* [Fejezet száma 5 Egyterületű OSPF protokoll konfigurálása és ellenőrzése](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#5)
* [5.0 Egyterületű OSPF protokoll konfigurálása és ellenőrzése](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#5.0)
* [5.0.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#5.0.1)
* [5.0.1.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#5.0.1.1)

# Bevezetés

Az OSPF (Open Shortest Path First, legrövidebb út protokoll) egy gyakran alkalmazott kapcsolatállapot alapú irányító protokoll, amely többféle módon finomhangolható. A leggyakrabban használt finomhangolási módszerek: a kijelölt és a tartalék kijelölt forgalomirányító (Designated Router/Backup Router, DR/BR) választás befolyásolása, az alapértelmezett útvonalak terjesztése, az OSPFv2 és OSPFv3 interfészek finomhangolása és a hitelesítés engedélyezése.

Ez az OSPF-ről szóló fejezet a finomhangolási lehetőségeket ismerteti, vagyis ezen funkciók IPv4 és IPv6 alatti megvalósításához szükséges konfigurációs módbeli parancsokat, valamint az OSPFv2 és az OSPFv3 protokollok hibajavításához használt összetevőket és parancsokat.

# Csoportos feladat - DR és BDR választás

**DR és BDR választás**

Döntést szeretnénk hozni arról, hogy az OSPF hálózatunkban miként befolyásolhatjuk a kijelölt (DR) és a tartalék kijelölt (BDR) forgalomirányítók választását. A feladat ezt a döntési folyamatot mutatja be.

Három különböző kijelölt forgalomirányító választási esetet mutatunk be. A középpontban a DR és BDR választás áll. További információk a gyakorlathoz tartozó PDF dokumentumban találhatók.

Ha marad rá idő, akkor két csoport közösen is szimulálhatja a DR és BDR választást.

[Csoportos feladat - DR és BDR választás](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/5.0.1.2%20DR%20and%20BDR%20Elections%20Instructions.pdf)

**A forgalomirányítás és a kapcsolás összehasonlítása**

Egy méretezhető hálózat hierarchikus hálózattervezést igényel. A korábbi fejezetek a hozzáférési- és elosztási rétegekkel foglalkoztak. Az 1. ábrán látható, hogy a 2. rétegbeli kapcsolás, az útvonal összevonás, a LAN redundancia és a vezetéknélküli LAN-ok olyan technológiák, amelyek a hálózati erőforrásokhoz való hozzáférést biztosítják.

A méretezhető hálózatok esetében is szükség van a telephelyek közötti optimális elérhetőségre. A távoli hálózatok elérhetőségét az elosztási és központi rétegben működő forgalomirányítók és 3. rétegbeli kapcsolók biztosítják, ez látható a 2. ábrán. A forgalomirányítók és a 3. rétegbeli kapcsolók a távoli hálózatokat az alábbi két mód egyikével tanulják meg:

* **Kézzel megadva** - A távoli hálózatok statikus útvonalként, manuálisan megadva kerülnek az irányítótáblába.
* **Dinamikusan** - A távoli útvonalak megtanulása automatikusan történik dinamikus irányító protokoll segítségével, mint például a továbbfejlesztett belső átjáró protokoll (Enhanced Interior Gateway Routing Protocol, EIGRP) vagy a legrövidebb út protokoll (Open Shortest Path First, OSPF).

**Statikus forgalomirányítás**

**Statikus forgalomirányítás**

Az ábrán egy statikus forgalomirányítási példa látható. A hálózati rendszergazda manuálisan is konfigurálhat egy adott hálózathoz statikus útvonalat. A dinamikus forgalomirányító protokollok működésével ellentétben a statikus útvonalak a hálózati topológia változásakor nem frissülnek automatikusan, minden esetben kézzel kell őket újra beállítani. A statikus útvonalak mindaddig nem változnak, amíg a rendszergazda újra nem konfigurálja őket.

A statikus forgalomirányítás három alapvető felhasználási területe:

* Olyan kisméretű hálózatok esetén, amelyek várhatóan nem növekednek meg jelentősen, így az irányítótáblák karbantartása egyszerűen elvégezhető.
* Forgalomirányítás egy véghálózat irányába és vissza. A véghálózat olyan hálózat, ahonnan csak egy út vezet az egyetlen szomszéd forgalomirányító felé.
* Egyszerű alapértelmezett útvonal használata olyan hálózatok irányába, amelyek felé nincs az irányítótáblában egyetlen másik útvonal sem. Az alapértelmezett útvonalak segítségével a forgalom bármilyen, a szomszédos forgalomirányítón túli célhálózat felé továbbítható.

# Dinamikus forgalomirányító protokollok

**Dinamikus forgalomirányítás**

Ahogy az ábrán is látható az irányító protokollok teszik lehetővé a forgalomirányítók számára, hogy dinamikusan információkat osszanak meg távoli hálózatokról. A forgalomirányítók a hozzájuk érkező frissítések információit automatikusan az irányítótáblájukhoz adják, majd az irányító protokollok meghatározzák a legjobb útvonalat minden hálózat felé. A dinamikus irányító protokollok legfontosabb előnye, hogy a topológiában bekövetkező változáskor a forgalomirányítók információt cserélnek egymással. Ez az információcsere teszi lehetővé a számukra, hogy automatikusan értesüljenek az új hálózatokról vagy egy meglévő hálózathoz vezető kapcsolat hibája esetén alternatív útvonalat találjanak.

A statikus forgalomirányításhoz képest a dinamikus irányító protokollok kevesebb adminisztrációt igényelnek. Ugyanakkor használatuk költségének egy részét a protokoll működéséhez szükséges forgalomirányító erőforrások teszik ki, beleértve a CPU-időt és a hálózati kapcsolatok sávszélességét. A dinamikus forgalomirányítás előnyei ellenére a statikus forgalomirányításnak még mindig megvan a létjogosultsága. Vannak esetek, amikor a statikus forgalomirányítás a megfelelőbb, míg máskor a dinamikus a jobb választás. Mindemellett fontos megérteni azt, hogy a statikus és dinamikus forgalomirányítás alkalmazása nem zárja ki egymást. Sőt a legtöbb esetben a hálózatok egyszerre mindkettőt alkalmazzák.

A két leggyakrabban használt dinamikus irányító protokoll az EIGRP és az OSPF. A fejezet a továbbiakban az OSPF protokollal foglalkozik.

**MEGJEGYZÉS**: Minden dinamikus irányító protokoll képes az irányítótáblájában lévő statikus útvonalak hirdetésére és terjesztésére.

# Legrövidebb utat először (Open Shortest Path First, OSPF)

Az OSPF egy gyakran alkalmazott kapcsolatállapot alapú irányító protokoll, amelyet a távolságvektor alapú RIP irányító protokoll leváltására fejlesztettek ki. A RIP-protokollal szemben az OSPF lényeges előnye, hogy nagyobb hálózatok esetében gyorsabb konvergenciát és méretezhetőséget biztosít.

Az ábrán az OSPF jellemzői láthatók:

* **Osztály nélküli** - Tervezéséből eredően osztály nélküli, azaz támogatja a VLSM-et és a CIDR-t.
* **Hatékony** - A forgalomirányítás változásai váltják ki az útvonal frissítéseket (nincsenek periodikus frissítések). A legjobb útvonal kiválasztásához az SPF-algoritmust használja.
* **Gyors konvergenciát biztosít** - A hálózat változásai gyorsan elterjednek.
* **Skálázható** - Egyaránt jól működik kis - és nagy hálózatok esetében is. Hierarchikus rendszerek kialakításához a forgalomirányítók csoportokba szervezhetők.
* **Biztonságos** - Támogatja a Message Digest 5 (MD5) alapú hitelesítést. MD5 alkalmazásakor az OSPF forgalomirányítók a társaiktól csak olyan kódolt irányítási frissítéseket fogadnak el, ahol megegyezik az előre megosztott kulcs.

# Egyterületű OSPF-protokoll konfigurálása

Bár a fejezet az egyterületű OSPF finomhangolásával és hibajavításával foglalkozik, azért hasznos lehet az OSPF-protokoll alapvető beállatásainak áttekintése is.

Az 1. ábrán az OSPFv2 konfigurációjához használt topológia látható. A topológia forgalomirányítói olyan alapkonfigurációval rendelkeznek, amely tartalmazza az engedélyezett interfész címeket. Pillanatnyilag egyik forgalomirányítón sincs statikus vagy dinamikus forgalomirányítás konfigurálva. Az R1, R2 és R3 forgalomirányítók valamennyi interfésze (kivéve az R2 loopback) az OSPF gerinc (backbone) területéhez tartozik. Az ISP forgalomirányító az adott irányítási terület átjárója az internet felé.

A 2. ábrán látható, hogy az R1 forgalomirányító GigabitEthernet 0/0 interfésze a valós 1.000.000 kbps (1.000.000.000 bps) sávszélességre van beállítva. Valamint láthatóak még az OSPF forgalomirányító konfigurációs mód beállításai: a router ID, az interfészek referencia sávszélessége a gyors interfészek megfelelő figyelembe vételéhez lett beállítva, és az R1-hez csatlakozó három hálózat hirdetése is. Figyeljük meg, hogy miként használható a helyettesítő maszk az adott hálózatok megadásához.

A 3. ábrán látható, hogy az R2 forgalomirányító GigabitEthernet 0/0 interfészén ugyancsak a valós sávszélesség lett beállítva, továbbá a router ID, az interfészek referencia sávszélessége a gyors interfészek megfelelő figyelembe vételéhez lett beállítva, és az R2-höz csatlakozó három hálózat hirdetése is beállításra kerültek. Figyeljük meg hogyan lehet a helyettesítő maszk használata nélkül csupa nulla maszk megadásával azonosítani a forgalomirányító interfészeit. Így az OSPF hatékonyan tudja a forgalomirányító interfészéhez rendelt alhálózati maszkot a hirdetett hálózat maszkjaként alkalmazni.

A 4. ábra parancsszimulátorában végezzük el a következő feladatokat: állítsuk be az R3 G0/0 interfészének sávszélességét, lépjünk OSPF konfigurációs módba, adjuk meg a megfelelő forgalomirányító azonosítót (router ID), állítsuk be a referencia sávszélességet, valamint a forgalomirányító interfészek és a 0.0.0.0 helyettesítő maszk segítségével hirdessük a három közvetlenül csatlakozó hálózatot.

Figyeljük meg a megjelenő információs üzenetet arról, hogy az R3 teljes értékű szomszédsági viszonyt létesített az 1.1.1.1 azonosítójú R1 forgalomirányítóval és a 2.2.2.2 azonosítójú R2 forgalomirányítóval. Így az OSPF-hálózat konvergáltnak tekinthető.

**Egyterületű OSPF protokoll ellenőrzése**

Az OSPF ellenőrzésére szolgáló hasznos parancsok:

* **show ip ospf neighbor** - A paranccsal a szomszédsági viszonyok kialakítása ellenőrizhető. Ha a szomszédos forgalomirányító azonosítója nem jelenik meg, vagy az nincs FULL állapotban, akkor a két forgalomirányító nincs egymással OSPF szomszédsági viszonyban.
* **show ip protocols** - A paranccsal gyorsan ellenőrizhetők az alapvető OSPF beállítások. Ezek a következők: OSPF folyamatazonosító (process ID), forgalomirányító azonosító (router ID), a forgalomirányító által hirdetett hálózatok, a szomszédok, akiktől a forgalomirányító frissítéseket kap és az alapértelmezett adminisztratív távolság, amely OSPF esetén 110.
* **show ip ospf** - A parancs megjeleníti az OSPF folyamatazonosítót (process ID), a forgalomirányító azonosítót (router ID), valamint az OSPF SPF és terület információkat.
* **show ip ospf interface** - A parancs az összes olyan interfész részletes információit megjeleníti, amelyeken az OSPF engedélyezett. Továbbá a használatával az is meghatározható, hogy a **network** parancs helyesen lett-e megadva.
* **show ip ospf interface brief** - A parancs az OSPF-interfészek állapotát és összefoglaló információit jeleníti meg.

Az 1-5. ábrán a felsorolt ellenőrző parancsok kimenete látható az R1 forgalomiránytón.

A 6. ábrán a parancsszimulátorral ellenőrizhetjük az R2 forgalomirányítón a szomszédsági viszonyokat, az alapvető OSPF konfigurációs paramétereket és az OSPF-interfészek összefoglaló adatait.

A 7. ábrán a parancsszimulátorral ellenőrizhetjük az R3 forgalomirányítón a szomszédsági viszonyokat, az alapvető OSPF konfigurációs paramétereket és az OSPF-interfészek összefoglaló adatait.

# Egyterületű OSPFv3-protokoll konfigurálása

Az alábbiakban az IPv6 hálózatokban használt OSPFv3 protokoll alapvető beállításait tekintjük át.

Az 1. ábrán az OSPFv3 konfigurálásához használt topológia látható. A topológia forgalomirányítói olyan alapkonfigurációval rendelkeznek, amelyek tartalmazzák az engedélyezett interfészek IPv6-címeit. Pillanatnyilag egyik forgalomirányítón sincs statikus vagy dinamikus forgalomirányítás konfigurálva. Az R1, R2 és R3 valamennyi interfésze (kivéve R2 loopback interfészét) az OSPF gerinc (backbone) területéhez tartozik.

A 2. ábra az R1 forgalomirányítón a router ID és a gyors interfészek megfelelő figyelembe vételéhez szükséges referencia sávszélesség beállítását mutatja OSPFv3 forgalomirányító konfigurációs módban. Ezt követően az OSPFv3 irányításban résztvevő interfészek konfigurálása látható. A Gigabit Ethernet 0/0 interfészen beállított sávszélesség a valós értéket tükrözi. Figyeljük meg, hogy OSPFv3 esetében nincs szükség helyettesítő maszk megadására.

A 3. ábra az R2 forgalomirányítón a router ID és a gyors interfészek megfelelő figyelembe vételéhez szükséges referencia sávszélesség beállítását mutatja OSPFv3 forgalomirányító konfigurációs módban. Ezt követően az OSPFv3 irányításban résztvevő interfészek konfigurálása látható. A Gigabit Ethernet 0/0 interfészen beállított sávszélesség a valós értéket tükrözi.

A 4. ábra parancsszimulátorában manuálisan állítsuk be a forgalomirányító azonosítót és adjuk meg a referencia sávszélességet. Ezt követően konfiguráljuk az interfészeket, kezdve a Gigabit Ethernet 0/0 interfésszel. Ezen az interfészen a valós sávszélességet is adjuk meg.

Figyeljük meg a megjelenő információs üzenetet arról, hogy az R3 teljes értékű szomszédsági viszonyt létesített az 1.1.1.1 azonosítójú R1 forgalomirányítóval és a 2.2.2.2 azonosítójú R2 forgalomirányítóval. Így az OSPF-hálózat konvergáltnak tekinthető.

**Egyterületű OSPFv3-protokoll ellenőrzése**

Az OSPFv3 ellenőrzésére szolgáló hasznos parancsok:

* **show ipv6 ospf neighbor** - A paranccsal a szomszédsági viszonyok kialakítása ellenőrizhető. Ha a szomszédos forgalomirányító azonosítója nem jelenik meg, vagy nem FULL állapotúnak látszik, akkor a két forgalomirányító nem alakított ki OSPF szomszédsági viszonyt egymással.
* **show ipv6 protocols** - A paranccsal gyorsan ellenőrizhető az alapvető OSPFv3 konfiguráció, beleértve az OSPF folyamatazonosítót, a forgalomirányító azonosítót és az OSPFv3 engedélyezett interfészeket.
* **show ipv6 route ospf** - A parancs az irányítótábla OSPFv3 specifikus útvonalait mutatja.
* **show ipv6 ospf interface brief** - A parancs az OSPFv3 interfészek állapotát és összefoglaló információit jeleníti meg.

Az 1-4. ábrákon az R1 forgalomirányítón kiadott ellenőrző parancsoknak megfelelő kimenetek láthatók.

# Laborgyakorlat - Egyterületű OSPFv2 alapvető konfigurálása

**Ebben a gyakorlatban a következő feladatokat végezzük el:**

* 1. rész: A hálózat felépítése és az eszközök alapvető konfigurálása.
* 2. rész: OSPF forgalomirányítás beállítása és ellenőrzése.
* 3. rész: Forgalomirányító azonosítók kiosztásának megváltoztatása.
* 4. rész: OSPF passzív interfészek konfigurálása.
* 5. rész: OSPF mértékek megváltoztatása.

[Laborgyakorlat - Configuring Basic Single-Area OSPFv2](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/5.1.1.9%20Lab%20-%20Configuring%20Basic%20Single-Area%20OSPFv2.pdf)

**OSPF hálózattípusok**

Az OSPF konfigurálását kezdjük az OSPF protokoll egy alapvető megvalósításával.

Az 1-5. ábrákon az OSPF protokoll által definiált alábbi öt hálózattípus látható:

* **Pont-pont** - Két forgalomirányító kapcsolódik egymáshoz egy közös összeköttetésen keresztül. Más forgalomirányító nincs ezen a hálózaton. Gyakran használják WAN-összeköttetéseknél. (1. ábra)
* **Szórásos többes hozzáférés (Broadcast multiaccess)** - Több forgalomirányító kapcsolódik egymáshoz egy Ethernet hálózaton keresztül. (2. ábra)
* **Nem szórásos többes hozzáférés (Nonbroadcast multiaccess, NBMA)** - Több forgalomirányító kapcsolódik egy hálózathoz, amely nem engedélyezi az üzenetszórást. Ilyen hálózat például a Frame Relay. (3. ábra)
* **Pont-többpont** - Több forgalomirányító kapcsolódik össze csillagponti (hub-and-spoke) topológiában egy NBMA hálózat felett. Gyakran használják a távoli telephelyek (spokes) központi telephellyel (hub) való összekötésére. (4. ábra)
* **Virtuális összeköttetések** - Speciális OSPF hálózat, amely távoli OSPF területeket kapcsol össze a gerinc területtel (backbone area). (5. ábra)

Egy többes hozzáférésű hálózatban több eszköz használja ugyanazt az osztott átviteli közeget a kommunikációra. Az Ethernet hálózat a leggyakrabban alkalmazott többes hozzáférésű hálózat. A szórásos hálózatokban minden csatlakoztatott eszköz látja a szórásos és csoportos címzésű kereteket. Ezek többes hozzáférésű hálózatok, mivel számos állomás, nyomtató, forgalomirányító vagy más eszköz lehet ugyanazon a hálózaton.

**A többes hozzáférésű hálózatok kihívásai**

A többes hozzáférésű hálózatok az OSPF LSA elárasztás szempontjából két kihívást is jelentenek:

* **Többszörös szomszédsági viszonyok létrehozása** – Az Ethernet hálózatok több OSPF forgalomirányítót is összekapcsolhatnak egyetlen összeköttetésen. Ebben az esetben minden szomszédos forgalomirányítóval szomszédsági viszonyt kialakítani szükségtelen és nem is célszerű. Hiszen ez ugyanazon hálózathoz kapcsolódó forgalomirányítók közötti túlzott mennyiségű LSA cserét eredményezne.
* **Túlzott LSA elárasztás** – A kapcsolatállapot alapú forgalomirányítók az OSPF indulásakor vagy a topológia megváltozása esetén elárasztással küldik el kapcsolatállapot üzeneteiket. Ez az elárasztás túlzottá válhat.

Az alábbi képlet segítségével számolhatjuk ki a szükséges szomszédsági kapcsolatok számát. Többes hozzáférésű hálózatban a szükséges szomszédsági kapcsolatok száma tetszőleges számú forgalomirányító (számukat jelölje *n*) esetén:

*n* (*n* – 1) / 2

Az 1. ábra egy négy forgalomirányítóból álló egyszerű topológiát mutat, melyben a forgalomirányítók ugyanazon többes hozzáférésű Ethernet hálózathoz kapcsolódnak. A 2. ábrán is látható, hogy bármely, a szomszédsági viszonyok számát csökkentő megoldás nélkül összesen 6 szomszédsági viszony alakulna ki közöttük: 4 (4 - 1) / 2 = 6. A 3. ábrán jól látható, hogy ha újabb forgalomirányítókat adnánk a hálózathoz, akkor a szomszédsági viszonyok száma drasztikusan megnőne.

# OSPF kijelölt forgalomirányító (designated router, DR)

Többes hozzáférésű hálózatokban a szomszédsági viszonyok számának és az LSA elárasztás problémájának a kezelésére nyújt megoldást a kijelölt forgalomirányító (DR). Többes hozzáférésű hálózat esetén az OSPF egy DR-t választ, hogy az gyűjtési és elosztási pontként kezelje a küldött és fogadott LSA-kat. A DR meghibásodásának esetére egy BDR-t is választanak. A BDR passzívan figyeli az üzenetváltásokat és kapcsolatot tart fenn minden forgalomirányítóval. Ha a DR nem küld Hello üzeneteket, akkor a BDR veszi át a szerepét.

Minden további nem DR és nem BDR forgalomirányítót DROTHER típusúnak tekintünk.

Az 1. ábrán az R1 forgalomirányító lett a kijelölt forgalomirányító (DR) az R2, R3 és R4 forgalomirányítókat összekötő Ethernet hálózatban. Figyeljük meg, hogy a szomszédsági viszonyok száma lecsökkent 3-ra.

A többes hozzáférésű hálózatokban a forgalomirányítók egy kijelölt forgalomirányítót (DR) és egy tartalék kijelölt forgalomirányítót (BDR) választanak. A DROTHER forgalomirányítók csak a DR és a BDR forgalomirányítóval alakítanak ki szomszédsági viszonyt. A DROTHER forgalomirányítók így nem árasztják el a hálózat minden forgalomirányítóját az LSA üzeneteikkel, hanem azokat csak a DR és BDR forgalomirányítóknak küldik a 224.0.0.6 csoportos címre.

Kattintsunk a 2. ábra Lejátszás gombjára a DR szerepéről készült animáció megtekintéséhez! Az animációban az R1 forgalomirányító LSA üzeneteket küld a DR-nek, a BDR pedig figyel. A DR felelős az R1 által küldött LSA üzenetek továbbításáért a többi forgalomirányítóhoz. A DR a 224.0.0.5 (minden OSPF forgalomirányító) csoportos címet használja. Ennek eredményeként a többes hozzáférésű hálózatban csak egyetlen forgalomirányító kezeli az LSA elárasztást.

**MEGJEGYZÉS**: DR/BDR választás csak többes hozzáférésű hálózatokban történik, pont-pont hálózatokban nem.

**A DR/BDR szerepek ellenőrzése**

Az 1. ábra többes hozzáférésű hálózatában három forgalomirányító kapcsolódik egymáshoz a 192.168.1.0/28-as Ethernet hálózaton keresztül. A forgalomirányítók Gigabit Ethernet 0/0 interfészein a jelzett IP-címek vannak beállítva.

Mivel a forgalomirányítók egy közös többes hozzáférésű, szórásos hálózaton csatlakoznak, az OSPF automatikusan DR-t és BDR-t választ. A példában az R3 forgalomirányító lesz a DR, mivel az ő azonosítója (3.3.3.3) a legnagyobb a hálózatban. Emellett az R2 lesz a BDR, mert az övé a második legnagyobb azonosító a hálózatban.

A forgalomirányító szerepének ellenőrzése a **show ip ospf interface** paranccsal végezhető el (lásd 2. ábra). R1 kimenete alapján megállapítható, hogy :

* R1 nem DR és nem BDR, hanem DROTHER alapértelmezett 1-es prioritással. (1)
* A 3.3.3.3 azonosítójú R3 forgalomirányító a DR a 192.168.1.3 IP-címével, a 2.2.2.2 azonosítójú R2 pedig a BDR a 192.168.1.2 IP-címével. (2)
* R1 két forgalomirányítóval alakított ki szomszédsági viszonyt; a BDR-rel és a DR-rel. (3)

A 3. ábrán R2 kimenete alapján megállapítható, hogy:

* R2 a BDR alapértelmezett 1-es prioritással. (1)
* A 3.3.3.3 azonosítójú R3 forgalomirányító a DR a 192.168.1.3 IP-címével, a 2.2.2.2 azonosítójú R2 pedig a BDR a 192.168.1.2 IP-címével. (2)
* R2 két forgalomirányítóval alakított ki szomszédsági viszonyt; az 1.1.1.1 azonosítójú (R1) forgalomirányítóval és a DR-rel. (3)

A 4. ábrán R3 kimenete alapján megállapítható, hogy:

* R3 a DR alapértelmezett 1-es prioritással. (1)
* A 3.3.3.3 azonosítójú R3 forgalomirányító a DR a 192.168.1.3 IP-címével, a 2.2.2.2 azonosítójú R2 pedig a BDR a 192.168.1.2 IP-címével. (2)
* R3 két forgalomirányítóval alakított ki szomszédsági viszonyt; az 1.1.1.1 azonosítójú (R1) forgalomirányítóval és a BDR-rel. (3)

**A DR/BDR viszonyok ellenőrzése**

Az OSPF viszonyok ellenőrzésére a **show ip ospf neighbor** parancs használható az 1. ábrán látható módon.

A soros összeköttetésekkel ellentétben, ahol a link állapota csak FULL/- lehet, a többes hozzáférésű hálózatokban a szomszédok a következő állapotokkal rendelkezhetnek:

* FULL/DROTHER - Egy DR vagy BDR forgalomirányító teljes értékű szomszédsági viszonyban van egy nem DR és nem BDR forgalomirányítóval. A szomszédok Hello üzeneteket, frissítéseket, kéréseket, válaszokat és nyugtákat cserélhetnek egymással.
* FULL/DR - A forgalomirányító teljes értékű szomszédsági viszonyban van a jelzett DR szomszéddal. A szomszédok Hello üzeneteket, frissítéseket, kéréseket, válaszokat és nyugtákat cserélhetnek egymással.
* FULL/BDR - A forgalomirányító teljes értékű szomszédsági viszonyban van a jelzett BDR szomszéddal. A szomszédok Hello üzeneteket, frissítéseket, kéréseket, válaszokat és nyugtákat cserélhetnek egymással.
* 2-WAY/DROTHER - Két nem DR és nem BDR forgalomirányító szomszédsági viszonyban van egymással. A szomszédok Hello csomagokat cserélnek egymással.

Egy OSPF forgalomirányító normál esetben FULLállapotban van. Ha egy forgalomirányító huzamosabb ideig egy ettől eltérő állapotban marad, akkor ez a szomszédsági viszony kialakításának problémáját jelzi. Az egyetlen kivétel ez alól a 2-WAY állapot, ami még normálisnak tekinthető többes hozzáférésű hálózatokban.

Többes hozzáférésű hálózatokban a DROTHER forgalomirányítók csak a DR és a BDR forgalomirányítóval alakítanak ki teljes értékű (FULL) szomszédsági viszonyt. Ugyanakkor a DROTHER forgalomirányítók kétutas (2-WAY) szomszédsági viszonyt alakítanak ki minden hálózathoz csatlakozó DROTHER forgalomirányítóval. Ez azt eredményezi, hogy a többes hozzáférésű hálózatokban minden DROTHER forgalomirányító Hello üzeneteket kap az összes többi DROTHER forgalomirányítótól, és így ismerik a hálózat minden forgalomirányítóját. Amikor két DROTHER forgalomirányító szomszédsági viszonyt létesít egymással, a szomszédsági viszonyuk állapota 2-WAY/DROTHERlesz.

