 5. Ordre de preferència dels atributs

- a. Sense modificar el que heu fet fins ara, quin atribut cal configurar i on per aconseguir que els paquets del PC03 al PC05 surtin de l'AS 102 per R02?

Feu la configuració que calgui on considereu necessari i comproveu que els paquets enviats pel PC03 al PC05 surten de l'AS 102 per R02 amb la comanda: `tracepath_api PC03 40.1.0.130`

Per forçar que el punt de sortida de l'AS 102 per anar al prefix 40.1.0.0/16 sigui R02 caldria incrementar el local-preference que R02 assigna al prefix 40.1.0.0/16 quan aprèn la ruta d'un dels seus veïns eBGP, per exemple, el veí 10.1.0.10 (R05). Per fer-ho, com que R02 ja té un route-map que s'aplica als paquets que arriben del veí 10.1.0.10 (RMAP2) només cal afegir una access-list pel prefix 40.1.0.0/16 i una entrada al route-map RMAP2 (amb número de seqüència menor que 20) per modificar l'atribut LOCAL-PREFERENCE d'aquest prefix, donant-li, per exemple un valor 250.

```
lxc-attach -n R01 -- vtysh

R02# show run
( . . . )
router bgp 102
bgp router-id 10.1.0.9
network 20.1.0.0/16
neighbor 10.1.0.10 remote-as 103
neighbor 10.1.0.10 route-map RMAP2 in
neighbor 10.1.0.14 remote-as 104
neighbor 10.1.0.14 route-map RMAP3 in
neighbor 10.1.0.14 route-map RMAP1 out
neighbor 20.1.0.1 remote-as 102
neighbor 20.1.0.6 remote-as 102
neighbor 20.1.0.10 remote-as 102
!
access-list P20 permit 20.1.0.0/16
access-list P50 permit 50.1.0.0/16
!
route-map RMAP1 permit 10
match ip address P20
set metric 4
!
route-map RMAP1 permit 20
!
route-map RMAP2 permit 10
match ip address P50
set weight 500
!
route-map RMAP2 permit 20
!
route-map RMAP3 permit 10
match ip address P50
set weight 200
!
route-map RMAP3 permit 20
!
( . . . )
```