R1 kimenete alapján megállapítható, hogy a forgalomirányító szomszédsági viszonyai az alábbiak:

* A 2.2.2.2 azonosítójú R2 forgalomirányító FULL állapotban van és ő a BDR. (1)
* A 3.3.3.3 azonosítójú R3 FULL állapotban van és a ő a DR. (2)

R2 kimenete alapján megállapítható, hogy a forgalomirányító szomszédsági viszonyai az alábbiak:

* Az 1.1.1.1 azonosítójú R1 forgalomirányító FULL állapotban van és nem ő a DR vagy a BDR. (1)
* A 3.3.3.3 azonosítójú R3 FULL állapotban van és ő a DR. (2)

R3 kimenete alapján megállapítható, hogy a forgalomirányító szomszédsági viszonyai az alábbiak:

* Az 1.1.1.1 azonosítójú R1 forgalomirányító FULL állapotban van és nem ő a DR vagy a BDR. (1)
* A 2.2.2.2 azonosítójú R2 forgalomirányító FULL állapotban van és ő a BDR. (2)

**Alapértelmezett DR/BDR választási folyamat**

Hogyan történik a DR/BDR választás? Az OSPF DR és BDR választás az alábbi sorrend szerint megadott lépések alapján zajlik:

1. A hálózat forgalomirányítói a legnagyobb interfész prioritással rendelkező forgalomirányítót választják meg DR-nek, a második legnagyobb prioritásút pedig BDR-nek. A prioritás értéke 0 és 255 közötti értékre állítható be. Minél nagyobb a prioritása egy forgalomirányítónak, annál nagyobb eséllyel lesz ő a DR. Ha a prioritás értéke 0, akkor a forgalomirányító biztosan nem lesz DR. A többes hozzáférésű interfészek alapértelmezett prioritása 1. Így, hacsak másképpen nem konfiguráljuk, minden forgalomirányító prioritása egyenlő és a DR/BDR választásnál más módszer alapján kell dönteni.

2. Az interfész prioritások egyezése esetén a legnagyobb azonosítójú forgalomirányító lesz a DR, a második legnagyobb azonosítójú pedig a BDR.

Emlékezzünk rá, hogy a forgalomirányító azonosítójának (router ID) meghatározása az alábbi három mód egyikével történik:

* A forgalomirányító azonosító beállítása kézzel.
* Ha nincs kézzel beállított forgalomirányító azonosító, akkor a legnagyobb IP-címmel rendelkező loopback interfész címét használjuk erre a célra.
* Ha nincs loopback interfész konfigurálva, akkor a legnagyobb aktív IPv4-cím lesz az azonosító.

**MEGJEGYZÉS**: Ha egy IPv6-hálózatban nincs IPv4-cím konfigurálva, akkor a forgalomirányító azonosítót kézzel kell beállítani a **router-id** *azonosító* paranccsal; máskülönben az OSPFv3 nem indul el.

Az ábrán minden Ethernet interfész prioritása alapértelmezetten 1. Éppen ezért a korábban említett választási kritériumoknak megfelelően az OSPF forgalomirányító azonosítók alapján történik a DR és a BDR megválasztása. R3 lesz a DR, mivel ő rendelkezik a legnagyobb azonosítóval, R2 pedig a BDR, mivel az ő azonosítója a második legnagyobb.

**MEGJEGYZÉS**: A soros interfészek alapértelmezett prioritása 0, így nem vesznek részt a DR és BDR választásban.

Többes hozzáférésű hálózatban a DR/BDR választási folyamat azonnal elindul, amint az első OSPF interfésszel rendelkező forgalomirányító aktívvá válik. Ez akkor történik meg, ha a forgalomirányítót bekapcsoljuk, vagy ha az OSPF **network** parancsot kiadjuk az adott interfészen. A választási folyamat mindössze pár másodpercig tart. Ha a többes hozzáférésű hálózatban a választásig nem minden forgalomirányító fejezte be az indulási folyamatát, akkor előfordulhat, hogy egy kisebb azonosítójú forgalomirányító lesz a DR. (Ez lehet egy kisebb teljesítményű forgalomirányító, amelynek rövidebb ideig tart az indulási folyamata.)

# DR/BDR választási folyamat

Az OSPF DR/BDR választás nem preemptív. Ha a DR és BDR választást követően egy nagyobb prioritású vagy azonosítójú forgalomirányító csatlakozik a hálózathoz, akkor az új forgalomirányító nem változtatja meg a korábban kialakult DR és BDR szerepeket. Mivel ezek a szerepek már kiosztásra kerültek, egy új forgalomirányító csatlakozása a hálózathoz nem eredményez új választási folyamatot.

A DR megválasztását követően a DR szerepköre mindaddig megmarad, amíg az alábbiak valamelyike nem történik:

* A DR kiesik a hálózatból.
* Az OSPF-folyamat a DR forgalomirányítón hibásan működik vagy leáll.
* A DR többes hozzáférésű interfésze hibásan működik vagy letiltott állapotban van.

Ha a DR kiesik, akkor a BDR automatikusan átveszi a DR szerepét. Ez fog történni még abban az esetben is, ha a kezdeti DR/BDR választást követően egy nagyobb prioritású vagy azonosítójú DROTHER forgalomirányító kerül a hálózatba. Ugyanakkor, amint a BDR átveszi a DR szerepét, egy új BDR választás indul, és a legnagyobb prioritású vagy azonosítójú DROTHER forgalomirányító lesz a BDR.

Az 1-4. ábrák a DR/BDR választási folyamat különböző eseteit mutatják.

Az 1. ábrán a DR (R3) kiesik, és a BDR (R2) veszi át a szerepét. Ezt követően megkezdődik az új BDR kiválasztása. Mivel az R1 az egyetlen DROTHER, így ő lesz az új BDR.

A 2. ábrán R3 pár perces kiesést követően újra csatlakozik a hálózathoz. Mivel a hálózatban van már DR és BDR, így az R3 DROTHER forgalomirányító lesz.

A 3. ábrán egy új forgalomirányító (R4) csatlakozik a hálózathoz az eddigieknél nagyobb azonosítóval. A DR (R2) és a BDR (R1) szerepe nem változik, az R4 pedig automatikusan DROTHER lesz.

A 4. ábrán az R2 kiesik. A BDR (R1) automatikusan átveszi a DR szerepét, majd egy választási folyamat eredményeként a legnagyobb azonosítójú R4 lesz a BDR.

**Az OSPF-prioritás**

Mivel a DR forgalomirányító központi szerepet játszik az LSA-k összegyűjtésében és szétosztásában, így ennek a forgalomirányítónak elegendő CPU és memória kapacitással kell rendelkeznie a terhelés kezeléséhez. A konfiguráció során lehetőség van a DR/BDR választás befolyásolására.

Ha az interfész prioritás minden forgalomirányítón megegyezik, akkor a legnagyobb azonosítójú forgalomirányító lesz a DR. Az azonosító (router ID) módosításával lehetőség van befolyásolni a DR/BDR választást. Ez a megoldás azonban csak akkor működik, ha van egy szigorú terv a forgalomirányítók azonosítójának beállítására. Nagyméretű hálózatok esetében ez körülményes lehet.

A forgalomirányító azonosító helyett jobb, ha a választást az interfész prioritás konfigurálásával befolyásoljuk. A prioritás interfész specifikus érték, így hatékonyabban képes egy többes hozzáférésű hálózat szabályozására. Ez lehetővé teszi azt is, hogy egy forgalomirányító DR legyen az egyik és DROTHER egy másik hálózatban.

Egy interfész prioritása az alábbi paranccsal állítható be:

* **ip ospf priority** *érték* - OSPFv2 interfész parancs
* **ipv6 ospf priority** *érték* - OSPFv3 interfész parancs

Az *érték* lehet:

* **0** - A forgalomirányító nem lesz DR vagy BDR.
* **1** – **255** - Minél nagyobb a prioritás értéke, annál valószínűbb, hogy egy forgalomirányító DR vagy BDR lesz az adott interfészén.

Az ábrán minden forgalomirányító OSPF-prioritása megegyezik, mivel a prioritás értéke minden interfészen alapértelmezetten 1. Ennek következtében az azonosító alapján történik a DR (R3) és a BDR (R2) megválasztása. A forgalomirányító interfész prioritásának 1-nél nagyobb értékre állításával elérhető, hogy a következő választás során az adott forgalomirányító legyen a DR vagy a BDR.

Ha az interfész prioritás beállítása az OSPF-folyamat engedélyezését követően történt, akkor az új DR/BDR választás kikényszerítéséhez a rendszergazdának minden forgalomirányítón le kell állítani és újraindítani az OSPF-folyamatot.

# Az OSPF-prioritás megváltoztatása

Az 1. ábrán látható topológián az R3 forgalomirányító a DR és R2 a BDR. A konfiguráció módosításával szeretnénk elérni, hogy:

* Az R1 legyen a DR 255-ös beállított prioritással.
* Az R2 legyen a BDR alapértelmezett 1-es prioritással.
* Az R3 soha ne legyen se DR, se BDR, amit a 0 prioritás beállításával érhetünk el.

A 2. ábrán R1 Gigabit 0/0 interfészének prioritása 1-ről 255-re változik.

A 3. ábrán R3 Gigabit 0/0 interfészének prioritása 1-ről 0-ra változik.

A változtatások automatikusan nem lépnek életbe, mivel a DR/BDR választás már lezajlott. Éppen ezért szükség van az OSPF választás kikényszerítésére az alábbi módszerek egyikével:

* A forgalomirányító interfészek leállítása és újraengedélyezése a következő sorrendben: DR, BDR majd az összes többi forgalomirányító.
* Az OSPF-folyamat újraindítása minden forgalomirányítón a **clear ip ospf process** privilegizált EXEC módbeli paranccsal.

A 4. ábrán az R1 OSPF-folyamatának törlése látható. Tételezzük fel, hogy a **clear ip ospf process** privilegizált EXEC módbeli parancs az R2 és az R3 forgalomirányítón is kiadásra került. Figyeljük meg a létrejött OSPF állapot információkat.

Az 5. ábra kimenetei alapján látható, hogy az R1 forgalomirányító a DR 255-ös prioritás értékkel, valamint R1 új szomszédsági viszonyai is azonosíthatók.

A 6. ábrán lévő parancsszimulátorban vizsgáljuk meg az R2 és R3 forgalomirányítók szerepét, valamint szomszédsági viszonyaikat.

**OSPFv2 statikus alapértelmezett útvonal terjesztése**

**Statikus alapértelmezett útvonal terjesztése**

OSPF esetén az Internetre csatlakoztatott forgalomirányítót használjuk arra, hogy az alapértelmezett útvonalat továbbítsa az OSPF forgalomirányítási tartományban lévő összes többi forgalomirányítónak. Ezt a forgalomirányítót szokták határponti vagy átjáró forgalomirányítónak is nevezni. Az OSPF-terminológiában az OSPF irányítási tartomány és a nem OSPF alapú hálózat között található forgalomirányítót autonóm rendszer határforgalomirányítónak (Autonomous System Boundary Router, ASBR) nevezzük.

Az 1. ábrán az R2 egyetlen szolgáltatóhoz kapcsolódik, így az Internet eléréséhez mindössze egy alapértelmezett statikus útvonalra van szüksége.

**MEGJEGYZÉS**: A példában a 209.165.200.225 IP-című loopback interfész szimulálja a kapcsolatot a szolgáltatóval.

Az alapértelmezett útvonal hirdetéséhez a határforgalomirányítón (R2) a következőket kell elvégezni:

* Egy alapértelmezett statikus útvonal létrehozása az **ip route 0.0.0.0 0.0.0.0** {*IP-cím* **|** *kimenő\_interfész*} paranccsal.
* A **default-information originate** forgalomirányító konfigurációs módbeli parancs kiadása. Ez arra utasítja az R2 forgalomirányítót, hogy az alapértelmezett útvonal információk forrása legyen, és az OSPF-frissítésekben hirdesse az alapértelmezett statikus útvonalat.

A 2. ábrán látható, hogyan kell egy szolgáltató felé mutató alapértelmezett statikus útvonalat beállítani.

# A terjesztett alapértelmezett útvonal ellenőrzése

Ellenőrizzük az alapértelmezett útvonal beállításokat az R2-n a **show ip route** paranccsal, ahogy az 1. ábrán látható.

A 2. ábrán lévő parancsszimulátorban ellenőrizzük le, hogy az R1 és R3 forgalomirányítókra megérkezett-e az alapértelmezett útvonal hirdetmény! Figyeljük meg, hogy az útvonal forrása **O\*E2**, amely azt jelzi, hogy az útvonalat a forgalomirányítók OSPF segítségével tanulták meg. A csillag azt jelenti, hogy ez egy kijelölt alapértelmezett útvonal, az E2 pedig azt, hogy egy külső útvonal.

A külső útvonalak típusai lehetnek: 1-es típusú külső (E1) vagy 2-es típusú külső (E2) útvonal. A két típus közötti különbség az útvonal költség (mérték) kiszámításának módja. Egy 2-es típusú útvonal költsége mindig a külső költség, függetlenül az útvonal eléréséhez szükséges belső költségtől. Egy 1-es típusú útvonal költsége a külső és belső költségek összege. Ugyanahhoz a célhoz vezető útvonalak közül mindig az 1-es típusú útvonal részesül előnyben a 2-es típusú útvonallal szemben.

**OSPFv3 alapértelmezett statikus útvonal terjesztése**

Az OSPFv3 alapértelmezett statikus útvonal terjesztésének folyamata szinte teljesen megegyezik az OSPFv2 esetében tanultakkal.

Az 1. ábrán az R2 egyetlen szolgáltatóhoz kapcsolódik, így az Internet eléréséhez mindössze egy alapértelmezett statikus útvonalra van szüksége.

**MEGJEGYZÉS**: A példában a 2001:DB8:FEED:1::1/64 IP-című loopback interfész szimulálja a kapcsolatot a szolgáltatóval.

A 2. ábrán az R1 aktuális IPv6-irányítótáblája látható. Figyeljük meg, hogy az R1-nek nincs információja az Internet felé vezető útvonalról.

Az alapértelmezett útvonal hirdetéséhez a határforgalomirányítón (R2) a következőket kell elvégezni:

* Egy alapértelmezett statikus útvonal létrehozása az **ipv6 route ::/0** {*IPv6-cím* | *kimenő\_interfész*} paranccsal.
* A **default-information originate** forgalomirányító konfigurációs módbeli parancs kiadása. Ez arra utasítja az R2 forgalomirányítót, hogy az alapértelmezett útvonal információk forrása legyen, és OSPF-frissítésekben hirdesse az alapértelmezett statikus útvonalat.

A 3. ábra példájában egy szolgáltató felé mutató alapértelmezett statikus útvonal beállítása látható.

# A terjesztett alapértelmezett IPv6-útvonal ellenőrzése

Ellenőrizzük az alapértelmezett statikus útvonal beállításokat az R2-n a **show ipv6 route** paranccsal, ahogy az 1. ábrán látható.

A 2. ábrán található parancsszimulátorban ellenőrizzük, hogy az alapértelmezett útvonal hirdetmény eljutott-e az R1 forgalomirányítóhoz! Figyeljük meg, hogy az útvonal forrása **OE2** , amely azt jelzi, hogy az útvonalat a forgalomirányító OSPFv3 segítségével tanulta meg. Az E2 azt mutatja, hogy ez egy külső útvonal.

Az IPv4-irányítótáblával ellentétben az IPv6 nem használja a \* szimbólumot a kijelölt alapértelmezett útvonal megjelölésére.

# OSPF Hello és Dead időzítők

Az OSPF Hello és Dead időzítői interfészenként konfigurálhatók. A szomszédokon az OSPF-időzítők értékeinek meg kell egyezni, különben a szomszédsági viszony nem jön létre.

Az interfész időzítők aktuális beállítását a **show ip ospf interface** paranccsal kérdezhetjük le, ahogy az 1. ábrán is látható. A Serial 0/0 interfész Hello és Dead időzítőjének értéke az alapértelmezett 10 és 40 másodperc.

A 2. ábrán egy parancskimenet szűrési technika látható, amellyel megjeleníthetők az R1 forgalomirányító Serial 0/0/0 interfészének OSPF-időzítői.

A 3. ábrán az R1 forgalomirányítón kiadott **show ip ospf neighbor** parancs kimenetéből látható, hogy az R1 szomszédsági viszonyban van az R2 és R3 forgalomirányítókkal. Figyeljük meg a kimenetben, hogy a Dead Time értéke 40-ről indulva csökken. Alapértelmezetten ez a számláló 10 másodpercenként frissül, amikor az R1 Hello üzenetet kap a szomszédtól.

**Az OSPFv2-időzítők módosítása**

Gyakran hasznos lehet az OSPF-időzítők megváltoztatása annak érdekében, hogy a forgalomirányítók a hálózati hibákat kevesebb idő alatt érzékeljék. Ezzel ugyan nő a forgalom, de sokszor a gyors konvergencia lényegesen fontosabb, mint az így létrehozott többletterhelés.

**MEGJEGYZÉS**: Az alapértelmezett Hello és Dead időzítők értékei tapasztalati úton alakultak ki és csak különleges esetben ajánlott a módosításuk.

Az OSPF Hello és Dead időzítők kézzel is módosíthatók az alábbi interfész konfigurációs parancsokkal:

* **ip ospf hello-interval** *másodperc*
* **ip ospf dead-interval** *másodperc*

Az időzítők a **no ip ospf hello-interval** és a **no ip ospf dead-interval** parancsokkal állíthatók vissza alapértelmezett értékükre.

Az 1. ábrán lévő példa a Hello időzítő értékének 5 másodpercre történő beállítását mutatja. A Hello időzítő megváltoztatását követően a Cisco IOS automatikusan módosítja a Dead időzítő értékét a Hello időzítő értékének négyszeresére. Az időzítő megváltoztatását azonban nem érdemes az IOS automatizmusára rábízni, érdemes azt inkább egyértelműen megváltoztatni, mivel a változás így dokumentálva lesz a konfigurációban. Ezért az R1 Serial 0/0/0 interfészének Dead időzítőjét is manuálisan állítjuk be 20 másodpercre.

Az 1. ábrán a kiemelt OSPFv2 szomszédsági üzenetből látható, hogy amint a Dead időzítő lejárt az R1 forgalomirányítón, megszűnt a szomszédsági viszony az R1 és R2 között. Ennek oka, hogy az időzítők értékei az R1 és R2 közötti soros kapcsolat egyik végén változtak csak meg. Emlékezzünk arra, hogy a szomszédokon az OSPF Hello és Dead időzítők értékeinek meg kell egyezni.

A **show ip ospf neighbor** paranccsal ellenőrizhetjük a szomszédsági viszonyokat az R1-en a 2. ábrán látható módon. Figyeljük meg, hogy a 3.3.3.3 (R3) forgalomirányító az egyetlen szomszéd, és az R1 már nincs szomszédsági viszonyban a 2.2.2.2 (R2) szomszéddal. A Serial 0/0/0 interfészen beállított időzítők nem befolyásolják a szomszédsági viszonyt az R3 forgalomirányítóval.

Az R1 és R2 közötti szomszédsági viszony visszaállításához az R2 Serial 0/0/0 interfészének Hello időzítőjét **5** másodpercre kell állítani a 3. ábrán látható módon. Szinte azonnal egy IOS üzenet jelenik meg arról, hogy a szomszédsági viszony létrejött, a szomszédok közötti állapot pedig **FULL**.

Az interfész időzítők beállítása a **show ip ospf interface** paranccsal ellenőrizhető a 4. ábrán látható módon. Figyeljük meg, hogy a Hello időzítő értéke 5 másodperc, a Dead időzítő értéke pedig automatikusan 20 másodpercre változott az alapértelmezett 40 helyett. Emlékezzünk rá, hogy az OSPF automatikusan beállítja a Dead időzítő értékét a Hello időzítő értékének négyszeresére.

**A forgalomirányítók, mint célpontok**

A forgalomirányítók a hálózatban betöltött fontos szerepük miatt gyakran célpontjai támadásoknak. A hálózati rendszergazdáknak tisztában kell lenniük azzal, hogy a forgalomirányítók ugyanúgy ki vannak téve a támadásoknak, mint a végfelhasználói eszközök.

A forgalomirányító rendszerek általában a szomszédos eszközök kapcsolatának megszakítása vagy a forgalomirányítási információk meghamisítása révén támadhatóak. A meghamisított irányítási információkat a támadók általában arra használják, hogy általuk a rendszerek félreinformálják egymást. Mindez szolgáltatásmegtagadási (denial-of-service, DOS) támadást vagy a forgalom hamis útvonal történő továbbítását okozhatja. Az irányítási információk hamisításának következményei:

* Irányítási hurkok kialakítása a forgalom átirányításával
* Az átirányított forgalom megfigyelése egy nem biztonságos összeköttetésen
* Az átirányított forgalom eldobása

Az animáció Lejátszás gombjára kattintva megnézhetünk egy irányítási hurkot létrehozó támadást. A támadó közvetlenül hozzáfér az R1 és R2 forgalomirányítók közötti összeköttetéshez. A támadó hamis irányítási információt küld az R1-nek, amiben az R2-t jelöli meg következő ugrásnak a 192.168.10.10/32 állomás felé vezető úton. Bár az R1 irányítótáblájában szerepel egy közvetlenül csatlakozó ("directly connected") 192.168.10.0/24 hálózat, a hosszabb alhálózati maszk miatt a hamis útvonal is bekerül az irányítótáblájába. A hosszabb egyező alhálózati maszkkal rendelkező útvonal mindig előnyben részesül. Következtetésképpen, ha egy forgalomirányító csomagot kap, akkor a hosszabb alhálózati maszkot választja, mivel az egy pontosabb útvonal a célhoz.

Amikor a PC3 csomagot küld a PC1-nek (192.168.10.10/24), akkor az R1 nem továbbítja azt a PC1 állomásnak. Ehelyett a csomagot az R2-nek küldi, mivel egyértelműen az a legjobb útvonal a 192.168.10.10/32 állomáshoz. Amikor az R2 megkapja a csomagot, az irányítótáblája alapján visszaküldi a csomagot az R1-nek, és így irányítási hurok alakul ki.

Az irányító protokollok elleni támadások kivédése érdekében konfiguráljunk OSPF-hitelesítést.

**Biztonságos útvonalfrissítések**

Ha egy forgalomirányítón a szomszédok hitelesítése be van állítva, akkor a forgalomirányító minden fogadott útvonalfrissítés forrását hitelesíti. Mindez egy hitelesítő kulcs (gyakran jelszónak is nevezik) cseréjével történik, amit a küldő és a fogadó forgalomirányítók mindegyike ismer.

Az útvonalfrissítések biztonságos cseréje érdekében engedélyezzük az OSPF-hitelesítést. Az OSPF-hitelesítés lehet null (nincs hitelesítés), egyszerű vagy Message Digest (MD5) típusú.

Az OSPF 3 fajta hitelesítést támogat:

* **Null** - Ez az alapértelmezett mód, és ebben az esetben nincs OSPF-hitelesítés.
* **Egyszerű jelszóhitelesítés** - Nyílt szöveges hitelesítésnek is hívják, mivel a frissítésekben a jelszó titkosítatlanul, nyílt szövegként kerül átküldésre a hálózaton. Az OSPF-hitelesítésnek ezt a fajtáját hagyományos módszernek is tekinthetjük.
* **MD5-hitelesítés** - Ez a legbiztonságosabb és egyben az ajánlott hitelesítési mód. Az MD5-hitelesítés nagyobb biztonságot jelent, mivel a hálózati csomópontok között nem történik jelszócsere. Ehelyett az aláírás az MD5-algoritmussal kerül kiszámításra. A jelszavak egyeztetése hitelesíti a küldőt.

Kattintsunk az animáció Lejátszás gombjára és tekintsük meg a szomszédtól érkező üzenetek MD5-hitelesítését!

**MEGJEGYZÉS**: A RIPv2, EIGRP, OSPF, IS-IS és BGP protokollok mindegyike támogatja az MD5 hitelesítés különböző formáit.

**MD5-hitelesítés**

Az ábrán látható példa két szomszédos OSPF forgalomirányító MD5-hitelesítési folyamatát mutatja be.

Az 1. ábrán R1 az irányítási üzenetet és az előre megosztott kulcsot felhasználva az MD5-algoritmus segítségével kiszámolja az aláírást. Az aláírást hash értéknek is nevezzük.

A 2. ábrán R1 az irányítási üzenetet az aláírással együtt elküldi R2-nek.

Mivel az MD5 nem titkosítja az üzenetet, így a tartalma könnyen kiolvasható.

A 3. ábrán R2 megnyitja a csomagot, egyesíti az irányítási üzenetet az előre megosztott kulccsal, majd az MD5-algoritmussal kiszámolja az aláírást.

* Ha az aláírások megegyeznek, akkor R2 elfogadja az útvonalfrissítést.
* Ha az aláírások nem egyeznek meg, akkor R2 eldobja az útvonalfrissítést.

Az OSPFv3 (IPv6-os OSPF) nem rendelkezik semmilyen saját hitelesítési képességgel. Ehelyett teljes mértékben az IPSec protokollra bízza a szomszédok közötti biztonságos kommunikációt a **ipv6 ospf authentication ipsec spi** interfész konfigurációs parancs használatával. Ennek előnye az OSPFv3 protokoll egyszerűsítésében és a hitelesítési eljárás szabványosításban rejlik.

**OSPF MD5-hitelesítés konfigurálása**

Az OSPF az MD5 irányítóprotokoll hitelesítést támogatja. Az MD5-hitelesítés engedélyezhető globálisan vagy interfészenként.

Az MD5 globális engedélyezéséhez használjuk az

* **ip ospf message-digest-key** *kulcs* **md5** *jelszó* interfész konfigurációs parancsot.
* **area** *terület\_azonosító* **authentication message-digest** forgalomirányító konfigurációs módbeli parancsot.

Ez a módszer minden OSPF interfészen engedélyezi a hitelesítést. Ha egy interfészen nincs beállítva az **ip ospf message-digest-key** parancs, akkor nem lesz képes szomszédsági viszonyt kialakítani más OSPF szomszédokkal.

A nagyobb rugalmasság érdekében a hitelesítés interfészenként is beállítható. Az MD5-hitelesítés interfészen történő engedélyezéséhez használjuk az

* **ip ospf message-digest-key** *kulcs* **md5** *jelszó* interfész konfigurációs parancsot.
* **ip ospf authentication message-digest** interfész konfigurációs parancsot.

Ugyanazon a forgalomirányítón konfigurálhatunk globális és interfészenkénti MD5-hitelesítést is. Ebben az esetben viszont az interfész beállítás felülírja a globális konfigurációt. Az MD5 hitelesítési jelszónak nem kell megegyeznie egy területen belül, viszont azonosnak kell lennie két szomszéd esetében.

Tegyük fel például, hogy az ábrán lévő összes forgalomirányító konvergált, és az OSPF forgalomirányítás is megfelelően működik. Továbbá minden forgalomirányítón OSPF-hitelesítés van beállítva.