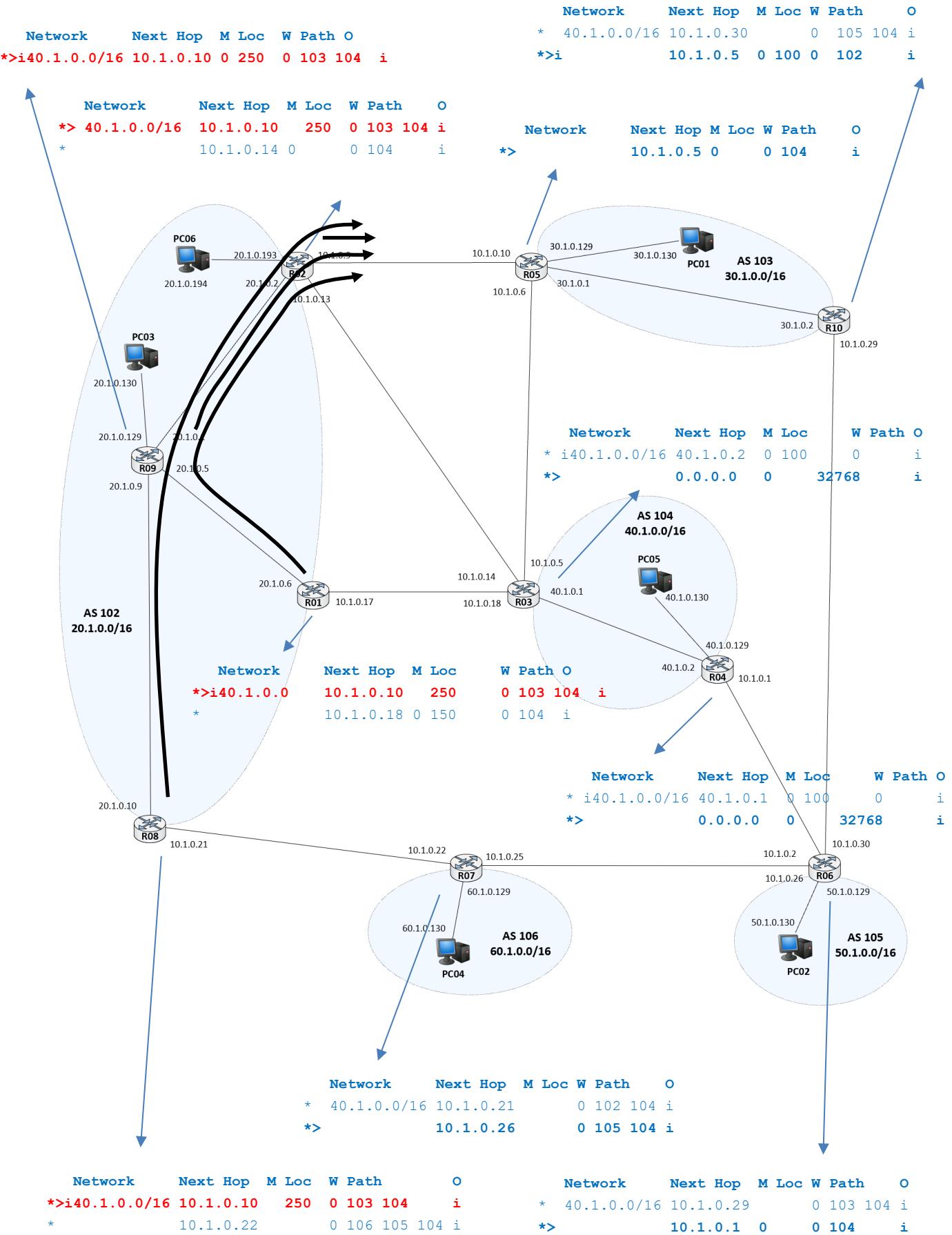
```

R02# configure terminal
R02(config)# access-list P40 permit 40.1.0.0/16
R02(config)# route-map RMAP2 permit 15
R02(config-route-map)# match ip address P40
R02(config-route-map)# set localpref 250
R02(config-route-map)# exit
R02(config)# exit
R02# show run
( . . . )
router bgp 102
  bgp router-id 10.1.0.9
  network 20.1.0.0/16
  neighbor 10.1.0.10 remote-as 103
  neighbor 10.1.0.10 route-map RMAP2 in
  neighbor 10.1.0.14 remote-as 104
  neighbor 10.1.0.14 route-map FILTER3 in
  neighbor 10.1.0.14 route-map FILTER1 out
  neighbor 20.1.0.1 remote-as 102
  neighbor 20.1.0.6 remote-as 102
  neighbor 20.1.0.10 remote-as 102
!
access-list P20 permit 20.1.0.0/16
access-list P50 permit 50.1.0.0/16
access-list P40 permit 40.1.0.0/16
!
route-map RMAP1 permit 10
  match ip address P20
  set metric 4
!
route-map RMAP1 permit 20
!
route-map RMAP2 permit 10
  match ip address P50
  set weight 500
!
route-map RMAP2 permit 15
  match ip address P40
  set localpreference 250
!
route-map RMAP2 permit 20
!
route-map RMAP3 permit 10
  match ip address P50
  set weight 200
!
route-map RMAP3 permit 20
!
( . . . )

```

Per tal que es pugui aplicar la nova política, cal forçar que el veí 10.1.0.10 torni a enviar les seves best a R02.

```
R02# clear ip bgp 10.1.0.10 in
```



```
lxc-attach -n R01 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
*> 40.1.0.0/16	10.1.0.18	0	150	0	104 i

Després:

*>i40.1.0.0/16	10.1.0.10		250	0	103 104 i
*	10.1.0.18	0	150	0	104 i

Criteri 3 → Guanya la ruta amb major LOCAL-PREFERENCE.