# Példa OSPF MD5-hitelesítésre

Az 1. ábrán lévő példában OSPF MD5-hitelesítés beállítása látható minden interfészen. Figyeljük meg a tájékoztató üzeneteket, amelyek szerint az R2 és R3 közötti OSPF szomszédsági viszony megváltozott Leállítva állapotúra, mivel sem az R2 sem az R3 forgalomirányítón nincs MD5-hitelesítés még beállítva.

A 2. ábra példája azt mutatja, hogy az MD5-hitelesítés globális engedélyezésének alternatívájaként miként lehet az R1 forgalomirányítón az OSPF MD5-hitelesítést interfészenként engedélyezni. Figyeljük meg ismét, hogy az OSPF szomszédsági viszonyok megváltoztak Leállítva állapotra.

A 3. ábra parancsszimulátorában konfiguráljunk globális OSPF MD5-hitelesítést az R2-n, interfészenkénti hitelesítést pedig az R3-on.

Ismét tájékoztató üzenetek jelennek meg. Az első azt mutatja, hogy helyreállt a szomszédsági viszony az R1-gyel. A szomszédsági viszony állapota az R3-mal viszont megváltozott Leállítva állapotúra, mivel R3 még nincs konfigurálva.

R3 beállítását követően minden szomszédsági viszony helyreáll.

# OSPF MD5-hitelesítés ellenőrzése

Annak ellenőrzésére, hogy az OSPF MD5-hitelesítés engedélyezve van-e, használjuk a **show ip ospf interface** privilegizált EXEC módbeli parancsot. Az irányítótábla teljességének ellenőrzésével igazolhatjuk a sikeres hitelesítést.

Az 1. ábrán az OSPF MD5-hitelesítés ellenőrzése látható az R1 forgalomirányító Serial 0/0/0 interfészén.

A 2. ábra azt igazolja, hogy a hitelesítés sikeres.

A 3. ábrán lévő parancsszimulátorban ellenőrizzük az OSPF MD5-hitelesítést az R2 és az R3 forgalomirányítókon!

# Packet Tracer - Az OSPFv2 haladó beállításai

**Háttér / Esetleírás**

A feladatban az OSPF konfigurálása már megtörtént, és minden felhasználói eszköz hálózati kapcsolata teljes értékű. Módosítani fogjuk az alapértelmezett OSPF konfigurációt a Hello és Dead időzítők megváltoztatásával, egy összeköttetés sávszélességének beállításával és MD5-hitelesítés konfigurálásával. Ezt követően ellenőrizni fogjuk az eszközök közötti kapcsolatok helyreállását.

[Packet Tracer - Configuring OSPF Advanced Features Instructions](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/5.1.5.7%20Packet%20Tracer%20-%20Configuring%20OSPF%20Advanced%20Features%20Instructions.pdf)

[Packet Tracer - Configuring OSPF Advanced Features - PKA](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/5.1.5.7%20Packet%20Tracer%20-%20Configuring%20OSPF%20Advanced%20Features.pka)

# Áttekintés

Az OSPF a nagyvállalati hálózatokban gyakran alkalmazott irányító protokoll. Egy OSPF belső irányító protokollt használó nagyvállalati hálózat megvalósításában és karbantartásában résztvevő szakember számára az irányítási információk cseréjével kapcsolatos hibaelhárítási problémák megoldása az egyik legfontosabb készség.

Az ábrán az OSPF szomszédsági viszonyok kialakulásának problémái láthatók.

# OSPF-állapotok

Az OSPF hibaelhárításához fontos megértenünk, hogy a forgalomirányítók a szomszédsági viszony kialakítása során miként kerülnek a különböző OSPF-állapotokba.

Az ábrán az OSPF-állapotok és azok feladatai láthatók.

Az OSPF szomszédok vizsgálatakor legyünk tisztában azzal, hogy a FULL vagy 2WAY állapotok jelentik a megfelelő működést. Az összes többi állapot csak átmeneti, azaz a forgalomirányító hosszabb ideig nem tartózkodik ezekben az állapotokban.

# OSPF hibakeresési parancsok

Számos OSPF parancs segíti a hibaelhárítási folyamatot. Az alábbiakban lássuk a leggyakrabban használtakat:

* **show ip protocols** (1. ábra) - A parancs az OSPF konfigurációs információkat jeleníti meg, köztük az OSPF folyamatazonosítót, a forgalomirányító azonosítóját, a forgalomirányító által hirdetett hálózatokat, a szomszédokat, akiktől a forgalomirányító frissítéseket kap és az alapértelmezett adminisztratív távolságot, ami OSPF esetében a 110.
* **show ip ospf neighbor** (2. ábra) - A paranccsal a forgalomirányító szomszédsági viszonyainak kialakítása ellenőrizhető. Megjeleníti a szomszédos forgalomirányító azonosítóját, a szomszéd prioritását, az OSPF-állapotot, a Dead időzítőt, a szomszéd interfész IP-címét és azt az interfészt, amin keresztül a szomszéd elérhető. Ha a szomszédos forgalomirányító azonosítója nem jelenik meg, illetve nem FULL vagy 2WAY állapotúnak látszik, akkor a két forgalomirányító között nem alakult ki OSPF szomszédsági viszony. Ha két forgalomirányító nincs egymással szomszédsági viszonyban, akkor kapcsolatállapot adatokat sem cserélnek. A hiányos kapcsolatállapot-adatbázisok (link-state database, LSDB) pedig pontatlan SPF fákat és irányítótáblákat eredményeznek. Egyes célhálózatokhoz vezető utak hiányozhatnak, vagy nem a legoptimálisabb utat biztosíthatják.
* **show ip ospf interface** (3. ábra) - Megjeleníti az interfészek konfigurált OSPF paramétereit, köztük az OSPF folyamatazonosítót, amihez az interfész hozzá van rendelve, a terület azonosítót, amihez az interfész tartozik, az interfész költségét valamint a Hello és Dead időzítőket. Az interfész típusának és számának megadásával a parancs az adott interfész paramétereit adja meg.
* **show ip ospf** (4. ábra) - A parancs megjeleníti az OSPF folyamatazonosítót és a forgalomirányító azonosítóját, valamint az OSPF területinformációt és az SPF-algoritmus legutóbbi futásának időpontját.
* **show ip route ospf** (5. ábra) - A parancs csak az OSPF által megtanult útvonalakat mutatja az irányítótáblában. A kimenet alapján megállapítható, hogy R1 négy távoli hálózatot tanult meg az OSPF protokoll segítségével.
* **clear ip ospf** [ *folyamat-azonosító* ] **process** - Az OSPFv2 szomszédsági viszonyok törlésére használható.

**Az OSPF hibaelhárítás összetevői**

Ahogy az ábrán is látható, az OSPF problémái leggyakrabban az alábbiakkal vannak összefüggésben:

* Szomszédsági viszonyok
* Hiányzó útvonalak
* Útvonalak kiválasztása

A szomszédsági problémák elhárításakor a **show ip ospf neighbor** paranccsal ellenőrizhetjük, hogy a forgalomirányító alakított-e ki viszonyt szomszédos forgalomirányítókkal. Ha nem, akkor a forgalomirányítók nem képesek útvonalak cseréjét végrehajtani. A **show ip interface brief** és a **show ip ospf interface** parancsokkal ellenőrizhetjük, hogy működnek-e az interfészek és engedélyezett-e rajtuk az OSPF protokoll. Ha az interfészek működnek és engedélyezett rajtuk az OSPF, akkor ellenőrizzük le, hogy mindkét forgalomirányító interfész ugyanahhoz az OSPF területhez tartozik és az interfészek nem passzív interfészek.

Ha a forgalomirányítók közötti szomszédsági viszony kialakult, akkor a **show ip route ospf** parancs segítségével győződhetünk meg arról, hogy az irányítótáblában vannak OSPF útvonalak. Ha nincsenek, akkor ellenőriznünk kell, hogy a hálózatban nem működik-e alacsonyabb adminisztratív távolsággal rendelkező irányító protokoll. Ellenőrizzük, hogy minden szükséges hálózat hirdetése megtörtént-e, illetve, hogy a forgalomirányítón nincs-e olyan hozzáférési lista, ami szűrné a kimenő vagy bejövő útvonalfrissítéseket.

Ha az összes szükséges útvonal megjelenik az irányítótáblában, de az útvonal, amelyen a forgalom halad, mégsem helyes, ellenőriznünk kell az útvonal interfészeinek költségét. Figyelnünk kell arra is, hogy a 100 Mb/s-nál nagyobb sebességű interfészek alapértelmezett OSPF költsége megegyezik.

# Szomszédsági problémák elhárítása

A példa a szomszédsági problémák hibaelhárítási folyamatát mutatja. Az 1. ábrán lévő topológia minden forgalomirányítóján OSPF irányító protokoll fut.

A 2. ábrán azt láthatjuk, hogy az R1 irányítótáblájában nincs OSPF útvonal. Ennek számos oka lehet. Két forgalomirányító közötti szomszédsági viszony kialakulásának előfeltétele elsősorban az OSI 3. rétegbeli kapcsolat megléte.

A 3. ábra kimenetéből látható, hogy az S0/0/0 interfész aktív és felkapcsolt állapotú. A sikeres ping pedig azt igazolja, hogy R2 soros interfésze is aktív. A sikeres ping viszont még nem jelenti a szomszédsági viszony kialakulását, mivel elképzelhető, hogy az alhálózatok átfedésben vannak egymással. Ellenőriznünk kell még, hogy a csatlakoztatott eszközök interfészei egy közös alhálózaton vannak-e. Ha a ping sikertelen, ellenőrizzük a kábelezést valamint a csatlakoztatott eszközök interfészeinek működőképességét és beállításaik megfelelőségét.

Egy interfészen az OSPF engedélyezéséhez a megfelelő **network** parancsot kell kiadni az OSPF irányítási folyamaton belül. Az aktív OSPF interfészek ellenőrzésére a **show ip ospf interface** parancs szolgál. A 4. ábra kimenetéből látható, hogy az S0/0/0 interfészen engedélyezett az OSPF protokoll. Amennyiben két forgalomirányító egymáshoz csatlakozó interfészein nincs engedélyezve az OSPF, nem alakul ki szomszédsági viszony közöttük.

Az OSPF beállítások a **show ip protocols** paranccsal ellenőrizhetők. Az 5. ábra kimenete alapján látható, hogy az OSPF működőképes, valamint a **network** paranccsal hirdetett hálózatok is megjelennek benne. Ha egy interfész IP-címe olyan hálózatba esik, amelyben engedélyezett az OSPF, akkor a protokoll ezen az interfészen is engedélyezetté válik.

Ugyanakkor figyeljük meg, hogy a Serial 0/0/0 interfész passzívként jelenik meg. Emlékezzünk rá, hogy a **passive-interface** parancs leállítja a kimenő és bejövő útvonalfrissítéseket. Ugyanis a parancs hatására a forgalomirányító nem küld és nem fogad Hello csomagokat az interfészén. Így a forgalomirányítók sem lesznek szomszédok.

Az interfész passzív állapotának megszüntetésére a **no passive-interface** forgalomirányító-konfigurációs módbeli parancsot használhatjuk a 6. ábrán látható módon. A passzív interfész megszüntetését követően a forgalomirányítók között létrejön a szomszédsági viszony, ahogy ez az automatikusan generált üzenetben is látható.

A 7. ábrán lévő irányítótábla igazolja, hogy az OSPF irányítási információk cseréje már működik.

Egy másik felmerülő probléma lehet, ha két szomszédos forgalomirányító kapcsolódó interfészein eltérő MTU érték van beállítva. Az MTU érték megadja a legnagyobb hálózati rétegbeli csomag méretét, amit a forgalomirányító kimenő irányba továbbít az interfészein. A forgalomirányítók alapértelmezett MTU értéke 1500 bájt. Ez az érték IPv4 csomagok esetén az **ip mtu** *méret*interfész konfigurációs paranccsal, IPv6 esetében pedig az **ipv6 mtu** *méret*interfész konfigurációs paranccsal változtatható meg. Ha két egymáshoz kapcsolódó forgalomirányítón eltérő MTU érték van beállítva, akkor a forgalomirányítók megpróbálják ugyan a szomszédsági viszony kialakítását, de mivel nem cserélnek LSDB üzeneteket egymással, így a viszony nem alakul ki közöttük.

# OSPF irányítótábla problémák hibaelhárítása

Az 1. ábrán lévő topológia minden forgalomirányítóján OSPF irányító protokoll fut.

Az R1 irányítótáblája a 2. ábrán azt mutatja, hogy a forgalomirányító megkapta az információkat az alapértelmezett útvonalról, az R2 LAN-járól (172.16.2.0/24) valamint az R2 és R3 közötti összeköttetésről (192.168.10.8/30). Az R3 LAN-ról viszont nem kapta meg az OSPF útvonalat.

A 3. ábra kimenete az R3 forgalomirányító OSPF beállításait mutatja. Figyeljük meg, hogy R3 csak az R3 és R2 közötti összeköttetést hirdeti. Nem hirdeti viszont a saját helyi hálózatát (192.168.1.0/24).

Egy interfészen az OSPF engedélyezéséhez a megfelelő **network** parancsot kell kiadni az OSPF irányítási folyamaton belül. A 4. ábra kimenete igazolja, hogy az R3 helyi hálózatát nem hirdeti az OSPF.

Az 5. ábrán az R3 helyi hálózatára vonatkozó **network** parancs kiadása látható. Az R3 most már hirdeti a saját helyi hálózatát az OSPF szomszédainak.

A 6. ábra kimenetében R3 helyi hálózata már szerepel az R1 irányítótáblájában.

**OSPFv3 hibakeresési parancsok**

A következőkben vegyük alapul az 1. ábrán látható OSPFv3-topológiát!

Az OSPFv3 protokoll hibakeresési eljárása szinte teljesen megegyezik az OSPFv2 folyamattal, így számos OSPFv3 parancs és hibakeresési feltétel érvényes az OSPFv3 protokollra is.

Az OSPFv3 esetében használt megfelelő parancsok például az alábbiak:

* **show ipv6 protocols** (2. ábra) - A paranccsal ellenőrizhetők a főbb OSPFv3 konfigurációs adatok, beleértve az OSPFv3 folyamatazonosítót, a forgalomirányító azonosítóját és azokat az interfészeket, ahonnan a forgalomirányító frissítéseket fogad.
* **show ipv6 ospf neighbor** (3. ábra) - A paranccsal a forgalomirányító szomszédsági viszonyainak kialakítása ellenőrizhető. A kimenet megjeleníti a szomszédos forgalomirányító azonosítóját, a szomszéd prioritását, az OSPFv3 állapotot, a Dead időzítőt, a szomszéd interfész azonosítóját és azt az interfészt, amin keresztül a szomszéd elérhető. Ha a szomszéd forgalomirányító azonosítója (router ID) nem jelenik meg, vagy a szomszéd nem **FULL** vagy **2WAY**állapotúnak látszik, akkor a két forgalomirányító között nem alakult ki OSPFv3 szomszédsági viszony. Ha két forgalomirányító nincs szomszédsági viszonyban egymással, akkor kapcsolatállapot adatokat sem cserélnek. A hiányos kapcsolatállapot-adatbázisok (link-state database, LSDB) pedig pontatlan SPF fákat és irányítótáblákat eredményeznek. Egyes hálózatokhoz hiányozhat az útvonalbejegyzés, vagy nem az optimális útvonal kerül bejegyzésre.
* **show ipv6 ospf interface** (4. ábra) - A parancs megjeleníti az interfészek konfigurált OSPFv3 paramétereit, köztük az OSPFv3 folyamatazonosítót, amihez az interfész hozzá van rendelve, a terület azonosítót, amihez az interfész tartozik, az interfész költségét valamint a Hello és Dead időzítőket. Az interfész típusának és számának megadásával a parancs az adott interfész paramétereit adja meg.
* **show ipv6 ospf** (5. ábra) - A parancs megjeleníti az OSPF folyamatazonosítót, a forgalomirányító azonosítóját, valamint információt az LSA átvitelekről.
* **show ipv6 route ospf** (6. ábra) - A parancs csak az OSPFv3 által megtanult útvonalakat mutatja az irányítótáblában. A kimenet alapján megállapítható, hogy R1 négy távoli hálózatot tanult meg az OSPFv3 protokoll segítségével.
* **clear ipv6 ospf** [ *folyamat-azonosító* ] **process** - Az OSPFv3 szomszédsági viszonyok törlésére használható.

# OSPFv3 hibaelhárítása

Az 1. ábrán lévő topológia minden forgalomirányítóján OSPFv3 irányító protokoll fut.

Az R1 IPv6-irányítótáblája a 2. ábrán azt mutatja, hogy a forgalomirányító megkapta az információkat az alapértelmezett útvonalról, az R2 LAN-járól (2001:DB8:CAFE:2::/64) valamint az R2 és R3 közötti összeköttetésről (2001:DB8:CAFE:A002::/64). Az R3 LAN-ról viszont nem kapta meg az OSPFv3 útvonalat (2001:DB8:CAFE:3::/64).

A 3. ábra kimenete az R3 forgalomirányító OSPFv3 beállításait mutatja. Figyeljük meg, hogy az OSPF csak a Serial 0/0/1 interfészen van engedélyezve, és úgy tűnik, hogy az R3 G0/0 interfészén viszont nincs engedélyezve.

Ellentétben az OSPFv2 protokollal az OSPFv3 nem használja a **network** parancsot. Ehelyett az OSPFv3 protokollt közvetlenül az interfészen kell engedélyezni. A 4. ábra kimenetéből látható, hogy az R3 interfészen nincs engedélyezve az OSPFv3 protokoll.

Az 5. ábra példájában az OSPFv3 protokoll engedélyezése látható az R3 Gigabit Ethernet 0/0 interfészén. Az R3 most már hirdeti a saját helyi hálózatát az OSPFv3 szomszédainak.

A 6. ábra kimenetében R3 helyi hálózata már szerepel az R1 irányítótáblájában.

# Csoportos feladat - OSPF hibaelhárítása

**OSPF hibaelhárítása**

Elhatároztuk, hogy a RIPv2 irányító protokollt OSPFv2-re cseréljük. A kisméretű vállalatunk hálózati topológiájának fizikai beállításai nem változnak. Használjuk a PDF fájlban lévő ábrát a feladatban a kisméretű vállalatunk hálózatának terveként.

A forgalomirányítókon konfiguráljunk IPv4-címzést és VLSM-et a címzési terv alapján. A hálózat irányító protokollja az OSPF, viszont vannak forgalomirányítók, amelyek megosztják egymással az irányítási információikat, míg mások nem.

Nyissuk meg a feladathoz tartozó PDF fájlt, és az utasításokat követve oldjuk meg a feladatot.

Az utasításokban lévő lépések elvégzése után rendezzük át az osztályt és hasonlítsuk össze a feladat javításához szükséges időket. Az a csoport a győztes, akinek a leghamarabb sikerül megtalálni és kijavítani a konfigurációs hibát. A győztes csapat mesélje el a többieknek, hogyan sikerült megtalálniuk a hibát és miként tették a hálózatot működőképessé.

[Csoportos feladat - OSPF hibaelhárítása](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/5.3.1.1%20OSPF%20Troubleshooting%20Mastery%20Instructions.pdf)

# Összefoglalás

Az OSPF öt hálózati típust határoz meg: pont-pont, szórásos többes hozzáférésű, nem szórásos többes hozzáférésű, pont-multipont és virtuális kapcsolatok.

A többes hozzáférésű hálózatok az OSPF LSA elárasztás szempontjából két kihívást is jelentenek: többes viszonyok kialakítása és túlzott LSA elárasztás. A többes hozzáférésű hálózatokban a szomszédsági viszonyok számának és az LSA elárasztás problémájának kezelésére jelent megoldást a kijelölt és tartalék kijelölt forgalomirányítók (DR és BDR) megválasztása. Ha a DR nem küld Hello üzeneteket, akkor a BDR veszi át a szerepét.

A hálózat forgalomirányítói a legnagyobb interfész prioritással rendelkező forgalomirányítót választják meg DR-nek. A második legnagyobb prioritású forgalomirányító lesz a BDR. Minél nagyobb a prioritása egy forgalomirányítónak, annál nagyobb eséllyel lesz ő a DR. Ha a prioritás értéke 0, akkor a forgalomirányító biztosan nem lesz DR. A többes hozzáférésű interfészek alapértelmezett prioritása 1. Így, hacsak másképpen nem konfiguráljuk, minden forgalomirányító prioritása egyenlő és a DR/BDR választásnál más módszer alapján kell dönteni. Az interfész prioritások egyezése estén a legnagyobb azonosítójú forgalomirányító lesz a DR, a második legnagyobb azonosítójú pedig a BDR. Egy új forgalomirányító csatlakozása a hálózathoz nem eredményez új választási folyamatot.

OSPF esetén az alapértelmezett útvonal hirdetéséhez a forgalomirányítón először létre kell hozni az alapértelmezett statikus útvonalat, majd ki kell adni a **default-information originate** parancsot. Az útvonalakat a **show ip route** vagy a **show ipv6 route** paranccsal ellenőrizhetjük.

A 100 Mb/s-nél gyorsabb hálózatok megfelelő kezelésének érdekében a referencia sávszélességet magasabb értékűre kell állítanunk annak érdekében, hogy az OSPF a helyes útvonalakat válassza. A referencia sávszélesség beállítására az **auto-cost reference-bandwidth** *Mbit/s* forgalomirányító konfigurációs módbeli parancs használható. Az interfész sávszélességének beállítására a **bandwidth** *kilobits* interfész konfigurációs parancs szolgál. Az interfészek költsége manuálisan is állítható az **ip ospf cost** *érték* interfész konfigurációs paranccsal.

Az OSPF Hello és Dead időzítőinek egyező értékekkel kell rendelkezniük a szomszédokon, különben a szomszédsági viszony nem jön létre. Az időzítők értékeinek megváltoztatásához az alábbi interfész parancsok használhatók:

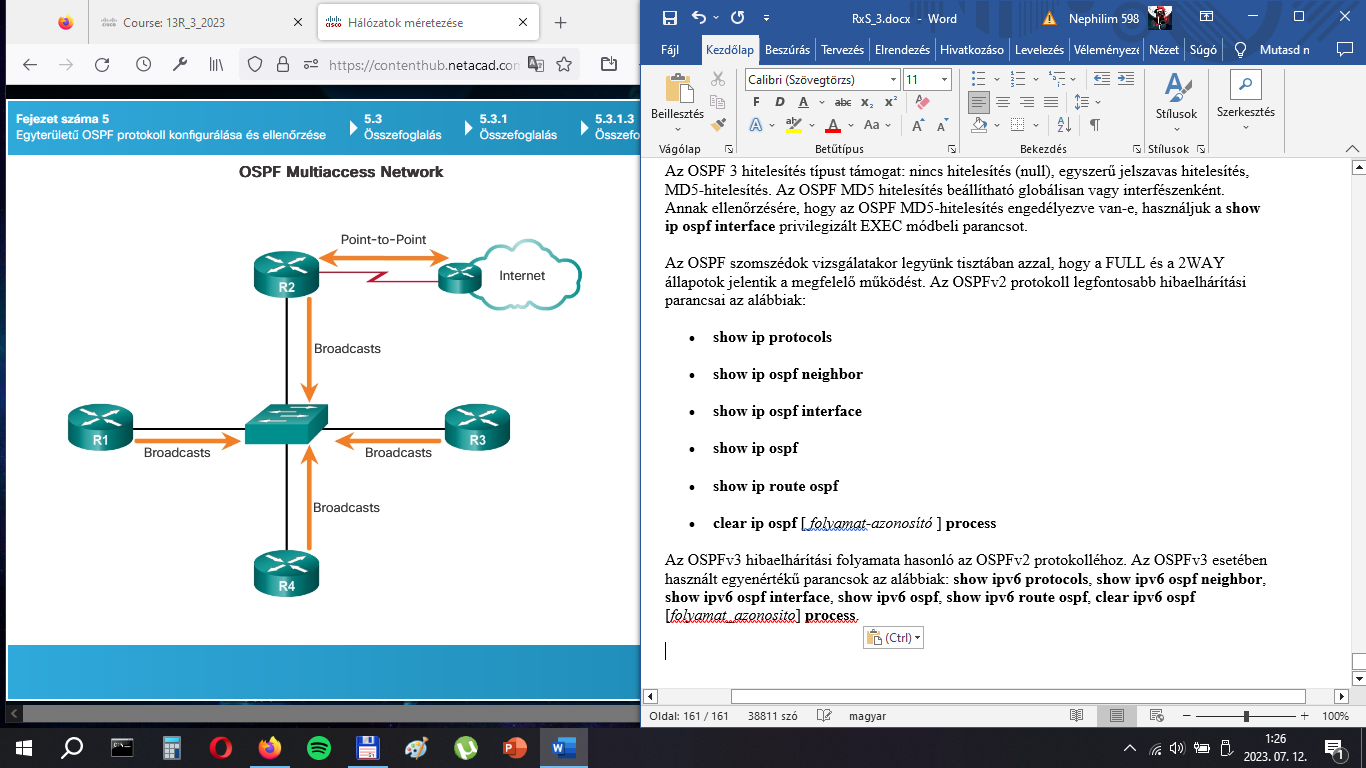
* **ip ospf hello-interval** *másodperc*
* **ip ospf dead-interval** *másodperc*
* **ipv6 ospf hello-interval** *másodperc*
* **ipv6 ospf dead-interval** *másodperc*

Az OSPF 3 hitelesítés típust támogat: nincs hitelesítés (null), egyszerű jelszavas hitelesítés, MD5-hitelesítés. Az OSPF MD5 hitelesítés beállítható globálisan vagy interfészenként. Annak ellenőrzésére, hogy az OSPF MD5-hitelesítés engedélyezve van-e, használjuk a **show ip ospf interface** privilegizált EXEC módbeli parancsot.

Az OSPF szomszédok vizsgálatakor legyünk tisztában azzal, hogy a FULL és a 2WAY állapotok jelentik a megfelelő működést. Az OSPFv2 protokoll legfontosabb hibaelhárítási parancsai az alábbiak:

* **show ip protocols**
* **show ip ospf neighbor**
* **show ip ospf interface**
* **show ip ospf**
* **show ip route ospf**
* **clear ip ospf** [ *folyamat-azonosító* ] **process**

Az OSPFv3 hibaelhárítási folyamata hasonló az OSPFv2 protokolléhoz. Az OSPFv3 esetében használt egyenértékű parancsok az alábbiak: **show ipv6 protocols**, **show ipv6 ospf neighbor**, **show ipv6 ospf interface**, **show ipv6 ospf**, **show ipv6 route ospf**, **clear ipv6 ospf** [*folyamat\_azonosito*] **process**.



* [Fejezet száma 6 Többterületű OSPF](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#6)
* [6.0 Többterületű OSPF](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#6.0)
* [6.0.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#6.0.1)
* [6.0.1.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#6.0.1.1)

# Bevezetés

A többterületű OSPF (multiarea OSPF) nagyméretű OSPF hálózatok felosztására használható. Ha egy területre túl sok forgalomirányító esne, az növelné a CPU terhelését és nagyméretű kapcsolatállapot-adatbázist (LSDB-t) eredményezne. Ez a fejezet útmutatást ad arra, hogy egy nagyméretű területet miként lehet hatékonyan több területre osztani. A 0-ás területet gerinchálózati területnek nevezzük.