```
lxc-attach -n R02 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i40.1.0.0/16	10.1.0.18	0	150	0	104 i
*	10.1.0.10			0	103 104 i
*	10.1.0.14	0		0	104 i

Després:

*> 40.1.0.0/16	10.1.0.10		250	0	103 104 i
*	10.1.0.14	0		0	104 i

Criteri 3 → Guanya la ruta amb major LOCAL-PREFERENCE. La ruta amb next-hop 10.1.0.18 apresa de R01, s'ha esborrat perquè R01 ha enviat un Update amb un withdrawn d'aquest prefix i s'ha après la ruta amb nexthop 10.1.0.10 que anuncia R02 amb LOCAL-PREFERENCE 250.

```
lxc-attach -n R08 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i40.1.0.0/16	10.1.0.18	0	150	0	104 i
*	10.1.0.22			0	106 105 104 i

Després:

*>i40.1.0.0/16	10.1.0.10	0	250	0	103 104 i
*	10.1.0.22			0	106 105 104 i

Criteri 3 → Guanya la ruta amb major LOCAL-PREFERENCE. La ruta amb next-hop 10.1.0.18 apresa de R01, s'ha esborrat perquè R01 ha enviat un Update amb un withdrawn d'aquest prefix i s'ha après la ruta amb nexthop 10.1.0.10 que anuncia R02 amb LOCAL-PREFERENCE 250.

```
lxc-attach -n R09 -- vtysh -c 'show ip bgp'
```

Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
*>i40.1.0.0/16	10.1.0.18	0	150	0	104 i

Després:

*>i40.1.0.0/16	10.1.0.10	0	250	0	103 104 i
----------------	-----------	---	-----	---	-----------

Només té una ruta per anar al prefix 40.1.0.0/16. La ruta amb next-hop 10.1.0.18 apresa de R01, s'ha esborrat perquè R01 ha enviat un Update amb un withdrawn d'aquest prefix i s'ha après la ruta amb nexthop 10.1.0.10 que anuncia R02 amb LOCAL-PREFERENCE 250.

```
lxc-attach -n R05 -- vtysh -c 'show ip bgp'
```

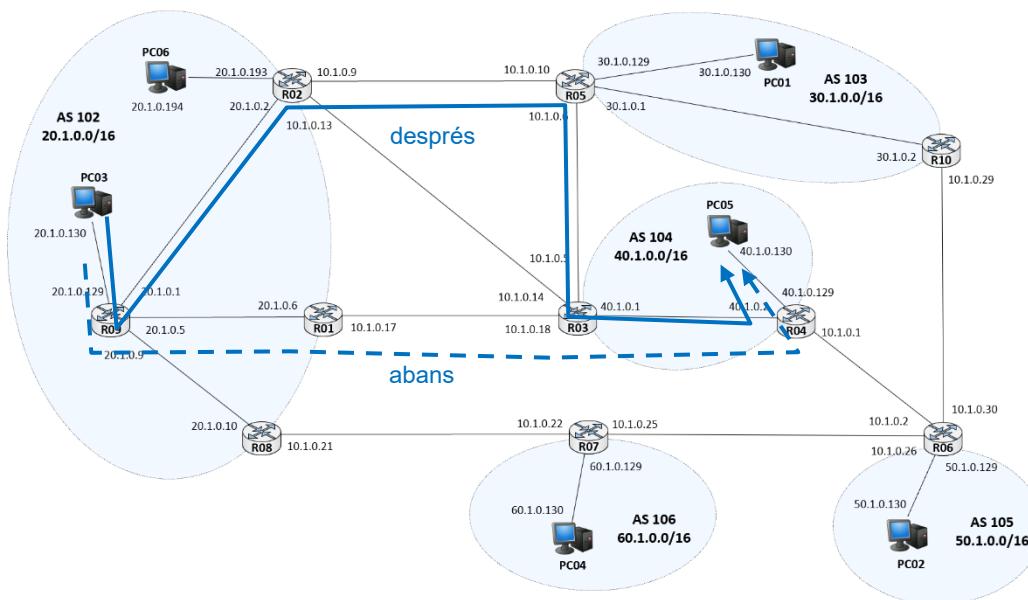
Abans:

Network	Next Hop	Metric	LocPrf	Weight	Path
* 40.1.0.0/16	10.1.0.9			0	102 104 i
*>	10.1.0.5	0		0	104 i

Després:

*> 40.1.0.0/16	10.1.0.5	0	0	104 i
----------------	----------	---	---	-------

Només té una ruta per anar al prefix 40.1.0.0/16. La ruta amb next-hop 10.1.0.9 apresa de R02, s'ha esborrat perquè R02 ha enviat un Update amb un withdrawn d'aquest prefix.