A fejezet a területek között áramló kapcsolatállapot-hirdetményekre (LSA-kra) helyezi a hangsúlyt. Ezen felül OSPFv2, illetve OSPFv3 beállítási feladatok is szerepelnek a témák között. A fejezet végül az OSPF beállítások ellenőrzésére szolgáló **show** parancsok tárgyalásával zárul.

# Csoportos feladat - Sugárhajtású repülőgépre fel!

**Sugárhajtású repülőgépre fel!**

Az egyik osztálytársaddal egy új légitársaság indítását tervezitek a kontinensen.

A székhelyen vagy a bázisrepülőtéren kívül keresnetek kell a térképen négy darab kontinensen belüli, valamint egy transzkontinentális repülőtéri szolgáltatási területet, amelyek további utazási kiindulópontok vagy célpontok lehetnek.

Használjátok a rendelkezésre álló vaktérképet a repülőterek helyének megtervezéséhez. A feladat megoldásához további információkat találtok az ide tartozó PDF-ben.

[Csoportos feladat - Sugárhajtású repülőgépre fel!](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.0.1.2%20Leaving%20on%20a%20Jet%20Plane%20Instructions.pdf)

**Egyterületű OSPF (Single-Area OSPF)**

Az egyterületű OSPF olyan kisebb hálózatok esetében hasznos, ahol a forgalomirányítók kapcsolati hálója nem túl összetett, illetve az egyes célokhoz vezető útvonalak könnyen számolhatók.

Ha viszont egy terület túlságosan nagy méretűvé válik, az alábbi problémákkal is foglalkozni kell (lásd az illusztrációt):

* **Nagy méretű irányítótábla** - Az OSPF alapértelmezés szerint nem végez útvonal-összevonást. Ha az útvonalak nem összevonhatók, akkor az irányítótábla mérete rendkívül megnőhet, a hálózat méretétől függően.
* **Nagy méretű kapcsolatállapot-adatbázis (LSDB)** - Mivel az LSDB lefedi a teljes hálózat topológiáját, ezért minden egyes forgalomirányítónak fenn kell tartania egy bejegyzést a területen található összes hálózat részére, még akkor is, ha nem minden útvonal kerül be az irányítótáblába.
* **Gyakori SPF (újra)számítások** - Egy nagy méretű hálózatban elkerülhetetlenek a változások, így a forgalomirányítók is rengeteg CPU ciklust áldoznak az SPF algoritmus újraszámítására, illetve az irányítótábla frissítésére.

Az OSPF úgynevezett területeket (area) használ a hierarchikus forgalomirányítás megvalósításához, amely ezáltal még hatékonyabbá és skálázhatóbbá válik. Az OSPF terület a forgalomirányítók olyan csoportja, melyek ugyanazon kapcsolatállapot-információkat tartalmazzák az LSDB-ikben.

# Többterületű OSPF (multiarea OSPF)

Amikor egy nagy méretű OSPF területet kisebb részekre bontunk, akkor többterületű OSPF-ről beszélünk. A többterületű OSPF a nagy méretű hálózati megvalósításoknál jöhet jól, ahol csökkenteni kell a feldolgozási és a memóriaterhelést.

Például valamennyi olyan esetben, amikor egy forgalomirányító értesül egy, a területen belüli topológiaváltozásról (új kapcsolatok megjelenése, meglévők törlése vagy változása), újrafuttatja az SPF-algoritmust, egy új SPF-fát hoz létre, majd frissíti az irányítótáblát. Az SPF-algoritmus számításigényes, a számítási idő pedig függ a terület méretétől. Ha egy területre túl sok forgalomirányító esne, akkor az megnövelné az LSDB méretét és a CPU terhelését. Ezért a forgalomirányítók területek szerinti szervezésével hatékonyan oszthatunk fel egy potenciálisan nagy adatbázist több kisebb, és ezért jobban kezelhető adatbázisra.

A többterületű OSPF hierarchikus hálózattervezést igényel. A fő területet gerinchálózati területnek (area 0) nevezik, és valamennyi területnek ehhez kell kapcsolódnia. Hierarchikus forgalomirányítás esetén a területek között továbbra is zajlik a forgalomirányítás (interarea routing), viszont a számításigényes feladatok, mint például az adatbázisok újraszámítása, az egyes területeken belül maradnak.

Ahogy az 1. ábrán is látható, a többterületű OSPF hierarchikus topológiája az alábbi előnyöket nyújtja:

* **Kisebb irányítótáblák** – Az irányítótáblák bejegyzéseinek száma csökken, mert a területek között a hálózati címek összevonhatók. Például az R1 összevonhatja az 1-es területről a 2-es területre vezető útvonalakat, az R2 pedig összevonhatja az 51-es területről a 0-ás területre vezető útvonalakat. Az R1 és az R2 ezen felül hirdetheti az 1-es és az 51-es területre vezető alapértelmezett statikus útvonalat is.
* **A kapcsolatállapot-frissítések okozta többletforgalom csökkenése** - Minimalizálja a feldolgozási és memóriaigényeket, mivel kevesebb forgalomirányító vesz részt az LSA üzenetek küldésében.
* **Az SPF-számítások gyakoriságának csökkentése** – A topológiaváltozások hatásának egyetlen területre történő lokalizálása. Például csökken a forgalomirányítási frissítések okozta terhelés, mert az LSA-k elárasztása megáll a terület határán.

A 2. ábránál feltételezzük, hogy az 51-es terület két belső forgalomirányítója közötti összeköttetés meghibásodik. Ilyenkor kizárólag az 51-es területen lévő forgalomirányítók küldenek LSA-kat, illetve futtatják újra az SPF algoritmust. Az R1 nem kap LSA-kat az 51-es területről, és nem számolja újra az SPF algoritmust.

**Az OSPF kétrétegű területi hierarchiája**

A többterületű OSPF a kétrétegű területi hierarchiának megfelelően valósul meg.

* **A gerinchálózati (tranzit) terület** - Ez egy olyan OSPF terület, amelynek elsődleges feladata az IP-csomagok gyors és hatékony továbbítása. A gerinchálózati területek más OSPF területtípusokhoz kapcsolódnak. A végfelhasználók jellemzően nem a gerinchálózati területen helyezkednek el. A gerinchálózati területet az OSPF 0-ás területének (area 0) is nevezik. A hierarchikus hálózattervezés a 0-ás területet egy olyan központi területként definiálja, melyhez az összes többi területnek közvetlenül kapcsolódnia kell. (1. ábra)
* **Normál (nem gerinchálózati) terület -** Feladata a felhasználók és az erőforrások összekapcsolása. A normál területek kialakítása általában funkcionális vagy földrajzi csoportok mentén történik. Alapértelmezés szerint a normál területek nem engedik, hogy a más területekről származó forgalom az összeköttetéseiken keresztül érjen el további területeket. Minden, egyéb területről származó forgalomnak a tranzit területen keresztül kell áthaladni. (2. ábra)

**MEGJEGYZÉS:**A normál területek különböző altípusúak lehetnek, létezik például szabványos (standard), elzárt (stub), teljesen elzárt (totally stubby) és részben elzárt (not-so-stubby, NSSA) terület is. Jelen fejezet nem tárgyalja az elzárt, teljesen elzárt és részben elzárt területeket.

Az OSPF ragaszkodik ehhez a merev, kétrétegű területi hierarchiához. A hálózat alapjául szolgáló fizikai kapcsolatokat le kell képezni a kétrétegű területi struktúrára, amelyben az összes, nem gerinchálózati terület közvetlenül a 0-ás területhez kapcsolódik. Az összes, egyik területről egy másik területre haladó forgalomnak érintenie kell a gerinchálózati területet. Az ilyen forgalmat területek közötti (interarea) forgalomnak nevezzük.

Az egy területre eső forgalomirányítók optimális száma több tényezőn múlik. Ilyen például a hálózati stabilitás, de a Cisco az alábbi irányelveket javasolja:

* Egy területen legfeljebb 50 forgalomirányító legyen.
* Egy forgalomirányító legfeljebb három területhez tartozzon.
* Egy forgalomirányítónak legfeljebb 60 szomszédja legyen.

**Az OSPF forgalomirányítók típusai**

A különböző típusú OSPF forgalomirányítók vezérlik a területekre érkező, illetve onnan távozó forgalmat. Az OSPF forgalomirányítók csoportosítása az egy tartományon belül végzett feladatuk alapján történik.

Az OSPF forgalomirányítóknak négy különböző típusa van:

* **Belső forgalomirányító** - Az ilyen forgalomirányítónak minden interfésze ugyanahhoz a területhez tartozik. Az egy területen lévő összes belső forgalomirányító LSDB-je azonos. (1. ábra)
* **Gerinchálózati forgalomirányító** - Az ilyen forgalomirányítók a gerinchálózati területhez tartoznak. A gerinchálózati terület általában a 0-ás területet jelenti. (2. ábra)
* **Határ-forgalomirányító (area border router)** **(ABR)** - Az ilyen forgalomirányító egyes interfészei különböző területekhez tartoznak. Minden egyes területhez, amelyhez kapcsolódik, különálló LSDB-t tart karban, és képes a területek közötti forgalomirányításra. Az ABR egy adott terület kilépési pontja, ami azt jelenti, hogy a más területekre irányuló forgalomirányítási információk kizárólag a helyi hálózat ABR-jén keresztül juthatnak célba. Az ABR-t be lehet állítani úgy, hogy vonja össze a szomszédos területek LSDB-jeiből származó útvonal-információkat. Az útvonal-információk gerinchálózat felé történő hirdetését is az ABR végzi. A gerinchálózati forgalomirányítók ezt az információt továbbítják a többi ABR felé. Többterületű hálózatok esetében bármely területhez egy vagy több ABR tartozhat. (3. ábra)
* **Autonóm rendszer határ-forgalomirányítója (ASBR, autonomous system boundary router)** - Az ilyen forgalomirányítónak legalább az egyik interfésze egy külső hálózathoz (egy másik autonóm rendszerhez) van csatlakoztatva. Ez természetesen lehet egy nem OSPF-et használó hálózat is. Az ASBR képes importálni a nem OSPF-et használó hálózatok információit egy OSPF hálózatba az útvonal-újrahirdetésnek nevezett folyamat segítségével, és ugyanez működik visszafelé is. (4. ábra)

Újrahirdetés akkor fordul elő többterületű OSPF környezetben, amikor az ASBR eltérő forgalomirányítást (pl.: EIGRP-t és OSPF-et) használó tartományokat köt össze.

Egy forgalomirányító többféle forgalomirányító-típusba is besorolható. Ha például egy forgalomirányító kapcsolódik a 0-ás és az 1-es területhez, ezen felül pedig útvonal-információkkal rendelkezik egy másik, nem OSPF-et használó hálózathoz, akkor három különböző csoportba is beletartozik: gerinchálózati forgalomirányító, ABR és ASBR is egyszerre.

# Az OSPF protokoll LSA típusai

Az LSA-k az OSPF protokoll LSDB-jének szerves alkotóelemei. Külön-külön úgy viselkednek, mint az adatbázis-rekordok, az OSPF hálózat konkrét részleteit ismerik. Együtt pedig leírják egy OSPF hálózat vagy terület teljes topológiáját.

Az OSPF-re vonatkozó szabványtervezetek jelenleg összesen 11 LSA típust határoznak meg. (1. ábra) A többterületű OSPF bármely megvalósításának támogatnia kell az első öt LSA-t: az 1-es LSA-tól az 5-ös LSA-ig (2. ábra) Ez a témakör elsődlegesen ezzel az öt LSA-val foglalkozik.

A forgalomirányító minden egyes összeköttetése egy LSA típusként definiálható. Az LSA tartalmaz egy kapcsolatazonosító (link ID) mezőt, amely -- a hálózat címével és maszkjával -- azonosítja az objektumot, amihez a kapcsolat vezet. Típustól függően, a kapcsolatazonosító különböző jelentéssel bírhat. Az LSA-k különböznek az előállításuk és a forgalomirányítási tartományon belüli terjedésük módja szerint.

**MEGJEGYZÉS:**Az OSPFv3 további LSA típusokat tartalmaz.

# 1-es típusú OSPF LSA

Ahogy az ábrán is látható, az összes forgalomirányító 1-es típusú LSA-val hirdeti a hozzá közvetlenül kapcsolódó, OSPF számára engedélyezett összeköttetéseket, és továbbítja ezen hálózati információkat az OSPF szomszédok felé. Az LSA tartalmaz egy listát a közvetlenül kapcsolódó interfészekről, kapcsolattípusokról és -állapotokról.

Az 1-es típusú LSA-kat a forgalomirányító kapcsolati bejegyzéseinek is nevezzük.

Az 1-es típusú LSA csak azt a területet árasztja el, ahonnan származik. Az ABR forgalomirányítók később 3-as típusú LSA-ként hirdetik tovább az 1-es típusú LSA-tól tanult hálózatokat.

Az 1-es típusú LSA kapcsolatazonosítóját a küldő forgalomirányító azonosítója (router ID) határozza meg.

# 2-es típusú OSPF LSA

A 2-es típusú LSA-t csak többszörös hozzáférésű és nem-szórásos többszörös hozzáférésű (NBMA) hálózatokban használják, ahol van DR forgalomirányító-választás, és legalább két forgalomirányító van a többszörös hozzáférésű szegmensben. A 2-es típusú LSA tartalmazza a forgalomirányító azonosítóját és a DR forgalomirányító IP-címét, valamint a többszörös hozzáférésű szegmensben található összes forgalomirányító azonosítóját. A területhez tartozó minden többszörös hozzáférésű hálózathoz létrejön egy 2-es típusú LSA.

A 2-es típusú LSA célja, hogy informálja a többi forgalomirányítót az egyazon területen belül található többszörös hozzáférésű hálózatokról.

A DR csak azt a területet árasztja el 2-es típusú LSA-kal, ahonnan azok származnak. A 2-es típusú LSA-k nem jutnak a területen kívülre.

A 2-es típusú LSA-kat a hálózati kapcsolatok bejegyzéseinek is nevezzük.

Ahogy az ábrán is látható, az ABR1 forgalomirányító az 1-es területen található Ethernet hálózat DR forgalomirányítója. Ez állítja elő a 2-es típusú LSA-t, amit majd továbbít az 1-es területre. Az ABR2 forgalomirányító a 0-ás területen található többszörös hozzáférésű hálózat DR forgalomirányítója. Mivel a 2-es területen nincsenek többszörös hozzáférésű hálózatok, ezért ott sohasem kell 2-es típusú LSA-kat hirdetni.

Egy hálózati LSA kapcsolatállapot-azonosítója mindig az azt hirdető DR forgalomirányító interfészének IP-címe.

# 3-as típusú OSPF LSA

A 3-as típusú LSA-t az ABR forgalomirányítók használják más területek hálózatainak hirdetésére. Az ABR forgalomirányítók az LSDB-be gyűjtik az 1-es típusú LSA-kat. Miután egy adott OSPF terület konvergált, az ABR minden egyes tanult OSPF hálózathoz létrehoz egy 3-as típusú LSA-t. Ezért a sok OSPF útvonalat ismerő ABR forgalomirányítónak minden egyes hálózathoz létre kell hoznia egy 3-as típusú LSA-t.

Ahogy az ábrán is látható, az ABR1 és az ABR2 az egyik területről 3-as típusú LSA-val árasztja el a többi területet. A 3-as típusú LSA-kat az ABR forgalomirányítók hirdetik a többi terület felé. Egy sok hálózattal rendelkező OSPF megvalósítás esetében a 3-as típusú LSA-k hirdetése jelentős elárasztásos problémákat okozhat. Ezért nagyon ajánlott, hogy az ABR forgalomirányítókon manuális útvonal-összevonást állítsunk be.

A kapcsolatállapot azonosítója megkapja a hálózat számát, de a maszk is hirdetve lesz.

Attól, hogy egy forgalomirányító 3-as típusú LSA-t kap a területére, még nem futtatja újra az SPF algoritmust. A 3-as típusú LSA-val hirdetett útvonalak megfelelően bekerülnek a forgalomirányító irányítótáblájába vagy törlődnek onnan, de nincs szükség az SPF teljes újraszámítására.

# 4-es típusú OSPF LSA

A 4-es és 5-ös típusú LSA-k együttesen azonosítanak egy ASBR forgalomirányítót, valamint külső hálózatokat hirdetnek befelé az OSPF irányítású tartományba.

4-es típusú összevont LSA-t csak akkor állít elő az ABR, ha egy ASBR is létezik a területen belül. A 4-es típusú LSA azonosítja az ASBR forgalomirányítót, és útvonalat kínál hozzá. Minden olyan forgalom számára, amely külső autonóm rendszer felé irányul, szükség van a külső útvonalakat küldő ASBR irányítótáblájának ismeretére.

Ahogy az ábrán is látható, az ASBR egy 1-es típusú LSA küldésével ASBR forgalomirányítóként mutatkozik be. Az LSA tartalmaz egy speciális bitet, amelyet külső bitnek (e-bitnek) is nevezünk, és amelynek használatával a forgalomirányító ASBR forgalomirányítóként jelentkezhet be. Amikor az ABR1 megkapja az 1-es típusú LSA-t, érzékeli az e-bitet, létrehoz egy 4-es típusút, majd hirdeti ezt az LSA-t a gerinchálózat felé (a 0-ás területre). Az utána következő ABR forgalomirányítók 4-es típusú LSA-val árasztják el a többi területet.

A kapcsolatállapot azonosítója az ASBR forgalomirányító azonosítója lesz.

# 5-ös típusú OSPF LSA

Az 5-ös típusú LSA azokat az útvonalakat határozza meg, amelyek az OSPF irányítású autonóm rendszeren kívül eső hálózatokhoz vezetnek. Az ASBR forgalomirányítótól származó 5-ös típusú LSA-k elárasztják a teljes autonóm rendszert.

Az 5-ös típusú LSA-kat az autonóm rendszer külső LSA bejegyzéseinek is nevezzük.

Az ábrán látható, ahogy az ASBR minden egyes külső útvonalhoz előállít egy 5-ös típusú LSA-t, majd elárasztja velük a területet. Az utána következő ABR forgalomirányítók szintén 5-ös típusú LSA-val árasztják el a többi területet. A más területeken található forgalomirányítók arra használják a 4-es típusú LSA információit, hogy elérjék a külső útvonalakat.

Egy sok hálózattal rendelkező OSPF megvalósítás esetében az 5-ös típusú LSA-k nagy számban történő hirdetése jelentős elárasztásos problémákat okozhat. Ezért nagyon ajánlott, hogy az ASBR forgalomirányítókon manuális útvonal-összevonást állítsunk be.

A kapcsolatállapot azonosítója a külső hálózat címe lesz.

# Az OSPF irányítótáblájának bejegyzései

Az 1. ábrán egy olyan többterületű OSPF topológiához tartozó irányítótábla látható, amely egy külső, nem OSPF-et használó hálózathoz kapcsolódik. Az IPv4-es irányítótábla OSPF útvonalait az alábbi jellemzőkkel lehet megadni:

* **O** - a kapcsolati (1-es típusú) és hálózati (2-es típusú) LSA-k megadják az adott területen belüli részleteket. Az irányítótábla ezen kapcsolatállapot-információkat egy **O**betűvel jelöli, ami területen belüli (intra-area) útvonalat jelent.
* **O IA** - Amikor az ABR összevont LSA-kat kap, ezeket hozzáadja az LSDB-jéhez, majd újragenerálja azokat a helyi hálózat felé. - Amikor az ABR külső LSA-kat kap, ezeket hozzáadja az LSDB-jéhez, majd elárasztja velük a területet. A belső forgalomirányítók ezt az információt beolvasztják az adatbázisukba. Az összevont LSA-k az irányítótáblában IA (interarea, területek közötti útvonal) rövidítéssel vannak jelölve.
* **O E1** vagy **O E2** - A külső LSA-k az irányítótáblában 1-es típusú külső (E1) vagy 2-es típusú külső (E2) útvonalként szerepelnek.

A 2. ábrán egy OSPF forgalomirányító IPv6-os irányítótáblája látható, területek közötti és külső irányítótábla-bejegyzésekkel.

# OSPF útvonalszámítás

Az SPF-fa felépítéséhez minden egyes forgalomirányító az SPF-algoritmust futtatja le az LSDB tartalmán. Az SPF-fa szolgál a legjobb útvonalak meghatározására.

Ahogy az ábrán is látható, a legjobb útvonalak kiszámítása az alábbi sorrend alapján történik:

1. Az összes forgalomirányító kiszámítja a területén belüli célokhoz vezető legjobb útvonalakat, majd hozzáadja ezen bejegyzéseket az irányítótáblájához. Ezek 1-es, illetve 2-es típusú LSA-k, amit az irányítótáblában az O rövidítés jelöl. (1)

2. Az összes forgalomirányító kiszámítja a többi -- hálózaton belüli -- területre vezető legjobb útvonalakat. Az ilyen legjobb útvonalak területek közötti útvonalbejegyzések, más néven 3-as vagy 4-es típusú LSA-k, és az O IA rövidítés jelöli ezeket. (2)

3. Az összes forgalomirányító (kivéve az elzárt területekhez tartozókat) kiszámítja a külső autonóm rendszerekhez vezető (5-ös típusú) legjobb útvonalakat. Ezeket -- a beállításoktól függően -- az O E1 vagy O E2 rövidítés jelöli. (3)

Konvergálás után a forgalomirányítók bármely hálózattal tudnak kommunikálni, az OSPF autonóm rendszeren belül vagy kívül.

# A többterületű OSPF megvalósítása

Az OSPF megvalósítása lehet egyterületű vagy többterületű. A kiválasztott OSPF megvalósítás függ a konkrét követelményektől, illetve a meglévő topológiától.

Ahogy az ábrán is látszik, a többterületű OSPF megvalósítása 4 lépésből áll.

Az 1. és 2. lépés a tervezési folyamat részét képezi.

**1. lépés: Hálózati követelmények és paraméterek összegyűjtése -** Ide tartozik a végfelhasználói és hálózati berendezések számának, az IP-címzés elvének (amennyiben már meglévő megvalósításról van szó), a forgalomirányítási tartomány és az irányítótábla méretének, a topológiaváltozások kockázatának, valamint a hálózat további jellemzőinek meghatározása.

**2.lépés: OSPF paraméterek megadása -** Az 1. lépés során begyűjtött információk alapján a hálózati rendszergazdának el kell döntenie, hogy az egyterületű vagy többterületű OSPF megvalósítás a megfelelőbb. Amennyiben a többterületű OSPF-re esik a választás, a hálózati rendszergazdának számos tényezőt kell figyelembe venni az OSPF paraméterek meghatározásánál. Ilyenek például:

* **IP címzési séma** - Ez határozza meg az OSPF alkalmazásának módját, illetve azt, hogy az mennyire skálázható. Létre kell hozni egy részletes IP címzési sémát, amely tartalmazza az alhálózati információkat is. Egy jó IP címzési séma lehetővé teszi az OSPF többterületű tervének elkészítését, illetve az összevonások használatát. Egy ilyen sémával nemcsak a hálózat skálázható sokkal jobban, hanem az OSPF működése és az LSA-k hirdetése is optimalizálható.
* **OSPF területek** - Az OSPF hálózat területekre osztása csökkenti az LSDB méretét, és a topológia megváltozása esetén korlátozza a kapcsolatállapot-frissítések hirdetését. Meg kell határozni a leendő ABR és ASBR forgalomirányítókat, mivel azok végeznek bármilyen összevonást, illetve újrahirdetést.
* **A hálózati topológia** - Olyan összeköttetésekből áll, amelyek összekapcsolják a hálózati berendezéseket, és a többterületű OSPF tervnek megfelelően különböző OSPF területekhez tartoznak. A hálózati topológia fontos szerepet játszik az elsődleges és tartalék kapcsolatok meghatározásában. Az elsődleges és tartalék kapcsolatokat az interfészekhez tartozó OSPF költségek változtatásával adjuk meg. Szükség van egy részletes hálózati topológiára is, amellyel meghatározhatók a különböző OSPF területek, az ABR illetve az ASBR csakúgy, mint az összevonási és újrahirdetési pontok.

**3. lépés:** A többterületű OSPF megvalósítás beállítása a paraméterek alapján

**4. lépés:** A többterületű OSPF megvalósítás ellenőrzése a paraméterek alapján

**A többterületű OSPF beállítása**

Az 1. ábrán a referenciaként használt többterületű OSPF topológia látható. A példában a következő esetek fordulnak elő:

* Az R1 egy ABR forgalomirányító, mivel van több 1-es területhez tartozó, illetve egy 0-ás területhez tartozó interfésze.
* Az R2 egy belső, gerinchálózati forgalomirányító, mivel minden interfésze a 0-ás területhez tartozik.
* Az R3 egy ABR forgalomirányító, mivel van több 2-es területhez tartozó, illetve egy 0-ás területhez tartozó interfésze.

Ennek a többterületű OSPF hálózatnak a megvalósításához nincs szükség speciális parancsokra. Egy forgalomirányítóból mindössze két, különböző területre vonatkozó **network** parancs kiadásával ABR válhat.

Ahogy a 2. ábrán is látható, az R1-hez az 1.1.1.1 forgalomirányító-azonosító van hozzárendelve. Ez a példa lehetővé teszi az OSPF számára, hogy a két LAN-interfész az 1-es területhez tartozzon. A soros interfész úgy van beállítva, hogy az OSPF 0-ás területének részét képezze. Mivel az R1 két interfésze két különböző területhez kapcsolódik, ezért ez egy ABR forgalomirányító.

A 3. ábrán parancsszimulátorral végezzük el az R2 és az R3 forgalomirányítók többterületű OSPF beállításait! A parancsszimulátorban az R2-n használjuk az interfész hálózati címének helyettesítő maszkját. Az R3-on adjuk meg a 0.0.0.0 helyettesítő maszkot az összes hálózat felé.

Az R2 beállítását követően figyeljük meg az R1 (1.1.1.1) szomszédságáról tájékoztató üzeneteket.

Az R3 beállítását követően figyeljük meg az R2 (2.2.2.2) szomszédságáról tájékoztató üzeneteket. Azt is figyeljük meg, hogy a címzési sémát követő forgalomirányító-azonosítók használata megkönnyíti a szomszédok azonosítását.

**MEGJEGYZÉS:**Az R2 és R3 beállításához használt inverz helyettesítő maszkok szándékosan különböznek, hogy demonstráljuk a **network** parancsok bevitelének két lehetséges módját. Az R3 esetében használt módszer egyszerűbb, mivel a helyettesítő maszk mindig **0.0.0.0** , így nem kell kiszámítani.

# A többterületű OSPFv3 beállítása

Az OSPFv2-höz hasonlóan, az 1. ábrán látható többterületű OSPFv3 topológia megvalósítása is egyszerű. Nincs szükség speciális parancsokra. Egy forgalomirányítóból egyszerűen ABR válik, amennyiben két, különböző területekhez tartozó interfésszel rendelkezik.

A 2. ábrán látható példában az R1-hez az 1.1.1.1 forgalomirányító-azonosító van hozzárendelve. A példában az OSPF engedélyezve van az 1-es területhez tartozó LAN interfészen, valamint a 0-ás területhez tartozó soros interfészen. Mivel az R1 két interfésze két különböző területhez kapcsolódik, ezért ABR forgalomirányítóvá válik.

A 3. ábrán látható parancsszimulátorral végezzük el az R2 és az R3 forgalomirányítók többterületű OSPFv3 beállításait!

Az R2 beállítását követően figyeljük meg az R1 (1.1.1.1) szomszédságáról tájékoztató üzenetet.

Az R3 beállítását követően figyeljük meg az R2 (2.2.2.2) szomszédságáról tájékoztató üzenetet.

# OSPF útvonal-összevonás (route summarization)

Az útvonal-összevonás segít abban, hogy az irányítótáblák mérete kicsi maradjon. Tulajdonképpen több útvonalat fog össze egyetlen hirdetménybe, amelyet azután hirdetni lehet a gerinchálózati terület felé.

Normál esetben, az egyes területeken belül 1-es, illetve 2-es LSA-k előállítása történik, melyek azután 3-as típusú LSA-kká alakítva kerülnek kiküldésre a többi terület irányába. Ha az 1-es terület 30 hálózatot szeretne hirdetni, akkor 30 db 3-as típusú LSA-t kellene továbbítani a gerinchálózat felé. Útvonal-összevonás segítségével az ABR 30 hálózatot fog össze a két hirdetmény egyikébe.

Az 1. ábrán az látszik, hogy az R1 az összes hálózati hirdetményt egyetlen összevont LSA-ba fogja össze. Ahelyett, hogy az 1-es terület mindegyik útvonalához külön LSA-t kellene továbbítani, az R1 egy összevont LSA-t küld a C1 gerinchálózati forgalomirányítónak. A C1 ezután továbbítja az összevont LSA-t az R2 és az R3 felé. Ezt az R2 és az R3 továbbküldi a hozzájuk tartozó belső forgalomirányítóknak.

Az útvonal-összevonás a hálózati stabilitás növelésében is segít, mivel csökkenti a felesleges LSA szórást. Ez viszont közvetlenül kihat az OSPF irányítási folyamata által felhasznált sávszélesség, CPU és memória mennyiségére. Útvonal-összevonás nélkül minden egyes konkrét kapcsolatról külön LSA hirdetmény kerül kiküldésre az OSPF gerinchálózatba és azon túlra, felesleges hálózati forgalmat és a forgalomirányító pluszterhelését eredményezve.

A 2. ábrán az látszik, hogy az R1a forgalomirányító egyik hálózati kapcsolata meghibásodik. Az R1a forgalomirányító egy LSA-t küld az R1-nek. Az R1 ennek ellenére nem hirdeti tovább a frissítést, mivel rendelkezik egy, már beállított összevont útvonallal. A területen kívülre az adott kapcsolatról LSA szórás nem történik.

**Területek közötti és külső útvonalak összevonása**

OSPF környezetben útvonal-összevonás kizárólag az ABR és ASBR forgalomirányítókon állítható be. A sok konkrét hálózat hirdetése helyett az ABR és ASBR forgalomirányítók egy összevont útvonalat hirdetnek. Az ABR forgalomirányítók 3-as típusú, az ASBR forgalomirányítók pedig 5-ös típusú LSA-val vonják össze az útvonalakat.

Alapértelmezés szerint az összevont (3-as típusú) LSA és a külső (5-ös típusú) LSA nem tartalmaz összevont (aggregált) útvonalakat. Más szóval, alapértelmezés szerint az összevont LSA már nem tartalmaz további összevonásokat.

Ahogy az 1. és 2. ábrán is látható, az útvonal-összevonás beállítása az alábbiak szerint történik:

* **Területek közötti (interarea) útvonal-összevonás** - Területek közötti útvonal-összevonás az ABR forgalomirányítókon történik, és az egyes területeken belüli útvonalakra vonatkozik. AZ OSPF-be újrahirdetéssel behozott külső útvonalakra viszont nem vonatkozik. A területek közötti hatékony útvonal-összevonás érdekében az egyes területekhez tartozó hálózati címeket folytonosan kell kiosztani, hogy az összevonásukhoz minél kevesebb összevont cím kelljen.
* **Külső útvonalak összevonása** - A külső útvonalak összevonása csak az OSPF-be újrahirdetéssel behozott külső útvonalakra vonatkozik. Itt is fontos, hogy az összevonandó külső címtartományok folytonossága biztosítva legyen. Általában csak az ASBR vonja össze a külső útvonalakat. Ahogy a 2. ábrán is látható, a külső EIGRP útvonalakat az R2 (ASBR) forgalomirányító egyetlen LSA-ba vonja össze, amit továbbküld az R1-nek és az R3-nak.

**MEGJEGYZÉS:**A külső útvonalak összevonásának beállítása az ASBR forgalomirányítókon kiadott **summary-address** *címmaszk (address mask)* forgalomirányító-konfigurációs paranccsal történik.

# Területek közötti (interarea) útvonal-összevonás

Az OSPF nem végez automatikus útvonal-összevonást. A területek közötti útvonal-összevonást manuálisan kell beállítani az ABR forgalomirányítókon.

A belső útvonalak -- területek közötti továbbítás céljából történő -- összevonása kizárólag az ABR forgalomirányítókon végezhető el. Amennyiben az ABR forgalomirányítón megtörtént az útvonal-összevonás beállítása, a gerinchálózati területre egy 3-as típusú (az összevont útvonalat leíró) LSA lesz beküldve. Ez az LSA egy területen belül több útvonalat is összefog.

Az összevont útvonal akkor áll elő, ha a területen belüli alhálózatok legalább egyike az összevont címtartományba esik. Az összevont útvonal metrikája megegyezik az összevont címtartományon belüli alhálózatok legalacsonyabb költségével.

**MEGJEGYZÉS:**Az ABR csak olyan útvonalakat vonhat össze, amelyek az ABR forgalomirányítóhoz kapcsolódó területeken belül találhatók.

Az 1. ábrán egy többterületű OSPF topológia látható. Az R1 és R2 irányítótábláit vizsgálva láthatjuk az útvonal-összevonás hatását.

A 2. ábra az R1, a 3. ábra pedig az R3 irányítótábláját mutatja az útvonal-összevonás beállítása előtt. Figyeljük meg, hogy az R3-nak jelenleg két, területek közötti bejegyzése van az R1 forgalomirányító 1-es területű hálózataihoz.

# Az összevont útvonal kiszámítása

Az ábra azt mutatja, hogy a hálózatok összevonása egyetlen címmé és maszkká három lépésben történik:

**1. lépés:** Az érintett hálózati címek felírása bináris formában. A példában két, 1-es területhez tartozó hálózat szerepel bináris formában: a 10.1.1.0/24 és a 10.1.2.0/24.

**2. lépés:** A balról megegyező bitek számának megállapítása az összevont útvonal maszkjának meghatározásához. A 22 darab balról megegyező bit van megjelölve. Ez eredményezi a **/22** előtagot vagy a **255.255.252.0**alhálózati maszkot.

**3. lépés:** Az egyező bitek kiegészítése 0 bitértékekkel 32 bit hosszúságúra az összevont hálózati cím meghatározásához. (A nem megegyező biteket 0 bitekkel helyettesítjük.) Ebben a példában az egyező bitek nullákkal kiegészítve a 10.1.0.0/22 hálózati címet adják eredményül. Az összevont cím négy hálózatot tartalmaz: ezek a 10.1.0.0/24, 10.1.1.0/24, 10.1.2.0/24, valamint a 10.1.3.0/24.

A példában szereplő összevont cím négy hálózatra illeszkedik, bár csupán két hálózat létezik.

# A területek közötti útvonal-összevonás beállítása

Az 1. ábrán -- az útvonal-összevonás hatásait demonstrálandó -- az R1 úgy van beállítva, hogy az 1-es területhez tartozó belső útvonalakat vonja össze.

Az útvonal-összevonás ABR forgalomirányítókon történő manuális beállításához használjuk az **area** *terület\_azonosító* **range** *címmaszk (address mask)* forgalomirányító-konfigurációs parancsot. Ez utasítja az ABR forgalomirányítót az egy konkrét területhez tartozó útvonalak összevonására, mielőtt a gerinchálózaton keresztül (3-as típusú összevont LSA-ként) beküldené azokat egy másik területre.

**MEGJEGYZÉS:**Az OSPFv3 esetében ugyanazt a parancsot használjuk, az IPv6-os hálózati cím kivételével. Az OSPFv3-ra vonatkozó parancs szintaktikája az alábbi: **area** *terület\_azonosító* **range** *előtag/előtag\_hosszúsága*.

A 2. ábrán látható az 1-es területhez tartozó két belső útvonal egyetlen, OSPF területek közötti összevont útvonalként történő összefogása az R1-en. A 10.1.0.0/22 összevont útvonal valójában négy hálózat címét fogja össze: ezek a 10.1.0.0/24-től a 10.1.3.0/24-ig terjednek.

A 3. ábrán az R1 aktuális IPv4-es irányítótáblájából származó OSPF útvonalak láthatók. Figyeljük meg, hogy új bejegyzés jelent meg Null0 kimenő interfésszel. A Cisco IOS automatikusan létrehozza a Null0 interfészhez tartozó összevont útvonalat az irányítási hurkok elkerülése céljából, amennyiben manuális útvonal-összevonás van beállítva. A null interfészre küldött csomagok eldobásra kerülnek.

Tételezzük fel, hogy az R1 kapott egy csomagot, melynek célja a 10.1.0.10 cím volt. Bár a cím illeszkedne az R1 összevont útvonalára, az R1-nek nincs még egy érvényes útvonala az 1-es területen. Ezért az R1 az irányítótábla következő leghosszabb egyezésére fog hivatkozni, amely a Null0 bejegyzés. A csomag továbbítása megtörténik a Null0 interfészre, majd eldobásra kerül. Ez akadályozza meg a forgalomirányítót, hogy a csomagot az alapértelmezett útvonalra továbbítsa, és ezzel egy lehetséges irányítási hurkot idézzen elő.

A 4. ábra az R3 aktualizált irányítótábláját mutatja. Figyeljük meg, hogy most már csak egyetlen területek közötti bejegyzés szerepel a 10.1.0.0/22 összevont útvonalhoz. Bár ez a példa csupán egy bejegyzéssel csökkentette az irányítótábla méretét, az útvonal-összevonás megvalósítása számos hálózat összefogását tenné lehetővé. Ez szintén hozzájárulna az irányítótábla méretének csökkentéséhez.

Az 5. ábrán található parancsszimulátorban az R3 forgalomirányító 2-es területhez tartozó útvonalait vonhatjuk össze.

**A többterületű OSPF ellenőrzése**

Az egyterületű OSPF ellenőrzéséhez használt parancsok az ábrán látható többterületű OSPF topológia ellenőrzéséhez is használhatók:

* **show ip ospf neighbor**
* **show ip ospf**
* **show ip ospf interface**

Az alábbi parancsok konkrét, többterületű információk ellenőrzésére használhatók:

* **show ip protocols**
* **show ip ospf interface brief**
* **show ip route ospf**
* **show ip ospf database**

**MEGJEGYZÉS:**A fentieknek megfelelő OSPFv3 parancsok esetében egyszerűen helyettesítsük az **ip** kulcsszót az **ipv6**kulcsszóval.

# Az általános többterületű OSPF beállítások ellenőrzése

Használjuk a **show ip protocols** parancsot az OSPF állapotának ellenőrzésére. A parancskimenet megmutatja, hogy mely irányítóprotokollok vannak beállítva a forgalomirányítón. Ezen kívül megmutatja az irányítóprotokoll sajátosságait is. Ilyen például a forgalomirányító azonosítója (router ID), a forgalomirányítóhoz tartozó területek száma, valamint az irányítóprotokoll konfigurációjában szereplő hálózatok.

Az 1. ábrán az R1 forgalomirányító OSPF beállításai láthatók. Figyeljük meg, hogy a parancs kimenetében két terület szerepel. A Routing for Networks résznél szerepelnek a hálózatok, valamint a hozzájuk tartozó területek.

Használjuk a **show ip ospf interface brief** parancsot az OSPF-et támogató interfészeknek a protokollhoz kapcsolódó tömör információinak megjelenítéséhez. Ez a parancs olyan hasznos információkat mutat meg, mint például az interfészhez hozzárendelt OSPF folyamatazonosító (process ID), azon területek, amelyekhez az interfészek tartoznak, valamint az interfész költsége.

A 2. ábra az OSPF-et támogató interfészek, valamint a hozzájuk tartozó területek ellenőrzését mutatja.

A 3. ábrán található parancsszimulátorban vizsgáljuk meg az R2 és R3 általános beállításait!

# Az OSPF útvonalak ellenőrzése

A többterületű OSPF beállításainak ellenőrzésére használt leggyakoribb parancs a **show ip route** . Az **ospf** paraméter hozzáadásával a csak OSPF-re vonatkozó információk jeleníthetők meg.

Az 1. ábra az R1 irányítótábláját mutatja. Figyeljük meg, hogy az irányítótábla **O IA** bejegyzései miként azonosítják a más területektől tanult útvonalakat. Konkrétan, az **O** jelöli az OSPF útvonalakat, az **IA** pedig a területek közötti (interarea) útvonalakat jelöli. Ez utóbbi azt jelenti, hogy az útvonal egy másik területről származik. Emlékezzünk vissza, hogy az R1 forgalomirányító a 0-ás területhez, az R3-hoz kapcsolódó 192.168.1.0 és 192.168.2.0 alhálózatok pedig a 2-es területhez tartoznak. Az irányítótáblában szereplő [110/1295] bejegyzés az OSPF-hez tartozó adminisztratív távolságot (110), valamint az útvonalak összes költségét ( 1295) jelöli.

Ellenőrizzük a 2. ábrán látható parancsszimulátorban az R2 és R3 forgalomirányítók irányítótábláit a **show ip route ospf** parancs segítségével!

# A többterületű OSPF LSDB-jének ellenőrzése

Használjuk a **show ip ospf database** parancsot az LSDB tartalmának ellenőrzéséhez!

Számos opció áll rendelkezésre a **show ip ospf database** parancshoz.

Az 1. ábrán látható példa az R1 forgalomirányító LSDB-jének tartalmát jeleníti meg. Figyeljük meg, hogy az R1-nek vannak a 0-ás és az 1-es területre vonatkozó bejegyzései is, mivel az ABR forgalomirányítóknak minden olyan területhez, amelyhez tartoznak, külön LSDB-t kell fenntartaniuk. A parancs kimenetének a 0-ás területre vonatkozó Router Link States részében három forgalomirányító szerepel. A Summary Net Link States részben a más területektől tanult hálózatok, valamint az ezeket hirdető szomszédok szerepelnek.

Ellenőrizzük a 2. ábrán látható parancsszimulátorban az R2 és R3 forgalomirányítók LSDB-jét a **show ip ospf database** parancs segítségével! Az R2 forgalomirányítónak csak a 0-ás területhez tartozó interfészei vannak, ezért itt csak egy LSDB szükséges. Az R3 forgalomirányítónak -- az R1-hez hasonlóan -- két LSDB-je van.

# A többterületű OSPFv3 ellenőrzése

Az OSPFv3 az OSPFv2-höz hasonló ellenőrző parancsokat használ. Nézzük meg az 1. ábrán látható OSPFv3 minta topológiát.

A 2. ábrán az R1 forgalomirányító OSPFv3 beállításai láthatók. Figyeljük meg, hogy a parancs kimenetében jelenleg két terület szerepel. Azt is megtudhatjuk, hogy mely interfészek vannak engedélyezve az egyes területekhez.

A 3. ábra az OSPFv3-at engedélyező interfészek, valamint a hozzájuk tartozó területek ellenőrzését mutatja.

A 4. ábra az R1 irányítótábláját jeleníti meg. Figyeljük meg, hogy az IPv6-os irányítótábla **OI** bejegyzései miként azonosítják a más területektől tanult útvonalakat. Konkrétan, az **O** jelöli az OSPF útvonalakat, az **I** pedig a területek közötti (interarea) útvonalakat jelöli. Ez utóbbi azt jelenti, hogy az útvonal egy másik területről származik. Emlékezzünk vissza, hogy az R1 forgalomirányító a 0-ás területhez, az R3-hoz kapcsolódó 2001:DB8:CAFE3::/64 alhálózat pedig a 2-es területhez tartozik. Az irányítótáblában szereplő [110/1295] bejegyzés az OSPF-hez tartozó adminisztratív távolságot (110), valamint az útvonalak összes költségét ( 1295) jelöli.

Az 5. ábra az R1 forgalomirányító LSDB-jének tartalmát jeleníti meg. A parancs az OSPFv2-es megfelelőjéhez hasonló információkat ad eredményül. Ugyanakkor az OSPFv3 protokoll LSDB-je olyan további LSA típusokat is tartalmaz, amelyek az OSPFv2-ben nem elérhetők.

# Packet Tracer - Többterületű OSPFv2 beállítása

**Háttér / esetleírás**

Ebben a feladatban egy többterületű OSPFv2-t fogunk beállítani. A hálózat fizikai kialakítása és az interfészek IPv4-címekkel történő beállítása már megtörtént. A mi feladatunk a többterületű OSPFv2 engedélyezése, a kapcsolatok ellenőrzése és a többterületű OSPFv2 működésének vizsgálata.

[Packet Tracer - Configuring Multiarea OSPFv2 Instructions](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.2.3.6%20Packet%20Tracer%20-%20Configuring%20Multiarea%20OSPFv2%20Instructions.pdf)

[Packet Tracer - Configuring Multiarea OSPFv2 - PKA](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.2.3.6%20Packet%20Tracer%20-%20Configuring%20Multiarea%20OSPFv2.pka)

# Packet Tracer - Többterületű OSPFv3 beállítása

**Háttér / esetleírás**

Ebben a feladatban egy többterületű OSPFv3-at fogunk beállítani. A hálózat fizikai kialakítása és az interfészek IPv6-os címekkel történő beállítása már megtörtént. A mi feladatunk a többterületű OSPFv3 engedélyezése, a kapcsolatok ellenőrzése és a többterületű OSPFv3 működésének vizsgálata.

[Packet Tracer - Configuring Multiarea OSPFv3 Instructions](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.2.3.7%20Packet%20Tracer%20-%20Configuring%20Multiarea%20OSPFv3%20Instructions.pdf)

[Packet Tracer - Configuring Multiarea OSPFv3 - PKA](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.2.3.7%20Packet%20Tracer%20-%20Configuring%20Multiarea%20OSPFv3.pka)

# Laborgyakorlat - Többterületű OSPFv2 beállítása

**Ebben a gyakorlatban a következő feladatokat végezzük el:**

* 1. rész: A hálózat felépítése és az eszközök alapvető konfigurálása.
* 2. rész: Többterületű OSPFv2 hálózat beállítása
* 3. rész: Területek közötti összevont útvonalak beállítása

[Lab - Configuring Multiarea OSPFv2](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.2.3.8%20Lab%20-%20Configuring%20Multiarea%20OSPFv2.pdf)

# Laborgyakorlat - Többterületű OSPFv3 beállítása

**Ebben a gyakorlatban a következő feladatokat végezzük el:**

* 1. rész: A hálózat felépítése és az eszközök alapvető konfigurálása.
* 2. rész: Többterületű OSPFv3 forgalomirányítás beállítása
* 3. rész: Területek közötti útvonal-összevonás beállítása

[Lab - Configuring Multiarea OSPFv3](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/6.2.3.9%20Lab%20-%20Configuring%20Multiarea%20OSPFv3.pdf)

**Összefoglalás**

Kisebb hálózatokban megfelelő lehet az egyterületű OSPF is, de nagyobb hálózatok esetében a többterületű OSPF használata javasolt. Ahogy az 1. és 2. ábrán látható, a többterületű OSPF használata megoldja a nagyméretű irányítótábla és kapcsolatállapot-adatbázis, valamint az SPF algoritmus gyakori újraszámítása által okozott problémákat.

A fő területet gerinchálózati területnek (area 0) nevezik, és valamennyi területnek ehhez kell kapcsolódnia. A forgalomirányítás a területek között zajlik, viszont a számításigényes forgalomirányítási feladatok, mint például az adatbázisok újraszámítása, az egyes területeken belül maradnak.

Az OSPF forgalomirányítók négy típusba sorolhatók: belső forgalomirányító, gerinchálózati forgalomirányító, határ-forgalomirányító (ABR), autonóm rendszer határ-forgalomirányítója (ASBR). Egy forgalomirányító többféle típusba is besorolható.

A kapcsolatállapot-hirdetmény (LSA) az OSPF protokoll alkotóeleme. Ez a fejezet elsősorban az 1-estől az 5-ös típusig terjedő LSA-kra helyezte a hangsúlyt. Az 1-es típusú LSA-t a forgalomirányító kapcsolati bejegyzéseinek nevezzük. A 2-es típusú LSA-t a hálózati kapcsolatok bejegyzéseinek nevezzük, hirdetésük a DR forgalomirányító feladata. A 3-as típusú LSA-t összevont kapcsolati bejegyzésnek is nevezzük, létrehozásuk és hirdetésük az ABR feladata. 4-es típusú összevont LSA-t csak akkor állít elő az ABR, ha egy ASBR is létezik a területen belül. Az 5-ös típusú LSA azokat az útvonalakat határozza meg, amelyek az OSPF autonóm rendszeren kívül eső hálózatokhoz vezetnek. Az ASBR forgalomirányítótól származó 5-ös típusú LSA-k elárasztják a teljes autonóm rendszert.

Az IPv4-es irányítótábla OSPF útvonalait az alábbi leírókkal lehet megadni: O, O IA, O E1 és O E2. Az SPF-fa felépítéséhez minden egyes forgalomirányító az SPF-algoritmust futtatja le az LSDB tartalmán. Az SPF-fa szolgál a legjobb útvonalak meghatározására.

Egy többterületű OSPF hálózat megvalósításához nincs szükség speciális parancsokra. Egy forgalomirányítóból mindössze két, különböző területre vonatkozó **network** parancs kiadásával ABR válhat.

Példa a többterületű OSPF beállítására:

R1(config)# **router ospf 10**

R1(config-router)# **router-id 1.1.1.1**

R1(config-router)# **network 10.1.1.1 0.0.0.0 area 1**

R1(config-router)# **network 10.1.2.1 0.0.0.0 area 1**

R1(config-router)# **network 192.168.10.1 0.0.0.0 area 0**

Az OSPF nem végez automatikus útvonal-összevonást. OSPF környezetben útvonal-összevonás kizárólag az ABR és ASBR forgalomirányítókon állítható be. A területek közötti útvonal-összevonást manuálisan kell beállítani. Az útvonal-összevonás az ABR forgalomirányítókon történik, és az egyes területeken belüli útvonalakra vonatkozik. Az útvonal-összevonás ABR forgalomirányítókon történő manuális beállításához használjuk az **area** *terület\_azonosító* **range** *címmaszk (address mask)* forgalomirányító-konfigurációs parancsot.

A külső útvonalak összevonása csak az OSPF-be újrahirdetéssel behozott külső útvonalakra vonatkozik. Általában csak az ASBR vonja össze a külső útvonalakat. A külső útvonalak összevonásának beállítása az ASBR forgalomirányítókon kiadott **summary-address** *címmaszk (address mask)* forgalomirányító-konfigurációs paranccsal történik.

Az OSPF beállítások ellenőrzéséhez az alábbi parancsok használhatók:

* **show ip ospf neighbor**
* **show ip ospf**
* **show ip ospf interface**
* **show ip protocols**
* **show ip ospf interface brief**
* **show ip route ospf**
* **show ip ospf database**
* [Fejezet száma 7 EIGRP](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#7)
* [7.0 EIGRP](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#7.0)
* [7.0.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#7.0.1)
* [7.0.1.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#7.0.1.1)

# Bevezetés

A továbbfejlesztett belső átjáró irányító protokoll (Enhanced Interior Gateway Routing Protocol, EIGRP) egy Cisco Systems által kifejlesztett távolságvektor alapú irányító porotokoll. Ahogy a neve is sugallja, az EIGRP egy másik Cisco fejlesztésű irányító protokoll, az IGRP (Interior Gateway Routing Protocol) továbbfejlesztése. Az IGRP egy régebbi osztály és távolságvektor alapú irányító protokoll, amely az IOS 12.3 verziójától kezdve elavultnak számít.

Az EIGRP egy távolság vektor alapú irányító protokoll, amely kapcsolatállapot alapú irányító protokollokra jellemző funkciókat is tartalmaz. Számos különböző topológián és átviteli közegen is alkalmazható. Egy jól tervezett hálózatban az EIGRP összetett topológiák esetében is használható, valamint rendkívül gyors konvergenciát képes biztosítani minimális hálózati forgalom mellett.

Ez a fejezet az EIGRP protokoll bemutatásával foglalkozik, valamint ismerteti a Cisco IOS-t futtató forgalomirányítókon használt alapvető konfigurációs parancsokat. Ezen kívül az irányító protokoll működését és a legjobb útvonalak meghatározásának módját is részletezi.

# Az EIGRP tulajdonságai

Az EIGRP 1992-es megjelenésekor még saját gyártói protokollnak számított és csak Cisco eszközön volt elérhető. A Cisco 2013-ban az EIGRP alap funkcióit nyílt szabványúvá tette az IETF számára egy RFC dokumentum formájában. Ez azt jelenti, hogy más gyártók is használhatják az eszközeiken az EIGRP-t, és így együttműködhetnek a szintén EIGRP-t futtató Cisco és nem Cisco forgalomirányítókkal. Ugyanakkor az EIGRP fejlettebb funkciói, mint például a dinamikus többpontú virtuális magánhálózat (Dynamic Multipoint Virtual Private Network, DMVPN) megvalósításához szükséges EIGRP stub funkció nem lett átadva az IETF-nek. Az RFC dokumentum kezelőjeként továbbra is a Cisco fogja ellátni az EIGRP működésének felügyeletét.

Az EIGRP kapcsolatállapot és távolságvektor alapú irányító protokollok funkcióit is tartalmazza, de a távolságvektor alapú irányító protokollok elvén alapul, amely szerint a hálózat többi részéről a közvetlenül csatlakozó szomszédoktól lehet információkat megtudni.

Az EIGRP egy fejlett távolságvektor alapú irányító protokoll, amelyben olyan funkciók is megtalálhatók, amelyek más távolságvektor alapú protokolloknál, mint például a RIP és az IGRP, nincsenek.

**Szétszóró frissítő algoritmus (Diffusing Update Algorithm)**

Az EIGRP központi eleme a szétszóró frissítő algoritmus (Diffusing Update Algorithm, DUAL), amelynek útvonalszámító algoritmusa biztosítja az irányító protokoll működését. A DUAL hurokmentes és tartalék útvonalakat biztosít az irányítási tartomány minden hálózata felé. A DUAL használatával az EIGRP az összes elérhető tartalék útvonalat eltárolja a célhálózatok felé, így szükség esetén gyorsan képes alternatív útvonalakra átváltani.