```
tracepath_api PC03 40.1.0.130
```

Abans:

```
*** PATH SUMMARY ***
PC03 --> R09(20.1.0.129) --> R01(20.1.0.6) --> R03(10.1.0.18) --> R04(40.1.0.2)
--> 40.1.0.130 (Destination Address)
```

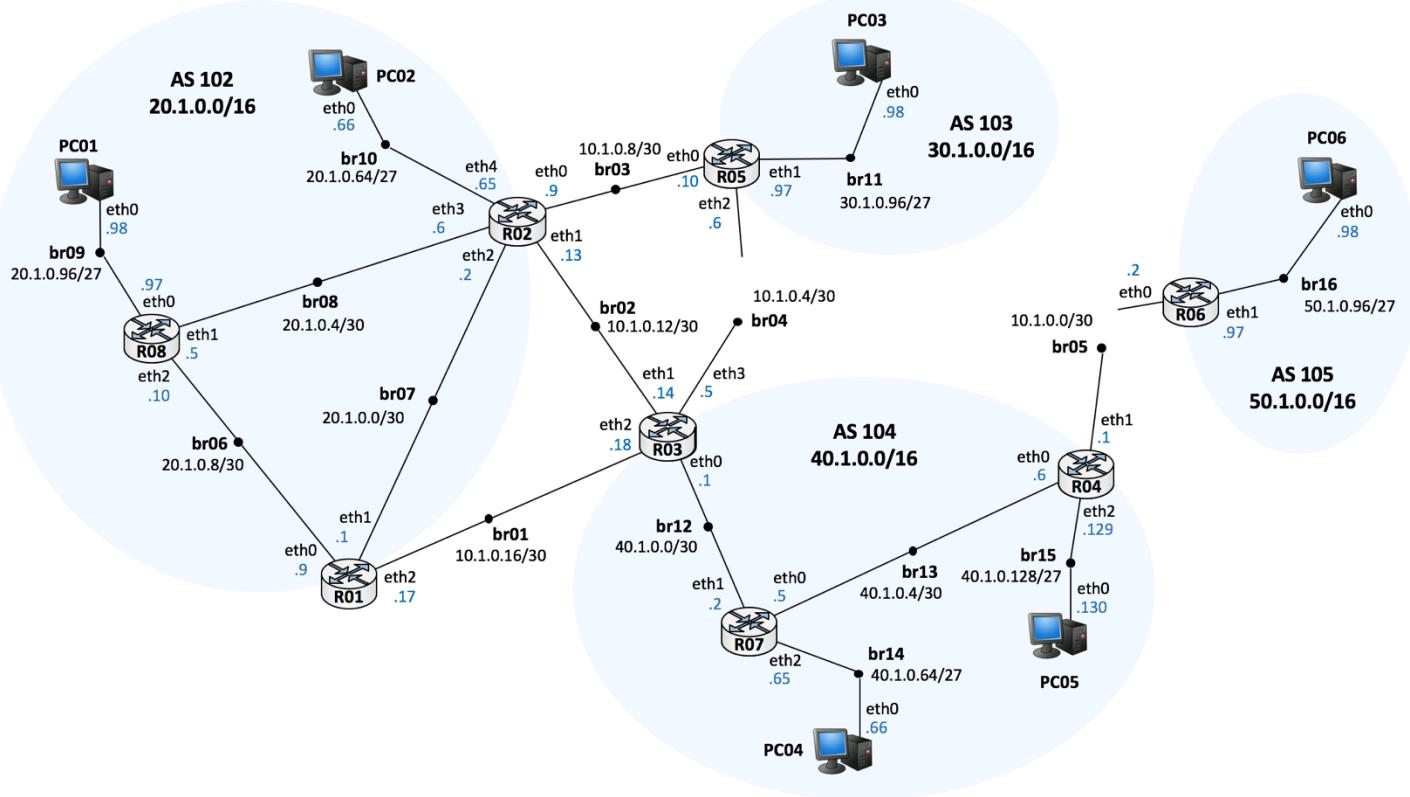
Després:

```
*** PATH SUMMARY ***
PC06 --> R09(20.1.0.129) --> R02(20.1.0.2) --> R05(10.1.0.10) --> R03(10.1.0.5)
--> R04(10.1.0.2) --> 40.1.0.130 (Destination Address)
```

Atureu l'escenari de la segona part amb la comanda: P05-E02-stop

FIGURES

ESCENARI PART I



ESCENARI PART II