**Szomszédsági kapcsolatok kialakítása**

Az EIGRP kapcsolatot alakít ki a közvetlenül csatlakozó, szintén EIGRP-t futtató forgalomirányítókkal. A szomszédsági viszonyok használatával nyomon követhető a szomszédok állapota.

**Megbízható szállítási protokoll (RTP)**

A kizárólag az EIGRP által használt megbízható szállítási protokoll (Reliable Transport Protocol, RTP) az EIGRP csomagok szomszédok közötti továbbítását biztosítja. Az RTP és a szomszédsági kapcsolatok nyomon követése biztosítják a DUAL algoritmus számára a működési feltételeket.

**Részleges és korlátozott útvonalfrissítések**

Az EIGRP útvonalfrissítései a részleges és a korlátozott kifejezésekkel jellemezhetők. A RIP-pel ellentétben az EIGRP nem küld rendszeres időközönként útvonalfrissítéseket, valamint az útvonalbejegyzései nem évülnek el. A részleges kifejezés azt jelenti, hogy az útvonalfrissítés csak az útvonal megváltozásáról tartalmaz információkat. Ilyen információ például egy új vagy meglévő kapcsolat elérhetővé válása. A korlátozott kifejezés pedig azt jeleni, hogy a részleges frissítések csak azokhoz a forgalomirányítókhoz lesznek továbbítva, akiket érint a változás. Ez csökkenti az EIGRP frissítések továbbításához szükséges sávszélességet.

**Egyenlő és különböző költségű útvonalak közötti terheléselosztás**

Az EIGRP támogatja az egyenlő és különböző költségű terheléselosztást, amely lehetővé teszi a rendszergazdák számára az adatfolyam jobb elosztását a hálózatokban.

**MEGJEGYZÉS:**Számos régebbi dokumentációban használják a hibrid irányító protokoll kifejezést az EIGRP-re hivatkozva. Ez a kifejezés azonban félrevezető, ugyanis az EIGRP nem a távolságvektor és a kapcsolatállapot alapú irányító protokollok hibridje, hanem kizárólag távolságvektor alapú protokoll. Éppen ezért a Cisco már nem használja ezt a kifejezést az EIGRP-re vonatkozóan.

# Protokollfüggő modulok

Az EIGRP protokollfüggő modulok (protocol-dependent modul, PDM) alkalmazásával több különböző irányító protokoll számára is képes forgalomirányítást végezni, beleértve az IPv4-et és IPv6-ot is. Az EIGRP PDM-ek segítségével alkalmas volt a mára elavultá vált Novell IPX és az Apple Computer AppleTalk nevű hálózati rétegbeli protokollok forgalomirányítására.

A PDM-ek a hálózati réteg protokoll-specifikus feladatainak elvégzéséért felelősek. Ennek egyik példája az az EIGRP modul, amely az IPv4-be beágyazott EIGRP csomagok küldéséért és fogadásáért felelős. Ez a modul felelős az EIGRP csomagok vizsgálatáért, valamint a kinyert információk DUAL-algoritmushoz továbbításáért is. Az EIGRP a DUAL algoritmusra bízza az irányítási döntéseket,de a végeredmény az IPv4-irányítótáblában lesz eltárolva.

A PDM-ek a hálózati rétegbeli protokollok speciális forgalomirányítási feladatainak elvégzéséért felelősek, mint például:

* Az EIGRP-t futtató forgalomirányítók szomszédsági és topológiai táblázatának karbantartása.
* Protokoll-specifikus csomagok összeállítása és értelmezése a DUAL algoritmus számára.
* A DUAL-algoritmus eredményeinek átvezetése a protokoll-specifikus irányítótáblába.
* A mérték kiszámolása és az érték továbbítása a DUAL-algoritmus számára.
* Szűrések és hozzáférési listák végrehajtása
* Más forgalomirányító protokolloktól érkező és számukra küldött információk továbbhirdetése.
* Más forgalomirányító protokolloktól tanult útvonalak továbbhirdetése.

Ha a forgalomirányító új szomszédot fedez fel, akkor egy bejegyzésben rögzíti a szomszéd interfészét és annak címét a szomszédsági táblában. Minden egyes protokoll függő modulhoz (PDM), például az IPv4-hez is külön szomszédsági tábla tartozik. Az EIGRP egy topológiatáblát is fenntart. A topológiatábla az összes olyan célhálózatot tartalmazza, amelyeket a szomszédos forgalomirányítók hirdetései tartalmaztak. Minden egyes PDM-hez külön topológiatábla is tartozik.

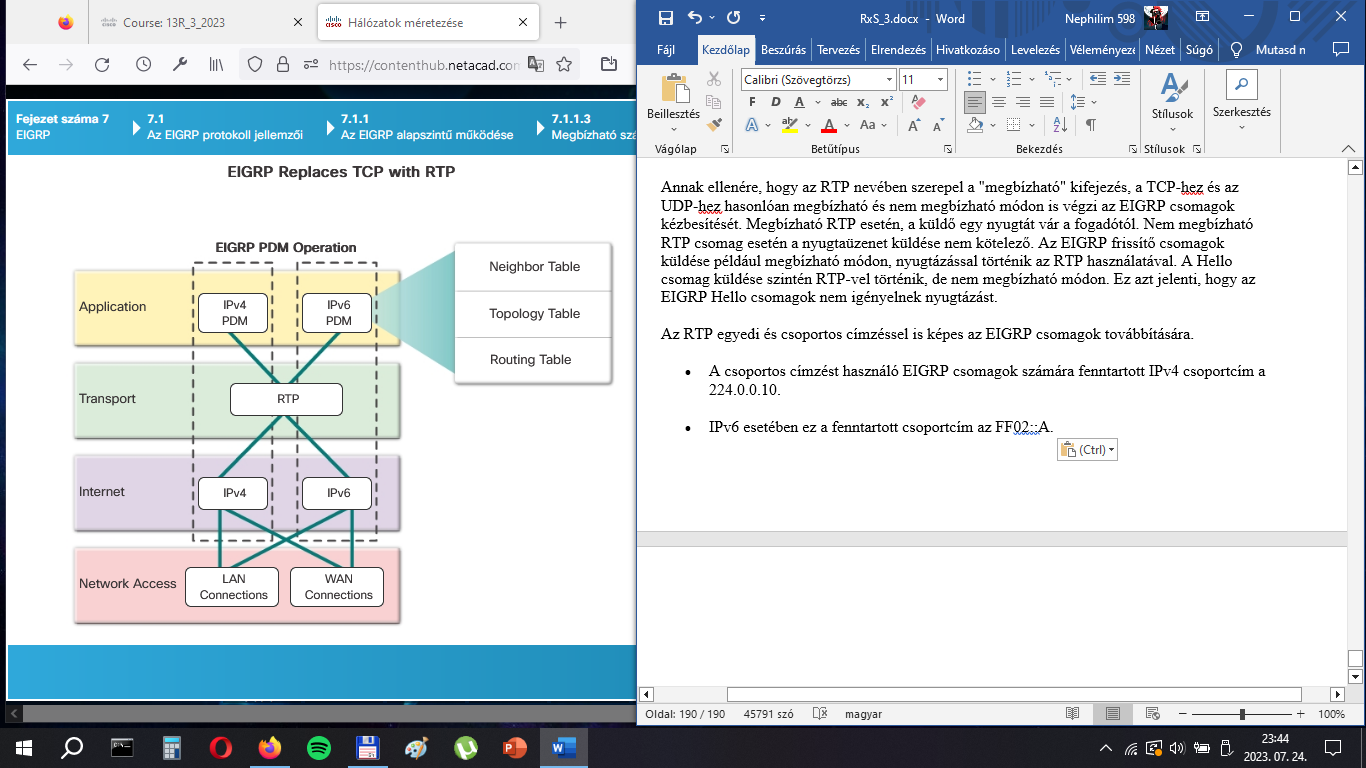
# Megbízható szállítási protokoll (RTP)

Az EIGRP a megbízható szállítási protokollt (RTP) használja a csomagjai továbbításához és fogadásához. Az EIGRP-t a hálózati rétegtől független forgalomirányító protokollnak tervezték, ezen kialakítás miatt nem képes az UDP vagy a TCP szolgáltatásait igénybe venni. Ez viszont lehetővé teszi, hogy a TCP/IP protokollkészleten kívüli protokollok, például az IPX és az AppleTalk is használják. Az ábra az RTP működését mutatja.

Annak ellenére, hogy az RTP nevében szerepel a "megbízható" kifejezés, a TCP-hez és az UDP-hez hasonlóan megbízható és nem megbízható módon is végzi az EIGRP csomagok kézbesítését. Megbízható RTP esetén, a küldő egy nyugtát vár a fogadótól. Nem megbízható RTP csomag esetén a nyugtaüzenet küldése nem kötelező. Az EIGRP frissítő csomagok küldése például megbízható módon, nyugtázással történik az RTP használatával. A Hello csomag küldése szintén RTP-vel történik, de nem megbízható módon. Ez azt jelenti, hogy az EIGRP Hello csomagok nem igényelnek nyugtázást.

Az RTP egyedi és csoportos címzéssel is képes az EIGRP csomagok továbbítására.

* A csoportos címzést használó EIGRP csomagok számára fenntartott IPv4 csoportcím a 224.0.0.10.
* IPv6 esetében ez a fenntartott csoportcím az FF02::A.



# Hitelesítés

Más forgalomirányító protokollokhoz hasonlóan az EIGRP protokollon is beállítható hitelesítés. A RIPv2, EIGRP, OSPF, IS-IS és BGP protokollok mindegyike konfigurálható az irányítási információik hitelesítésére.

A továbbított irányítási információk hitelesítése jó gyakorlatnak számít. Alkalmazásával biztosítható, hogy a forgalomirányítók csak olyan forgalomirányítóktól fogadjanak el irányítási információkat, amelyeket egyező jelszóval vagy hitelesítési információkkal konfiguráltak.

**MEGJEGYZÉS:**A hitelesítés nem titkosítja az EIGRP útvonalfrissítéseket.

# EIGRP csomagtípusok

Az EIGRP öt különböző csomagtípust használ, néhányat párban egymással. A csomagok küldése RTP használatával megbízható vagy nem megbízható kézbesítéssel történik, egyedi, csoportos vagy néha mindkétféle címzést használva. Az EIGRP csomagtípusokat EIGRP csomagformátumnak vagy EIGRP üzenetnek is nevezik.

Ahogy az 1. ábrán is látható, az öt EIGRP csomagtípus a következő:

**Hello csomagok** - A szomszédsági kapcsolatok létrehozását és karbantartását végzik.

* Továbbításuk nem megbízható kézbesítéssel történik.
* Csoportos címzésűek (a legtöbb hálózattípus esetén).

**Frissítő csomagok** - Irányítási információk továbbítását végzik az EIGRP szomszédok között.

* Továbbításuk megbízható kézbesítéssel történik.
* Egyedi vagy csoportos címzésűek.

**Nyugtacsomagok** - A megbízható kézbesítéssel továbbított EIGRP üzenetek nyugtázását végzik.

* Továbbításuk nem megbízható kézbesítéssel történik.
* Egyedi címzésűek.

**Lekérdező csomagok** - Útvonal információkat kérdeznek le a szomszédoktól.

* Továbbításuk megbízható kézbesítéssel történik.
* Egyedi vagy csoportos címzésűek.

**Válasz csomagok** - Az EIGRP lekérdezésekre küldött válasz üzenetek.

* Továbbításuk megbízható kézbesítéssel történik.
* Egyedi címzésűek.

A 2. ábrán az látható, hogy az EIGRP csomagok általában IPv4 vagy IPv6 csomagokba vannak beágyazva. Az EIGRP IPv4-es csomagjai az IPv4-et használják hálózati rétegbeli protokollként. A Protokoll (Protocol) mező 88-as értéke azt jelzi, hogy a csomag adat része egy EIGRP IPv4-es üzenetet tartalmaz. Az EIGRP IPv6-os üzenetei IPv6 csomagokba vannak beágyazva, ezt a csomag Következő fejrész (Next Header) mezőjének 88-as értéke jelzi. Az IPv4 Protokoll mezőjéhez hasonlóan az IPv6 csomagban az IPv6 Következő fejrész mező jelzi a továbbított adat típusát.

**EIGRP Hello csomagok**

Az EIGRP kisméretű Hello csomagokat használ a többi EIGRP protokollt használó, közvetlenül csatlakozó forgalomirányító felderítéséhez. A Hello csomagokat az EIGRP szomszédsági viszonyok vagy más néven kapcsolatok kialakítására használják a forgalomirányítók.

Az EIGRP Hello csomagok IPv4 vagy IPv6 csoportos címzéssel továbbítódnak, és nem megbízható RTP kézbesítést használnak. Ez azt jelenti, hogy a vevő nem küld válasz nyugtacsomagot.

* Az EIGRP számára fenntartott IPv4 csoportos cím a 224.0.0.10.
* Az EIGRP számára fenntartott IPv6 csoportos cím az FF02::A.

Az EIGRP forgalomirányítók Hello csomagok használatával fedezik fel a szomszédos forgalomirányítókat és alakítanak ki szomszédsági viszonyt velük. A legtöbb hálózatban az EIGRP Hello csomagok csoportos címzéssel öt másodpercenként kerülnek elküldésre. Azonban a többpontos, nem szórásos, többes hozzáférésű (NBMA) hálózatok, mint például az X.25, Frame Relay és a T1 (1.544 Mb/s) vagy lassabb kapcsolattal rendelkező ATM esetén a Hello csomagok egyedi címzéssel 60 másodpercenként kerülnek elküldésre.

Az EIGRP a már kiépített szomszédsági viszonyok fenntartására is használja a Hello csomagokat. Az EIGRP forgalomirányító feltételezi, hogy amíg Hello csomagokat kap a szomszédjától, addig a szomszéd és az útvonalai elérhetőek.

Az EIGRP egy megtartási időzítőt (Hold timer) alkalmaz annak a legnagyobb időértéknek a meghatározására, amennyit a forgalomirányítónak várnia kell a következő Hello csomagra, mielőtt a szomszédot elérhetetlennek minősítené. Alapértelmezés szerint a megtartási időzítő értéke a Hello időzítő értékének háromszorosa, vagy 15 másodperc a legtöbb hálózaton és 180 másodperc a kis sebességű NBMA hálózatokon. Ha a megtartási időzítő lejár, az EIGRP elérhetetlennek nyilvánítja az útvonalat és a DUAL algoritmus kérések kiküldésével új útvonalat keres.

# EIGRP frissítő és nyugtacsomagok

**EIGRP frissítő csomagok**

Az EIGRP protokoll az útvonal információk hirdetésére frissítő csomagokat használ. A frissítő csomagok csak szükség esetén kerülnek elküldésre. Az EIGRP frissítések csak a szükséges irányítási információkat tartalmazzák, és csak azok a forgalomirányítók kapják meg, akiknek ez szükséges.

A RIP-pel ellentétben az EIGRP (szintén távolságvektor alapú irányító protokoll) nem küld rendszeres időközönként útvonalfrissítéseket, valamint az útvonalbejegyzései sem évülnek el. Ehelyett részleges frissítéseket küld ki akkor, ha a cél állapota megváltozik. Ez akkor áll fenn, ha egy új hálózat elérhetővé vagy egy meglévő hálózat elérhetetlenné válik, vagy változás áll be egy meglévő hálózathoz tartozó irányítási mértékben.

EIGRP esetén a *részleges* és *korlátozott* kifejezéseket használjuk, amikor az EIGRP frissítéseire hivatkozunk. A részleges kifejezés azt jelenti, hogy az útvonalfrissítés csak az útvonal megváltozásáról tartalmaz információkat. A korlátozott kifejezés pedig azt jeleni, hogy a részleges frissítések csak azokhoz a forgalomirányítókhoz lesznek továbbítva, akiket érint a változás.

Azzal, hogy csak a szükséges irányítási információk küldése történik meg és csak az érintett forgalomirányítóknak, az EIGRP a minimálisra csökkenti a frissítések elküldéséhez szükséges sávszélességet.

Az EIGRP frissítő csomagok küldése megbízható módon történik, ami azt jelenti, hogy a küldést végző forgalomirányító nyugtát vár vissza. A frissítő csomagokat a forgalomirányítók csoportos címzéssel küldik, ha több forgalomirányító érintett, vagy egyedi címzéssel ha csak egy. Mivel az ábrán látható kapcsolatok pont-pont összeköttetések, ezért a frissítések egyedi címzéssel továbbítódnak.

**EIGRP nyugtacsomagok**

Megbízható kézbesítés esetén az EIGRP nyugtacsomagokat (ACK) küld. Az EIGRP nyugtacsomag egy adatok nélkül elküldött Hello csomag. Az RTP megbízható kézbesítést használ az EIGRP frissítő, lekérdező, és válasz csomagjai esetén, a nyugtacsomagok viszont minden esetben nem megbízható kézbesítéssel és egyedi címzéssel kerülnek elküldésre. A nem megbízható kézbesítésnek is van értelme, különben a nyugták végtelen hurokba kerülnének.

Az ábrán látható, hogy az R2 elvesztette a kapcsolatot a Gigabit Ethernet interfészéhez csatlakozó helyi hálózattal. R2 azonnal frissítést küld ki az R1 és R3 forgalomirányítóknak, melyben megjelöli az elérhetetlen útvonalat. Az R1 és az R3 nyugtával válaszolnak, értesítve ezzel az R2-őt, hogy megkapták a frissítést.

**MEGJEGYZÉS:**Számos dokumentáció úgy hivatkozik a Hello és a nyugtacsomagokra, mint az egyedüli EIGRP csomagtípusokra.

# EIGRP lekérdező és válasz csomagok

**EIGRP lekérdező csomagok**

A DUAL algoritmus lekérdező és válasz csomagokat használ a hálózatok kereséséhez és egyéb feladatok végrehajtásához. A lekérdezések és válaszok megbízható kézbesítést használnak. A lekérdezések használhatnak csoportos és egyedi címzést is, míg a válaszok minden esetben egyedi címzéssel továbbítódnak.

Az ábrán látható, hogy az R2 elvesztette a kapcsolatot a helyi hálózattal, és lekérdezéseket küld ki az összes EIGRP szomszédjának a helyi hálózathoz vezető lehetséges új útvonalakat keresve. Mivel a lekérdezések megbízható kézbesítést használnak, a vevő forgalomirányítónak EIGRP nyugtát kell visszaküldenie. A nyugta üzenetek tájékoztatják a küldőt, hogy a lekérdező üzeneteket megkapták. Az egyszerűség kedvéért a nyugta üzenetek nincsenek feltüntetve az ábrán.

**EIGRP válasz csomagok**

A szomszédok mindegyikének válaszolnia kell, függetlenül attól, hogy rendelkeznek-e útvonallal az elérhetetlen hálózat felé. Mivel a válaszok szintén megbízható kézbesítést használnak, a forgalomirányítóknak, így az R2-nek is, nyugtacsomagot kell küldeniük.

Miért is kell az R2-nek lekérdezést kiküldenie egy olyan hálózatról, amelyről tudja, hogy nem elérhető? Valójában az R2 csak azon interfésze elérhetetlen, amelyhez a hálózat csatlakozik. Egy másik forgalomirányító is csatlakozhat ugyanehhez a helyi hálózathoz, és így lehet egy másik útvonal is a hálózat felé. Az R2 ezért egy ilyen forgalomirányító keresésébe kezd, mielőtt teljesen eltávolítaná a hálózatot a topológiatáblájából.

# EIGRP üzenetek beágyazása

Az EIGRP üzenetek adat része egy csomagba van beágyazva. Ez az adat mező az úgynevezett TLV (Type/Length/Value; típus/hossz/érték). A TLV típusok fejezethez tartozó részei közé az EIGRP paraméterek, a belső és külső IP-útvonalak tartoznak.

Az EIGRP csomag fejrész minden egyes EIGRP csomagban megtalálható, függetlenül annak típusától. Az EIGRP csomag fejrész és a TLV egy IPv4-csomagba vannak beágyazva. Az IPv4 csomagfejrészben a protokoll mező értéke 88, amely az EIGRP-t jelzi, az IPv4-célcím pedig a 224.0.0.10 csoportcímre van állítva. Ha az EIGRP csomag egy Ethernet keretbe van beágyazva, a cél MAC-cím szintén csoportos cím lesz, ez a 01-00-5E-00-00-0A.

Az 1-4. ábrák az adatkapcsolati Ethernet keretet mutatják. Az EIGRP IPv4-es változata IPv4-csomagba van beágyazva. Az IPv6-os változat hasonló típusú beágyazást alkalmazna IPv6-fejrésszel. Ebben az esetben az IPv6 célcím a csoportos címzésű FF02::A, a következő fejrész mező értéke pedig 88 lenne.

**Az EIGRP csomag fejrész és a TLV mezők**

Minden EIGRP üzenet az 1. ábrán látható fejrészt tartalmazza. A fontosabb mezők közé az üzenet típusát jelző (Opcode) és az autonóm rendszer azonosító (Autonomous System Number) mezők tartoznak. Az Opcode mező a következő EIGRP csomag típusokat határozza meg:

* Frissítő
* Lekérdező
* Válasz
* Hello

Az autonóm rendszer szám határozza meg az EIGRP forgalomirányítási folyamatot. A RIP-pel ellentétben az EIGRP több példánya is futhat egy hálózaton, és az egyes folyamatok nyomon követésére az autonóm rendszer azonosítószám használható.

A 2. ábrán az EIGRP paramétereket tartalmazó TLV mező látható, amely tartalmazza az összetett EIGRP mérték meghatározásához használt súlyozási értékeket. Alapértelmezés szerint csak a sávszélesség és a késleltetés van súlyozva, mégpedig egyenlő mértékben.Így mind a sávszélességhez tartozó K1 mező, mind a késleltetésre vonatkozó K3 mező értéke 1-re van állítva. A többi K mező értéke 0.

A megtartási idő (Hold time) alatt azt az időt értjük, amennyit az üzenetet fogadó EIGRP szomszédnak várnia kell, mielőtt a hirdetést küldő forgalomirányítót elérhetetlennek minősíti.

A 3. ábrán a belső IP-útvonalak TLV-je látható. A belső IP-üzenetet az EIGRP útvonalak autonóm rendszeren belüli hirdetéséhez használják. A fontosabb mezők közé a mérték (késleltetés és sávszélesség), az alhálózati maszk (előtag hossz) és a cél mezők tartoznak.

A késleltetés 10 mikroszekundumban megadott értéke a forrás és cél közötti összesített késleltetésből adódik. A sávszélesség az útvonal bármely interfészén beállított legalacsonyabb sávszélességet jelenti.

Az alhálózati maszk mező az előtag hosszát vagy az alhálózati maszk hálózati bitjeinek számát adja meg. Például a 255.255.255.0 alhálózati maszkhoz tartozó előtag hossz 24, mivel a hálózati bitek száma 24.

A cél (Destination) mező a célhálózat címét tartalmazza. Habár az ábra csak 24 bitet mutat, a mező értéke a 32 bites hálózati cím hálózati részének hosszától függően változik. Például a 10.1.0.0/16 cím hálózati része a 10.1, emiatt a cél mező az első 16 bitet tartalmazza. Mivel a mező minimális hossza 24 bit, a többi része nullával van kitöltve. Ha a hálózati cím 24 bitnél hosszabb (például 192.168.1.32/27), a cél mező újabb 32 bittel ki van egészítve (összesen 56 bitre), a nem használt bitek pedig nullával vannak kitöltve.

A 4. ábrán a külső IP-útvonalak TLV-je látható. A külső IP-üzenetet külső útvonalak EIGRP irányító folyamatba importálásakor használják. Ebben a fejezetben egy alapértelmezett statikus útvonalat fogunk az EIGRP-be importálni vagy más néven továbbhirdetni. Figyeljük meg, hogy a külső IP-útvonal TLV alsó fele azokat a mezőket tartalmazza, amelyeket a belső IP-útvonal TLV-je is használ.

**MEGJEGYZÉS:**A maximális átviteli egység (MTU) nem az EIGRP által használt mérték. Az útvonalfrissítések tartalmazzák az MTU-t, de az irányítási mérték meghatározásához nem használják azt.

# EIGRP hálózati topológia

Az 1. ábra azt a hálózati topológiát mutatja, amelyet az EIGRP IPv4-es konfigurációjakor használunk a kurzus során. A soros interfészek típusa és sávszélessége nem szükségszerűen felel meg a napjainkban leginkább elterjedt kapcsolattípusoknak. A topológiában használt soros vonalak sávszélességei a forgalomirányító protokollok mértékszámítása és a legjobb út kiválasztási folyamat bemutatása céljából kerültek kiválasztásra.

A topológia forgalomirányítói olyan alapkonfigurációval rendelkeznek, amely az interfész címeket is tartalmazza. Pillanatnyilag egyik forgalomirányítón sincs statikus vagy dinamikus forgalomirányítás konfigurálva.

A 2., 3. és 4. ábrán a topológia EIGRP forgalomirányítóinak interfész beállításai láthatók. Csak az R1, R2 és az R3 forgalomirányítók részei az EIGRP forgalomirányítási tartománynak. Az ISP forgalomirányító az adott irányítási terület átjárója az internet felé.

# Autonóm rendszerazonosítók

Az EIGRP irányítási folyamatot a **router eigrp** *autonóm\_rendszer\_azonosító* paranccsal lehet engedélyezni. Az EIGRP konfigurálásához használt autonóm rendszer azonosító nem egyezik meg az IANA által globálisan kiosztott autonóm rendszer azonosítóval, azokat ugyanis külső forgalomirányító protokollok használják.

Mi tehát a különbség az IANA globálisan hozzárendelt és az EIGRP által használt autonóm rendszer azonosítók között?

Az IANA által kezelt autonóm rendszer egyazon felügyelet alatt álló hálózatok együttesét jelenti, amely közös forgalomirányítási stratégiát alkalmaz az internet felé. Az ábrán látható, hogy az A, B, C és D vállalatok mindegyike az ISP1 felügyelete alá tartoznak. Az ISP1 egységes forgalomirányítási szabályokat biztosít ezen vállalatok számára, amikor útvonalakat hirdet az ISP2 felé.

Az autonóm rendszerek létrehozásának, kiválasztásának és regisztrálásának irányelveit az RFC 1930 dokumentum írja le. A globális autonóm rendszer azonosítók kiosztását az IANA végzi, ugyanaz a szervezet, amely az IP-címek kiosztásáért is felelős. A helyi regionális internet regisztrátor (RIR) felelős azért, hogy a hozzá rendelt tartományból autonóm rendszer azonosítót osszon ki a szervezetekhez. 2007 előtt az autonóm rendszer azonosítók 0 és 65535 értékek közötti, 16 bites számok voltak. Manapság már 32 bites azonosítókat osztanak ki, több mint 4 milliárd autonóm rendszer elérhetőségét biztosítva ezzel.

Általában internetszolgáltatók (ISP), gerinchálózati szolgáltatók és egymáshoz kapcsolt nagyméretű szervezetek igényelnek autonóm rendszer azonosítót. Ezek a szolgáltatók és nagyméretű szervezetek egy külső határátjáró protokollt, a BGP-t (Border Gateway Protocol) használják az irányítási információk továbbítására. A BGP az egyetlen olyan irányító protokoll, amely egy valódi autonóm rendszer azonosítót használ a konfigurációjában.

Az IP-hálózatokkal rendelkező vállalatok és szervezetek túlnyomó többségének nincs szüksége autonóm rendszer azonosítóra, mivel egy nagyobb szervezet, például egy internetszolgáltató (ISP) felügyelete alatt állnak. Ezek a vállalatok belső határátjáró protokollokat, például RIP, EIGRP, OSPF és IS-IS protokollt használnak a saját hálózataikon belüli forgalomirányításra. Így ők az ISP autonóm rendszerén belül található számos független és önálló hálózat egyike. Az internetszolgáltató felelős a csomagok saját autonóm rendszeren belüli és más autonóm rendszerek közötti irányításáért.

Az EIGRP konfigurációjában használt autonóm rendszer azonosítónak csak az EIGRP irányítási területen belül van jelentősége. Folyamatazonosítóként (process ID) működik, amely az EIGRP többszörösen futó példányainak nyomon követésében segíti a forgalomirányítókat. Azért van rá szükség, mert előfordulhat, hogy az EIGRP több példánya is fut a hálózatban. Beállítható, hogy az EIGRP minden példánya eltérő hálózatok között támogassa és cserélje az irányítási információkat.

# A router EIGRP parancs

A Cisco IOS tartalmaz olyan folyamatokat, amelyekkel számos különböző dinamikus irányító protokoll engedélyezhető és konfigurálható. A globális konfigurációs módbeli **router** parancs használható bármely dinamikus irányító protokoll konfigurációjának megkezdéséhez. Az 1. ábrán látható topológiát használjuk a parancs bemutatásához.

Ahogy a 2. ábrán látható a kérdőjel (**?**), amely a **router** globális konfigurációs parancs után látható, kilistázza az összes elérhető irányító protokollt, amelyet a forgalomirányítón futó IOS verzió támogat.

A forgalomirányító konfigurációs módba történő belépéshez és az EIGRP folyamat konfigurációjának megkezdéséhez a következő globális konfigurációs módbeli parancs használható:

Router(config)# **router eigrp** *autonóm\_rendszer\_azonosító*

Az *autonóm\_rendszer\_azonosító* paraméterhez tetszőleges 1 és 65535 közötti 16 bites értéket lehet hozzárendelni. Az EIGRP irányítási tartományon belül minden forgalomirányítón ugyanazt az autonóm rendszer azonosítót kell használni.

A 3. ábra az R1, R2 és az R3 forgalomirányítók EIGRP konfigurációját mutatja. Figyeljük meg, hogy a globális konfigurációs mód parancssora forgalomirányító konfigurációs módbeli parancssorra változik.

Ebben a példában az **1** azonosítja a forgalomirányítón futó EIGRP folyamatot. Ahhoz, hogy EIGRP esetén a szomszédsági viszonyok létrejöjjenek, azonos autonóm rendszer azonosítót kell beállítani az ugyanazon irányítási tartományhoz tartozó forgalomirányítókon. A 3. ábrán látható, hogy mindhárom forgalomirányítón ugyanaz az EIGRP folyamat van engedélyezve és a rajtuk használt autonóm rendszer azonosító **1**.

**MEGJEGYZÉS:**Az EIGRP és az OSPF protokollok mindegyike képes az irányító protokoll több példányának kezelésére is, habár erre a többpéldányos megvalósításra általában nincs szükség vagy használata nem ajánlott.

A **router eigrp** *autonóm\_rendszer\_azonosító* parancs önmagában még nem indítja el az EIGRP irányítási folyamatot, és a forgalomirányító nem kezdi el a frissítések kiküldését. Ez a parancs csak hozzáférést biztosít az EIGRP beállításokhoz.

Az EIGRP irányítási folyamat adott eszközről történő teljes eltávolításához használjuk a **no router eigrp** *autonóm\_rendszer\_azonosító* globális konfigurációs módbeli parancsot, amely leállítja az EIGRP folyamatokat és minden létező EIGRP konfigurációt eltávolít a forgalomirányítóról.

# Az EIGRP forgalomirányító azonosító (Router ID)

**A forgalomirányító azonosító meghatározása**

Az EIGRP forgalomirányító azonosítót (router ID) az EIGRP irányítási tartományon belüli forgalomirányítók egyedi azonosítására használják. A forgalomirányító azonosítót az EIGRP és az OSPF irányító protokolloknál is használják, bár ennek szerepe az OSPF esetén lényegesen jelentősebb.

Az EIGRP IPv4-es megvalósításai esetén a forgalomirányító azonosító használata nem olyan egyértelmű. Az EIGRP IPv4 esetén egy 32 bites forgalomirányító azonosítót használ a továbbhirdetett külső útvonalak forrását jelentő forgalomirányító azonosításához. A forgalomirányító azonosító szükségessége még inkább nyilvánvalóvá válik az EIGRP IPv6-os változatának tárgyalásakor. A forgalomirányító azonosítóra az útvonalak továbbhirdetésekor van szükség. Az EIGRP útvonalak továbbhirdetése túlmutat jelen tananyag keretein. A tananyag szempontjából csupán arra van szükség, hogy megértsük mi a forgalomirányító azonosító és honnan származtatható.

A Cisco forgalomirányítók sorrendben a következő három kritérium alapján határozzák meg a forgalomirányító azonosítót:

1. Az **eigrp router-id** forgalomirányító konfigurációs módbeli parancs használatával beállított IPv4-cím.

2. Ha az azonosító nincs beállítva, akkor a forgalomirányító bármelyik loopback interfészén beállított legnagyobb IPv4-címet választja.

3. Ha nincs loopback interfész konfigurálva, akkor a forgalomirányító bármelyik fizikai interfészére konfigurált legnagyobb IPv4-címet választja.

Ha a rendszergazda nem állít be közvetlenül forgalomirányító azonosítót a **eigrp router-id** paranccsal, akkor az EIGRP előállít egy saját azonosítót a loopback vagy a fizikai interfészek IPv4-címének segítségével. A loopback interfész egy olyan virtuális interfész, amely automatikusan felkapcsolt állapotba kerül a konfigurálásakor. Az interfészt nem kell engedélyezni az EIGRP számára, vagyis nem kell megadni egyetlen network utasításban sem. Az interfésznek viszont felkapcsolt (up/up) állapotban kell lennie.

Az ábrán a fentebb leírt kritériumok alkalmazásával meghatározott alapértelmezett EIGRP forgalomirányító azonosítók láthatók, amelyek a forgalomirányítók legnagyobb IPv4-címeiből lettek meghatározva.

**MEGJEGYZÉS:**Az **eigrp router-id** parancs az EIGRP forgalomirányító azonosítójának beállítására használható. Az IOS egyes verziói elfogadják a **router-id**parancsot, az **eigrp**előtag használata nélkül is. A futó konfigurációban viszont megjelenik az **eigrp router-id** sor függetlenül attól, hogy melyik utasítást használtuk.

# Az EIGRP forgalomirányító azonosító konfigurálása

**eigrp router-id parancs**

Az **eigrp router-id** parancs az EIGRP forgalomirányító azonosító beállítására használható, és elsőbbséget élvez bármely loopback vagy fizikai interfész IPv4-címével szemben. A parancs szintaxisa:

Router(config)# **router eigrp** *autonóm\_rendszer\_azonosító*

Router(config-router)# **eigrp router-id** *IPv4-cím*

**MEGJEGYZÉS:**A forgalomirányító azonosítót jelölő IPv4-cím tulajdonképpen bármilyen 32 bites pontozott decimális szám lehet.

Forgalomirányító azonosítóként bármely IPv4-cím megadható a 0.0.0.0 és a 255.255.255.255 kivételével. Az azonosítónak egy egyedi 32 bites számnak kell lennie az EIGRP irányítási tartományon belül, különben irányítási ellentmondások fordulhatnak elő.

Az 1. ábrán az R1 és az R2 forgalomirányítókon konfigurált EIGRP forgalomirányító azonosítók láthatók, amelyek a **router eigrp** *autonóm\_rendszer\_azonosító* paranccsal lettek beállítva.

**Loopback cím használata forgalomirányító azonosítóként**

Az EIGRP forgalomirányító azonosító megadásának másik módja egy loopback interfész IPv4-címének használata. A loopback interfész használatának előnye egy fizikai interfész IPv4-címével szemben az, hogy a fizikai interfészekkel ellentétben ez nem képes meghibásodni. Nincsenek ugyanis valódi kábelek vagy szomszédos eszközök, amelyektől a loopback interfész felkapcsolt állapota függhetne. Emiatt a loopback cím forgalomirányító azonosítóként történő használata megbízhatóbb azonosítót eredményezhet egy fizikai interfész címhez képest.

Ha egy forgalomirányítón nincs kiadva az **eigrp router-id** parancs, de loopback interfészek vannak konfigurálva, akkor az EIGRP a loopback interfészeken lévő legmagasabb IPv4-címet választja. Egy loopback interfész engedélyezéséhez és konfigurálásához a következő parancsok használhatók:

Router(config)# **interface loopback** *szám*

Router(config-if)# **ip address** *IPv4-cím* *alhálózati-maszk*

**MEGJEGYZÉS:**Az EIGRP forgalomirányító azonosító nem változik meg mindaddig, amíg az EIGRP folyamatot nem távolítjuk el a **no router eigrp** paranccsal, vagy a forgalomirányító azonosítót kézzel be nem állítjuk az **eigrp router-id** parancs segítségével.

**Az EIGRP folyamat ellenőrzése**

A 2. ábrán a **show ip protocols** parancs kimenete látható az R1 forgalomirányítón, amely tartalmazza a forgalomirányító azonosítót is. A **show ip protocols** parancs az aktív forgalomirányító protokoll (például EIGRP és OSPF) folyamatainak paramétereit és aktuális állapotát jeleníti meg. A **show ip protocols** parancs az adott forgalomirányító protokolltól függően eltérő kimenetet adhat.

A 3. ábrán található parancsszimulátorban állítsuk be és ellenőrizzük a forgalomirányító azonosítót az R3-on.

**A network parancs**

Az EIGRP irányító protokollt EIGRP forgalomirányító konfigurációs módban konfigurálhatjuk. Az 1. ábrán látható, hogy az R1, R2 és az R3 mindegyike tartalmaz olyan hálózatot, amelyet az EIGRP irányítási tartománynak tartalmaznia kell. Az EIGRP forgalomirányítás interfészen történő engedélyezéséhez használjuk a **network** forgalomirányító konfigurációs módbeli parancsot, majd adjuk meg az összes közvetlenül csatlakozó hálózat osztály alapú hálózati címét.

A **network** parancsnak minden belső irányító protokoll (IGP) esetén ugyanaz a szerepe. A **network** parancs EIGRP esetén:

* A forgalomirányító minden olyan interfésze számára engedélyezi az EIGRP frissítések küldését és fogadását, amelynek címe beleesik a forgalomirányító konfigurációs módbeli **network** paranccsal megadott hálózatba.
* Az EIGRP útvonalfrissítései tartalmazzák a interfészek hálózatát.

Router(config-router)# **network** *IPv4-hálózati-cím*

Az *IPv4-hálózati-cím* paraméter az interfész osztály alapú IPv4 hálózati címét jelenti. A 2. ábrán az R1 forgalomirányítón konfigurált network parancsok láthatók. Az ábrán egyetlen osztály alapú **network** utasítás, a **network 172.16.0.0**lett kiadva az R1-en, amely mindkét interfész alhálózatát, a 172.16.1.0/24 és a 172.16.3.0/30 alhálózatokat is tartalmazza. Figyeljük meg, hogy csak az osztály alapú cím lett használva.

A 3. ábra azt a **network** parancsot mutatja, amely az R2 interfészein engedélyezi az EIGRP-t a 172.16.1.0/24 és a 172.16.2.0/24 alhálózatok számára. Miután az R2 S0/0/0 interfészén beállításra került az EIGRP, a DUAL algoritmus egy értesítő üzenetet jelenít meg a konzolképernyőn, amely szerint az interfészen szomszédsági viszony épült ki egy másik EIGRP forgalomirányítóval. Ez az új viszony automatikusan jön létre, mivel az R1 és az R2 ugyanazt az **eigrp 1** autonóm rendszer azonosítót használja, és mindkét forgalomirányító frissítések küldésébe kezd a 172.16.0.0 hálózathoz tartozó interfészein.

Alapértelmezés szerint az **eigrp log-neighbor-changes** forgalomirányító konfigurációs módbeli parancs engedélyezett. Ez a parancs a következőkre használható:

* Az EIGRP szomszédsági viszonyokban bekövetkezett bármely változás megjelenítése.
* A szomszédsági kapcsolatok ellenőrzésének támogatása az EIGRP konfigurálása során.
* A hálózati rendszergazda segítése bármely EIGRP szomszédsági viszony megszüntetése esetén.

# A network parancs és a helyettesítő maszk

Alapértelmezés szerint a **network** utasítás és egy IPv4 hálózati cím, mint például a 172.16.0.0, a forgalomirányító összes olyan interfészén engedélyezi az EIGRP-t, amely ebbe az osztály alapú hálózatba tartozik. Előfordulhat azonban, hogy a rendszergazda nem akarja egy hálózat összes interfészén engedélyezni az EIGRP-t. Az 1. ábra alapján tegyük fel például, hogy a rendszergazda engedélyezni szeretné az EIGRP-t az R2 forgalomirányítón, de csak a 192.168.10.8 255.255.255.252 alhálózatot az S0/0/1 interfészen.

Ha az EIGRP-t csak bizonyos alhálózatok hirdetésére szeretnénk beállítani, akkor használjuk a *helyettesítő-maszk* opciót a network parancsban:

Router(config-router)# **network** *hálózati\_cím* [*helyettesítő-maszk*]

Tekintsünk úgy a helyettesítő maszkra, mint az alhálózati maszk inverzére. A 255.255.255.252 alhálózati maszk inverze a 0.0.0.3. Az alhálózati maszk inverzének kiszámolásához vonjuk ki az alhálózati maszkot a 255.255.255.255-ből a következő módon:

      255.255.255.255

 - 255.255.255.252

   ---------------

     0.  0.  0.  3   Helyettesítő maszk

A 2. ábrán az R2 EIGRP konfigurációjának folytatása látható. A **network 192.168.10.8 0.0.0.3** parancs speciálisan a 192.168.10.8 255.255.255.252 alhálózathoz tartozó S0/0/1 interfészen engedélyezi az EIGRP forgalomirányítást.

Néhány IOS verzió az alhálózati maszk használatát is engedélyezi a helyettesítő maszk helyett. A 3. ábrán az R2 ugyanazon S0/0/1 interfészének konfigurációja látható alhálózati maszk használatával a **network** parancs után. Alhálózati maszk használatakor az IOS átalakítja a konfigurációban a parancsot *helyettesítő-maszk* formátumúvá. Ennek ellenőrzése látható a **show running-config** parancs kimenetében a 3. ábrán.

A 4. ábrán található parancsszimulátorban adjuk meg az EIGRP **network** utasításait az R3 forgalomirányítón.

**Passzív interfész**

Miután egy új interfész elérhetővé válik az EIGRP hálózaton, a protokoll megpróbál szomszédsági viszonyt kiépíteni a szomszédos forgalomirányítók valamelyikével, hogy EIGRP frissítéseket tudjon küldeni és fogadni.

Időnként szükséges, sőt előnyös lehet az, ha egy interfészen nem engedélyezett a szomszédsági viszony kiépítése, de az EIGRP útvonalfrissítések tartalmazzák ezt a közvetlenül csatlakozó hálózatot. A **passive-interface** parancs használható a szomszédsági viszonyok kialakításának megakadályozására. Két fő oka van a **passive-interface** parancs használatának:

* A szükségtelen útvonalfrissítő forgalom visszaszorítása miatt, például egy LAN interfész esetén, amelyhez nem csatlakoznak további forgalomirányítók.
* A biztonság fokozása érdekében, például annak megakadályozására, hogy ismeretlen, nem megbízható forgalomirányító eszközök EIGRP frissítéseket kapjanak.

Az 1. ábrán látható, hogy az R1, R2 és az R3 eszközök nem rendelkeznek szomszédokkal a GigabitEthernet 0/0 interfészükön.

A **passive-interface** forgalomirányító konfigurációs módbeli parancs letiltja az EIGRP Hello csomagok küldését és fogadását ezeken az interfészeken.

Router(config)# **router eigrp** *autonóm\_rendszer\_azonosító*

Router(config-router)# **passive-interface** *interfész\_típus interfész\_szám*

A 2. ábrán a **passive-interface** parancs konfigurálása látható, amely letiltja az R1 és az R3 helyi hálózatain a Hello csomagokat. Az R2 konfigurálása a parancsszimulátorban végezhető el.

Szomszédsági viszony kialakítása nélkül az EIGRP nem képes útvonalakat cserélni egy szomszéddal. Így tehát a **passive-interface** parancs az útvonalak cseréjét is megakadályozza az interfészen. Habár az EIGRP nem küld és nem fogad útvonalfrissítéseket a **passive-interface** paranccsal konfigurált interfészeken, az interfész címe attól még benne van az útvonalfrissítésekben, amelyeket a nem passzív interfészeken küld ki.

**MEGJEGYZÉS:**Az összes interfész passziválása a **passive-interface default** parancs használatával állítható be. Az interfész passzív állapotának megszüntetésére a **no passive-interface** *interfész\_típus interfész\_szám* parancs használható.

Passzív interfész használatával például akkor növelhető a biztonság, ha a hálózatot egy olyan idegen szervezethez kell csatlakoztatni, például egy internetszolgáltatóhoz, amely fölött a rendszergazdának nincs befolyása. Ebben az esetben a rendszergazda érdeke, hogy az interfész csak a saját hálózatán belül legyen hirdetve, és egy harmadik fél ne végezhesse az irányítási frissítések cseréjét a helyi forgalomirányítóval, mivel az biztonsági kockázatot jelentene.

**Passzív interfész ellenőrzése**

Egy interfész passzív állapotának ellenőrzésére a **show ip protocols** privilegizált módbeli parancs használható a 3. ábrán látható módon. Figyeljük meg, hogy az EIGRP útvonalfrissítés annak ellenére tartalmazza az R3 GigabitEthernet 0/0 interfészének 192.168.1.0 hálózatát, hogy az interfészt passzíváltuk.

A 4. ábrán lévő parancsszimulátorban állítsuk be, hogy az R2 ne küldjön EIGRP Hello csomagokat a GigabitEthernet 0/0 interfészén!

**Az EIGRP ellenőrzése: a szomszédság vizsgálata**

Az EIGRP frissítések küldése vagy fogadása előtt a forgalomirányítónak szomszédsági viszonyt kell kiépítenie a szomszédjaival. Az EIGRP-t futtató forgalomirányítók Hello csomagok cseréjével építik ki a szomszédsági viszonyt a szomszédos forgalomirányítókkal.

A **show ip eigrp neighbors** parancs használható a szomszédsági tábla megjelenítéséhez, valamint az EIGRP kiépített szomszédsági kapcsolatainak ellenőrzésére. Minden egyes forgalomirányító esetén a kimenetben látnunk kell a szomszédos forgalomirányító IP-címét, és a szomszéd eléréséhez használt interfészt. A minta topológiában mindegyik forgalomirányítónak két szomszédja van, ez látható R1 szomszédsági táblájában is.

A **show ip eigrp neighbors** parancs kimenete a következőket tartalmazza:

* **H oszlop** - Kilistázza a szomszédokat a megismerésük sorrendjében.
* **Cím (Address)** - A szomszéd IPv4-címe
* **Interfész (Interface)** - A helyi interfész, amelyen a Hello csomag érkezett.
* **Megtartási idő (Hold)** - A megtartási idő aktuális értéke. Hello csomag érkezésekor az interfészen ez a számláló visszaáll a legmagasabb értékre, és onnan kezd el csökkenni. Amint eléri a nullát, a szomszédot a forgalomirányító elérhetetlennek minősíti.
* **Üzemidő (Uptime)** - A szomszédsági táblában létrejött bejegyzés óta eltelt idő.
* **Körülfordulási idő (Smooth Round Trip Timer, SRTT)** és **Újraküldési idő (Retransmission Timeout** ,**RTO**) - Az RTP által a megbízható EIGRP csomagok kezelésére használt mezők.
* **Várakozási sor (Queue Count, Q Cnt)** - Az értékének mindig nullának kell lennie. Ha nagyobb mint nulla, akkor EIGRP csomagok vannak a várakozási sorban.
* **Sorszám (Sequence Number, Seq Num)** - Frissítő, lekérdező és válasz csomagok nyomon követésére használható.

A **show ip eigrp neighbors** parancs az EIGRP ellenőrzésénél és hibaelhárításánál jelent nagy segítséget. Ha a forgalomirányító szomszédsági viszonyainak kiépítése után egy szomszéd nem jelenik meg a táblában, akkor a helyi interfész aktív állapotának ellenőrzését a **show ip interface brief** parancs segítségével végezhetjük el. Aktív interfész esetén a szomszéd IPv4-címének megpingelésével próbálkozhatunk. A sikertelen pingelés azt jelzi, hogy a szomszéd interfésze lekapcsolt állapotban van és aktiválni kell. Ha a ping sikeres, az EIGRP viszont még mindig nem látja a forgalomirányítót szomszédnak, akkor a következő beállításokat ellenőrizzük:

* Mindkét forgalomirányító azonos autonóm rendszer azonosítóval lett konfigurálva?
* Meg lett adva a közvetlenül csatlakozó hálózat az EIGRP **network** utasításaiban?

# Az EIGRP ellenőrzése: a show ip protocols parancs

A **show ip protocols** parancs a forgalomirányítón konfigurált IPv4 irányító protokoll folyamatainak aktuális állapotáról jelenít meg paramétereket és egyéb információkat. A **show ip protocols** parancs az adott forgalomirányító protokolltól függően eltérő kimenetet adhat.

Az 1. ábra parancskimenetéből számos EIGRP paraméter olvasható ki, többek között:

1. Az R1-en az 1-es autonóm rendszer azonosítóval konfigurált EIGRP az aktív dinamikus forgalomirányító protokoll.

2. Az R1 forgalomirányító azonosítója 1.1.1.1.

3. Az R1-en az EIGRP-hez tartozó belső adminisztratív távolság 90, a külső pedig 170 (alapértelmezett értékek).

4. Alapértelmezés szerint az EIGRP nem végez automatikus útvonal összevonást. Az útvonalfrissítések az alhálózatokat is tartalmazzák.

5. Az R1 forgalomirányító EIGRP szomszédsági viszonyai, ahonnan EIGRP útvonalfrissítéseket fogad.

**MEGJEGYZÉS:**Az IOS 15-ös verzióját megelőzően az EIGRP automatikus útvonal-összevonása alapértelmezés szerint engedélyezett volt.

A **show ip protocols** parancs kimenete forgalomirányítási műveletek hibakeresésénél hasznos. Az irányítási információk forrása (Routing Information Sources) mezőben található adatok segíthetnek a feltételezhetően hibás irányítási információkat továbbító forgalomirányító azonosításában. A mező minden olyan EIGRP irányítási forrást megjelenít, amelyeket a Cisco IOS szoftver az IPv4-irányítótábla felépítéséhez használ. Minden egyes forráshoz a következőket jegyzi fel:

* IPv4-cím
* Adminisztratív távolság
* Az utolsó frissítés óta eltelt idő

Ahogy a 2. ábrán is látható, az EIGRP belső útvonalaihoz az alapértelmezett 90-es adminisztratív távolság, a külső forrástól importált útvonalakhoz, például az alapértelmezett útvonalakhoz pedig a 170-es érték van hozzárendelve. Más belső protokollokkal szemben a Cisco IOS az EIGRP-t részesíti előnyben, mivel az rendelkezik a legkisebb adminisztratív távolsággal. Az EIGRP összevont útvonalak 5-ös értéke a harmadik legjobb adminisztratív távolság érték.

**Az EIGRP ellenőrzése: Az IPv4-irányítótábla vizsgálata**

Egy másik mód annak ellenőrzésére, hogy az EIGRP és a forgalomirányító egyéb funkciói megfelelően vannak konfigurálva, az IPv4-irányítótábla vizsgálata a **show ip route** paranccsal. Bármelyik dinamikus forgalomirányító protokoll is legyen használatban, a hálózati rendszergazdának minden esetben ellenőriznie kell az irányítótáblában található információkat, hogy a konfigurációnak megfelelő adatokat tartalmazza-e. Emiatt fontos az irányító protokollok konfigurációs utasításainak, a protokollok működésének és az irányítótábla felépítéséhez használt folyamatoknak a megértése.

Ne felejtsük el, hogy a kurzus során használt parancskimenetek a Cisco IOS 15-ös verziójából származnak. A korábbi verziókban az EIGRP automatikus útvonal-összevonás alapértelmezés szerint engedélyezett volt. Az automatikus útvonal-összevonás ki és bekapcsolt állapotától függően eltérő információk jelenhetnek meg az IPv4-irányítótáblában. Az IOS korábbi verziói esetén az automatikus útvonal-összevonást a **no auto-summary** forgalomirányító konfigurációs módbeli paranccsal lehet kikapcsolni:

Router(config-router)# **no auto-summary**

Az 1. ábra az R1, R2 és az R3 forgalomirányítókat tartalmazó topológiát mutatja.

A 2. ábrán az IPv4-irányítótábla vizsgálata látható a **show ip route** parancs segítségével. Az irányítótáblában az EIGRP útvonalak **D**jelöléssel szerepelnek. Az EIGRP jelöléséhez használt D betű azzal magyarázható, hogy a protokoll a DUAL algoritmuson alapul.

A **show ip route** paranccsal ellenőrizhető, hogy az EIGRP szomszédoktól érkezett útvonalak bekerültek-e az irányítótáblába. A **show ip route** parancs az irányítótábla teljes tartalmát megjeleníti, beleértve a dinamikusan megtanult távoli hálózatokat, a közvetlenül csatlakozó és a statikus útvonalakat. Éppen ezért ez az elsődlegesen használt parancs a konvergencia vizsgálatakor. A forgalomirányítás megfelelő konfigurálását követően az összes forgalomirányítón a **show ip route** parancs azt mutatja, hogy a forgalomirányítók irányítótáblái teljesek, azaz a topológia minden hálózatához tartozik útvonal.

Figyeljük meg, hogy az R1 három távoli IPv4-hálózathoz vett fel útvonalat az irányítótáblájába:

* A 172.16.2.0/24 hálózatot, amelyet az R2 forgalomirányítótól kapott a Serial0/0/0 interfészén keresztül.
* A 192.168.1.0/24 hálózatot, amelyet az R3 forgalomirányítótól kapott a Serial0/0/1 interfészén keresztül.
* A 192.168.10.8/30 hálózatot, amelyet az R2 forgalomirányítótól a Serial0/0/0, és az R3 forgalomirányítótól a Serial0/0/1 interfészén keresztül is megkapott.

Az R1 két útvonallal is rendelkezik a 192.168.10.8/30 hálózathoz, mivel a hálózat eléréséhez tartozó költség vagy mérték azonos bármelyik forgalomirányítót is használjuk. Ezek az úgynevezett egyenlő költségű útvonalak. Az R1 mindkét útvonalat használja a hálózat eléréséhez, ezt hívjuk terheléselosztásnak (load balancing). Az EIGRP mértékről a fejezet későbbi részében lesz szó.

A 3. ábra az R2 irányítótábláját mutatja. Figyeljük meg, hogy itt is hasonló értékek jelennek benne, beleértve a 192.168.10.4/30 hálózathoz tartozó egyenlő költségű útvonalat is.

A 4. ábra az R3 irányítótábláját mutatja. Az eredmény hasonló az R1 és az R2 eszközöknél látottakhoz. Az R3 a távoli hálózatokat EIGRP használatával tanulta meg, beleértve a 172.16.3.0/30 hálózathoz tartozó egyenlő költségű útvonalat is.

# Laborgyakorlat - EIGRP alapszintű konfigurálása IPv4-hálózatban

**Ebben a gyakorlatban a következő feladatokat végezzük el:**

* 1. rész: A hálózat felépítése és a kapcsolatok ellenőrzése.
* 2. rész: EIGRP forgalomirányítás beállítása.
* 3. rész: EIGRP forgalomirányítás ellenőrzése.
* 4. rész: Sávszélesség és passzív interfészek beállítása.

[Laborgyakorlat - Configuring Basic EIGRP for IPv4](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/7.2.2.5%20Lab%20-%20Configuring%20Basic%20EIGRP%20for%20IPv4.pdf)

# EIGRP szomszédsági viszonyok

Minden dinamikus irányító protokoll célja, hogy ismereteket szerezzen távoli hálózatokról a többi forgalomirányítótól és megvalósítsa a konvergenciát az irányítási tartományon belül. Mielőtt megkezdődhetne a forgalomirányítók közötti EIGRP frissítő csomagok cseréje, fel kell fedezniük saját szomszédjaikat. Az EIGRP-szomszédok a közvetlenül kapcsolódó hálózatokon található, szintén EIGRP-t futtató forgalomirányítók.

Az EIGRP Hello csomagokat használ a szomszédsági viszonyok kialakításához és fenntartásához. Ahhoz, hogy két EIGRP-t futtató forgalomirányító egymás szomszédja lehessen, számos paraméternek egyeznie kell közöttük. Például ugyanazt az EIGRP mértéket kell használniuk és mindkettőn azonos autonóm rendszer azonosító használatát kell beállítani.

Minden EIGRP forgalomirányító egy szomszédtáblát tart fenn, amely a vele szomszédsági viszonyban lévő forgalomirányítók listáját tartalmazza. Ez a szomszédtábla az EIGRP szomszédok állapotának nyomon követésére használható.

Az ábrán látható, ahogy két EIGRP forgalomirányító Hello csomagokat cserél egymással. Ha egy forgalomirányító Hello csomagot kap valamely interfészén, akkor felveszi az adott forgalomirányítót a szomszédtáblájába.

1. Egy új forgalomirányító (R1) elérhetővé válik a kapcsolaton és EIGRP Hello csomagot küld ki minden EIGRP interfészén.

2. Az R2 forgalomirányító fogadja a Hello csomagot az EIGRP interfészén, és egy EIGRP frissítő üzenettel válaszol. A frissítés tartalmazza az irányítótáblája összes útvonalát, kivéve azokat, amelyeket ezen az interfészen keresztül tanult meg (látóhatár megosztás). A szomszédsági viszony azonban mindaddig nem jön létre, amíg az R2 szintén nem küld egy Hello csomagot az R1-nek.

3. A forgalomirányítók Hello csomagjainak cseréje után a szomszédsági viszony létrejön. Az R1 és az R2 frissítik az EIGRP szomszédtáblájukat és szomszédként veszik fel a másik eszközt.

# EIGRP topológiatábla

Az EIGRP frissítések azokat a hálózatokat tartalmazzák, amelyek elérhetőek a frissítést küldő forgalomirányító számára. A szomszédok közti frissítő csomagok cseréje után a fogadó forgalomirányító felveszi ezeket a bejegyzéseket az EIGRP topológiatáblájába.

Minden EIGRP forgalomirányító külön topológiatáblát tart fenn a konfigurált irányított protokolljai, például az IPv4 és az IPv6 számára. Ez a tábla minden célhálózathoz eltárolja azokat az útvonalakat, amelyeket a forgalomirányító a közvetlenül csatlakozó EIGRP szomszédaitól tanult.

Az ábra az előző oldalon lévő kezdeti útvonal felderítési folyamat folytatását mutatja. Most a topológiatábla frissítése látható rajta.

Az útvonalfrissítés fogadása után a forgalomirányító hozzáadja az irányítási információt az EIGRP topológiatáblához majd egy nyugtacsomaggal válaszol.

1. Az R1 fogadja az EIGRP frissítést az R2-től, amely a szomszéd által hirdetett útvonalakról tartalmaz információkat, például minden célhoz a hozzá tartozó mértéket. Ezt követően az R1 hozzáadja az összes frissítő információt a topológiatáblájához. A topológiatábla a szomszédos forgalomirányítók által hirdetett összes célt, valamint a hálózatok eléréshez szükséges mértékeket tartalmazza.

2. Az EIGRP frissítő üzenetek megbízható kézbesítést használnak, ezért az R1 egy EIGRP nyugtacsomaggal válaszol az R2-nek, értesítve ezzel őt a frissítés fogadásáról.

3. Az R1 egy EIGRP frissítést küld az R2-nek, amelyben az általa ismert útvonalakat hirdeti a számára azok kivételével, amelyeket az R2-től tanult (látóhatár megosztás).

4. Az R2 fogadja az EIGRP frissítést az R1 nevű szomszédjától, majd hozzáadja a benne lévő információkat a topológiatáblájához.

5. Az R1 frissítő csomagjára az R2 válaszként egy EIGRP nyugtaüzenetet küld.

# EIGRP konvergencia

Az ábra a kezdeti útvonal felderítés folyamatának utolsó lépéseit mutatja.

1. Az R2-től érkező EIGRP frissítő csomag fogadása után az R1 a topológiatábla információinak felhasználásával frissíti az IP-irányítótábláját a célokhoz vezető legjobb útvonallal, valamint a hozzájuk tartozó mértékekkel és következő ugrásokkal.

2. Az R1-hez hasonlóan az R2 is frissíti az irányítótábláját a hálózatokhoz vezető legjobb útvonallal.

Ezen a ponton tekinthető az EIGRP mindkét forgalomirányítón konvergált állapotúnak.

# EIGRP összetett mérték

Alapértelmezés szerint az EIGRP a következő értékeket használja a legjobb útvonalak meghatározásánál alkalmazott összetett mértékben:

* **Sávszélesség** - A forrás és a cél közötti útvonalon a kimenő interfészeken megtalálható legkisebb sávszélesség.
* **Késleltetés** - Az útvonalon szereplő interfészek összesített késleltetése (mikroszekundumban).

A most következő értékek használata megengedett, de nem ajánlott, mert jellemzően a topológiatábla gyakori újraszámítását eredményezik.

* **Megbízhatóság** - Megadja a forrás és a cél közötti útvonalon lévő értékek közül a legrosszabb megbízhatóságot, amelynek az életjelek az adnak alapot.
* **Terhelés** - Megadja az interfészen beállított sávszélesség és a csomagok számából meghatározott legrosszabb terhelési értéket a forrás és a cél között.

**MEGJEGYZÉS:**Annak ellenére, hogy az irányítótábla frissítései tartalmazzák az MTU-t, az nem tartozik az EIGRP irányítási mértékei közé.

**Összetett mérték**

Az 1. ábra az EIGRP által használt összetett mérték kiszámítási képletét mutatja. A képlet K1-től K5-ig tartalmaz paramétereket, ezek az úgynevezett EIGRP mérték súlyozások. K1 a sávszélességet, K3 pedig a késleltetést jelöli. K2 jelenti a terhelést, K4 és K5 pedig a megbízhatóságot. Alapértelmezésben a K1 és K3 értéke 1, a K2, K4 és K5 pedig 0. Ennek eredményeképpen alapértelmezetten az összetett mérték kiszámítása csak a sávszélesség és a késleltetés értékek alapján történik. A számításhoz használt képlet IPv4 és IPv6 esetén megegyezik.

A mérték számítási módjának (*k* értékek) és az autonóm rendszer azonosítónak egyeznie kell a szomszédos EIGRP forgalomirányítókon. Ellenkező esetben a forgalomirányítók nem alakítanak ki szomszédsági viszonyt.

Az alapértelmezett *k* értékek megváltoztathatók a **metric weights** forgalomirányító konfigurációs módbeli paranccsal:

Router(config-router)# **metric weights** *tos* *k1 k2 k3 k4 k5*

**MEGJEGYZÉS:**A **metric weights** paraméterek megváltoztatása azonban nem ajánlott és túlmutat a kurzus anyagán. A szomszédsági viszonyok kialakításában viszont jelentős szerepük van. Abban az esetben, ha egy forgalomirányítón megváltoztatásra kerülnek a mérték súlyozások egy másikon pedig nem, akkor szomszédság nem jöhet létre közöttük.

**A *k* értékek ellenőrzése**

A **show ip protocols** parancs használható a *k* értékek ellenőrzésére. A parancs egy lehetséges kimenete látható a 2. ábrán. Figyeljük meg, hogy a *k* értékek alapértelmezettre vannak állítva.

**Interfészek értékeinek vizsgálata**

**A mértékek vizsgálata**

A **show interfaces** parancs interfészekre vonatkozó információkat jelenít meg, többek között az EIGRP mérték számításához használt paramétereket is. Az ábrán a **show interfaces** parancs kimenete látható az R1 Serial 0/0/0 interfészére vonatkozóan.

* **BW** - Az interfész sávszélessége (kbit/s)
* **DLY** - Az interfész késleltetése (mikroszekundum)
* **Megbízhatóság** - Az interfész megbízhatósága, amely 255 hányadosaként van megadva (255/255 jelenti a 100%-os megbízhatóságot), és értékét 5 perc átlagából számítják. Alapértelmezés szerint az EIGRP nem használja ezt az értéket a mérték meghatározásához.
* **Txload, Rxload** - A küldés és fogadás során az interfészen fellépő terhelés, amely 255 hányadosaként van megadva (255/255 jelenti a teljes telítettséget), és értékét 5 perc átlagából számítják. Alapértelmezés szerint az EIGRP nem használja ezt az értéket a mérték meghatározásához.

**MEGJEGYZÉS:**A tanfolyam során a sávszélesség értékét mindig kbit/s-ban értjük. A forgalomirányítók kimenetében megjelenő sávszélesség értékeknél viszont a Kbit/sec rövidítés látható. A kimenetekben a késleltetés értéke 'usec' mértékegységben jelenik meg. A kurzusban a késleltetés mindig mikroszekundummal megadva szerepel.

# A sávszélesség

Bizonyos irányító protokollok, például az EIGRP és az OSPF által az irányítási mérték kiszámításához használt sávszélesség egy statikus érték. A sávszélesség kilobit/másodperc (kb/s) értékben jelenik meg. A legtöbb soros interfész az alapértelmezett 1544 kbit/s-os vagy 1544000 bit/s-os értéket (1,544 Mbit/s) használja. Ez a T1 kapcsolatnak megfelelő sávszélesség. Néhány soros interfész viszont ettől eltérő alapértelmezett sávszélesség értéket használ. Az 1. ábrán az ebben a részben használt hálózati topológia látható. A soros interfészek típusa és sávszélessége nem szükségszerűen felel meg a napjainkban elterjedtebb kapcsolattípusoknak.

A sávszélesség értékét mindig ellenőrizzük le a **show interfaces** paranccsal.

A sávszélesség alapértelmezett értéke lehet, hogy nem tükrözi az adott interfész tényleges fizikai sávszélességét. Ha a kapcsolatok tényleges sávszélessége különbözik az alapértelmezett sávszélesség értéktől, a sávszélesség értékét módosítani kell.

**A sávszélesség beállítása**

A legtöbb soros kapcsolat alapértelmezett sávszélessége 1544 kbit/s. Mivel az EIGRP és az OSPF is használja a sávszélességet a mérték alapértelmezett számításában, ezért a sávszélesség megfelelő értéke nagyon fontos a forgalomirányítási információk pontossága érdekében.

A sávszélesség értékének megváltoztatásához használjuk a következő interfész konfigurációs módbeli parancsot:

Router(config-if)# **bandwidth** *sávszélesség-érték-kbps-ban.*

Használjuk a **no bandwidth** parancsot az alapértelmezett érték visszaállításához.

A 2. ábrán látható, hogy az R1 és az R2 közötti kapcsolat sávszélessége 64 kbit/s, az R2 és az R3 közöttié pedig 1024 kbit/s. Az ábrán a három forgalomirányító megfelelő soros interfészein a sávszélesség értékek módosítására alkalmazott konfigurációk láthatók.

**A sávszélesség értékének ellenőrzése**

Használjuk a **show interfaces** parancsot az új sávszélesség értékek ellenőrzésére a 3. ábrán látható módon. Fontos, hogy a sávszélesség értékek a kapcsolat mindkét oldalán meg legyenek változtatva, hogy a megfelelő forgalomirányítás mindkét irányban biztosított legyen.

A sávszélesség értékének módosítása a kapcsolat tényleges sávszélességét nem változtatja meg. A **bandwidth** parancs csak az irányító protokollok (például az EIGRP és az OSPF) által használt sávszélesség mértéket módosítja.

# Az EIGRP mérték kiszámítása

Habár az EIGRP automatikusan kiszámítja a legjobb útvonal megállapításához használt irányítási mértéket, fontos, hogy a hálózati rendszergazda megértse a mértékek meghatározásának módját.

Az ábra az EIGRP által használt összetett mértéket mutatja. A K1 és K3 alapértelmezett értékeit használva a számítás leegyszerűsíthető a leglassabb (vagy minimális) sávszélességre és az összesített késleltetés összegére.

Más szóval a sávszélesség és késleltetés értékeit az útvonal minden egyes kimenő interfészén megvizsgálva megkaphatjuk az EIGRP mértéket a következő módon:

**1. lépés** Határozzuk meg a legkisebb sávszélességű kapcsolatot. Használjuk ezt az értéket a sávszélesség kiszámításához (10000000/sávszélesség).

**2. lépés** Határozzuk meg a késleltetés értékét a célhoz vezető útvonal minden egyes kimenő interfészére. Adjuk össze a késleltetés értékeket és osszuk el 10-zel (összes késleltetés/10).

**3. lépés** Az összetett mérték egy 24-bites értéket ad eredményül, az EIGRP viszont 32-bites értéket használ. Ha megszorozzuk a 24-bites értéket 256-tal, az összetett mérték kiegészül 32-bitre. Ezért kell az EIGRP mérték meghatározásához a sávszélesség és a késleltetés számított értékeit összeadni, majd az összeget megszorozni 256-tal.

Az R2 irányítótáblájának kimenete alapján láthatjuk, hogy a 192.168.1.0/24 hálózathoz tartozó útvonal EIGRP mértéke 3012096.

# Az EIGRP mérték kiszámítása

Az 1. ábra a három forgalomirányítóból álló topológiát mutatja. A példa azt szemlélteti, hogy az EIGRP milyen módon határozza meg az R2 irányítótáblájában megjelenő mértéket a 192.168.1.0/24 hálózat számára.

**Sávszélesség**

Az EIGRP a legkisebb sávszélesség értéket használja a mérték számításakor. A legkisebb sávszélesség az R2 és a 192.168.1.0 célhálózat között található interfészek vizsgálatával határozható meg. Az R2 Serial 0/0/1 interfésze 1024 kbit/s nagyságú sávszélességgel rendelkezik, az R3 GigabitEthernet 0/0 interfészén pedig 1.000.000 kbit/s a sávszélesség. Ezért a mérték kiszámításához használt legkisebb sávszélesség értéke 1024 kbit/s.

EIGRP esetén a 10.000.000 nagyságú referencia sávszélesség van elosztva az interfész kbit/s-ban megadott sávszélességével. Ez a számítás nagyobb sávszélesség esetén kisebb mértéket, alacsonyabb sávszélesség esetén nagyobb mértéket eredményez. 10.000.000 osztva 1024-gyel. Ha az eredmény nem egész szám, akkor az értéket lefelé kell kerekíteni. Ebben az esetben a 10.000.000 osztva 1024-el 9756,625-öt eredményez. A 0,625-öt elhagyva 9765 lesz az összetett mérték sávszélessége. (Lásd 2. ábra)

**Késleltetés**

A 3. ábrán látható, hogy a késleltetés értékének meghatározásához ugyanazokat a kimenő interfészeket kell használni.

Az EIGRP a célhoz vezető útvonal késleltetéseinek összegét használja. Az R2 Serial 0/0/1 interfészének késleltetése 20000 mikroszekundum, az R3 GigabitEthernet interfészéé pedig 10 mikroszekundum. Ezen késleltetések összegét kell elosztani 10-zel. A példában a (20000+10)/10 művelet eredménye 2001, ami az összetett mérték késleltetése.

**A mérték számítása**

Helyettesítsük be a sávszélesség és a késleltetés számított értékeit a mérték képletébe. Ez 3012096 nagyságú mértéket eredményez. (Lásd 4. ábra) Ez az érték megegyezik az R2 irányítótáblájában megjelenő értékkel.

**Fogalmak**

Az EIGRP a szétszóró frissítő algoritmust (DUAL) használja a hurokmentes útvonalak és a hurokmentes tartalék útvonalak meghatározásához.

A DUAL számos olyan kifejezést használ, amelyekkel ebben a fejezetben részletesebben is foglalkozunk:

* Legjobb útvonal (Successor)
* Legkisebb távolság (Feasible Distance, FD)
* Tartalék útvonal (Feasible Successor, FS)
* Jelentett távolság (Reported Distance, RD) vagy hirdetett távolság (Advertised Distance, AD)
* Az alkalmasság feltétele (Feasible Condition, FC)

Ezek a kifejezések és fogalmak állnak a DUAL algoritmus hurokelkerülési mechanizmusának középpontjában.

**Bevezetés a DUAL algoritmusba**

Az EIGRP protokoll a DUAL algoritmust használja a konvergencia biztosításához. A konvergencia az irányítási hurkok elkerülése miatt fontos a hálózatoknál.

Az irányítási hurkok, akár az ideiglenesek is hátrányosan befolyásolhatják a hálózat teljesítményét. A távolságvektor alapú irányító protokollok, mint például a RIP visszatartási időzítők és látóhatár megosztás alkalmazásával akadályozzák meg az irányítási hurkok létrejöttét. Az EIGRP is alkalmazza ezeket a technikákat, de egy kicsit másképp; az EIGRP az irányítási hurkok elkerülését elsődlegesen a DUAL algoritmus alkalmazásával oldja meg.

Kattintsunk az ábra Lejátszás gombjára a DUAL algoritmus alapvető működésének megtekintéséhez!

Az útvonalszámítás minden lépésében a hurokmentességet a DUAL algoritmus biztosítja. Ez a topológiaváltozásban érintett minden forgalomirányító számára lehetővé teszi az egyidejű szinkronizációt. A topológiaváltozásban nem érintett forgalomirányítók nem vesznek részt az újraszámításban. Ez a módszer gyorsabb konvergenciaidőt biztosít az EIGRP számára a többi távolságvektor alapú irányító protokollhoz képest.

A döntési folyamatot minden útvonalszámításnál a DUAL véges állapotú automata végzi el (Finite State Machine, FSM). Az FSM egy a folyamatábrához hasonló munkafolyamat (workflow) modell, amely a következő részekből áll:

* Véges számú állapot
* Az állapotok közötti átmenetek
* Műveletek

A DUAL FSM minden útvonalat nyomon követ, az EIGRP mérték segítségével kiválasztja a megfelelő hurokmentes útvonalakat, majd meghatározza a legkisebb költséggel rendelkezőket, amelyeket az irányítótáblába kell beilleszteni.

A DUAL algoritmus újraszámítása processzor igényes művelet lehet, így az EIGRP lehetőség szerint elkerüli azt. Ennek érdekében egy olyan tartalék útvonal listát is kezel, amelyekről a DUAL már eldöntötte, hogy hurokmentesek. Amennyiben az irányítótáblában lévő elsődleges útvonal elérhetetlenné válik, a legjobb tartalék útvonal azonnal az irányítótáblába kerül helyette.

# Legjobb útvonal és legkisebb távolság

Az 1. ábrán a témához kapcsolódó hálózati topológia látható. A legjobb útvonal egy olyan csomagtovábbításra használt szomszédos forgalomirányító, amely a célhálózathoz vezető legkisebb költségű útvonalon található. A legjobb útvonal IP-címe az irányítótábla bejegyzésekben is megjelenik közvetlenül a via kulcsszó után.

A legkisebb távolság (FD) a legalacsonyabb számított mérték a célhálózat eléréséhez. Értéke az irányítótábla bejegyzésekben lévő szögletes zárójel második száma. Ahogy más irányító protokollnál is, ezt az értéket is az útvonal mértékének nevezzük.

A 2. ábrán az R2 irányítótábláját megvizsgálva láthatjuk, hogy a 192.168.1.0/24 hálózathoz tartozó legjobb útvonal az R3 forgalomirányítón keresztül vezet, és a legkisebb távolság (FD) értéke 3012096. Ezt ugyanaz az érték, amit az előző részben kiszámoltunk.

**Tartalék útvonalak (FS), az alkalmasság feltétele (FC) és a jelentett távolság (RD)**

Egy topológiaváltozást követően a DUAL gyors konvergenciát tesz lehetővé, mivel a hálózatok elérésére tartalék útvonalakat is képes használni az algoritmus újraszámítása nélkül. Ezeket a tartalék útvonalakat más néven második legjobb útvonalaknak is hívjuk (Feasible Successor, FS).

Az FS egy olyan szomszéd, amely hurokmentes tartalék útvonallal rendelkezik ugyanahhoz a hálózathoz, mint a legjobb útvonal, valamint megfelel az alkalmassági feltételnek (Feasibility Condition, FC). Az R2 legjobb útvonala a 192.168.1.0/24 hálózathoz az R3, amely a legalacsonyabb mértékű, vagyis a legjobb útvonalat biztosítja a célhálózathoz. Figyeljük meg az 1. ábrán, hogy az R1 rendelkezik egy alternatív útvonallal is. Vajon ez a második legjobb útvonal (FS)? Mielőtt az R1 tartalék útvonal lehetne az R2 számára, teljesítenie kell az alkalmassági feltételt (FC).

Az alkalmassági feltétel akkor teljesül, ha a szomszéd jelentett távolsága (Reported Distance, RD) egy hálózathoz kisebb, mint a helyi forgalomirányító legkisebb távolsága (Feasible Distance, FD) ugyanahhoz a hálózathoz. A kisebb jelentett távolság egy hurokmentes útvonalat is jelent egyben. A jelentett távolság egyszerűen csak egy EIGRP szomszéd legjobb távolságát jelenti ugyanahhoz a célhálózathoz. A jelentett távolság tehát egy olyan mérték, amelyet egy forgalomirányító hirdet a szomszédja felé a hálózathoz tartozó saját költségével.

A 2. ábrán látható, hogy az R1 legkisebb távolsága a 192.168.1.0/24 hálózathoz 2170112.

* Az R1 R2-nek küldött hirdetése tartalmazza, hogy a 192.168.1.0/24 hálózathoz tartozó legkisebb távolsága 2170112.
* Az R2 szemszögéből nézve az R1 hirdetett távolsága 2170112.

Az R2 ezt az információt használja annak meghatározására, hogy az R1 teljesíti-e az alkalmassági feltételt, és így második legjobb útvonal lehet-e.

Ahogy azt a 3. ábra is mutatja az R1 hirdetett távolsága (2170112) kisebb, mint az R2 saját legkisebb távolsága (3012096), ezért az R1 lesz a tartalék útvonal.

Így most az R1 a második legjobb útvonal az R2-n a 192.168.1.0/24 hálózat számára.

Amennyiben meghibásodás történik az R2 azon útvonalán, amellyel a 192.168.1.0/24 hálózat érhető el az R3 forgalomirányítón keresztül (legjobb útvonal), akkor az R2 azonnal felveszi az irányítótáblába az R1-en keresztül vezető útvonalat (második legjobb útvonal). Így az R1 lesz az új legjobb útvonal az R2-n az adott hálózat felé, ahogy ez a 4. ábrán is látható.

# Topológiatábla: a show ip eigrp topology parancs

Az 1. ábrán a hálózati topológia látható.

Az EIGRP topológiatábla az összes olyan útvonalat tartalmazza, amelyek ismertek az EIGRP szomszédok számára. Amikor egy EIGRP forgalomirányító útvonalakat tanul a szomszédaitól, akkor az új útvonalak egyből belekerülnek a topológiatáblába.

A 2. ábrán látható módon használjuk a **show ip eigrp topology** parancsot a topológiatábla megtekintéséhez. A topológiatábla az összes olyan legjobb és második legjobb útvonalat tartalmazza, amelyeket a DUAL algoritmus meghatározott a célhálózatokhoz. Az irányítótáblába csak a legjobb útvonalak kerülnek.

**Topológiatábla: show ip eigrp topology parancs (folytatás)**

Az 1. ábrán a topológiatábla első sorában a következők láthatók:

* **P** - Az útvonal passzív állapotban van. Ha a DUAL algoritmus nem végez számítást, hogy meghatározzon egy hálózat felé vezető útvonalat, az útvonal egy stabil, más néven passzív állapotba kerül. Ha az algoritmus újraszámolást vagy új útvonal keresést végez, akkor az útvonal aktív állapotba kerül és megjelenik mellette az A betű. Az irányítási tartomány stabil működéséhez a topológiatábla minden útvonalának passzív állapotúnak kell lennie.
* **192.168.1.0/24** - A célhálózat, amely az irányítótáblában is megtalálható.
* **1 legjobb útvonal** - A hálózathoz tartozó legjobb útvonalak számát jeleníti meg. Ha a hálózathoz több egyforma költségű útvonal is létezik, akkor több legjobb útvonal van.
* **FD értéke 3012096** - A legkisebb távolság (FD) érték, ami a célhálózat eléréséhez tartozó EIGRP mérték. Ez az érték szerepel az irányítótáblában is.

A 2. ábrán látható, hogy a kimenet első albejegyzése a legjobb útvonalat mutatja:

* **via 192.168.10.10** - A legjobb útvonalhoz (R3) tartozó következő ugrás IP-címe. Ez a cím az irányítótáblában is megjelenik.
* **3012096** - A 192.168.1.0/24 hálózathoz tartozó legkisebb távolság (FD). Ez az érték szerepel az irányítótáblában is.
* **2816** - A legjobb útvonalhoz tartozó jelentett távolság (RD) és egyben az R3 költsége is a hálózat eléréséhez.
* **Serial 0/0/1** - A hálózat eléréséhez használt kimenő interfész, ami az irányítótáblában is megjelenik.

A 3. ábrán a második albejegyzés a tartalék útvonalat mutatja, amely az R1 (ha nincs második bejegyzés, akkor nincs tartalék útvonal).

* **via 172.16.3.1** - A tartalék útvonal következő ugrás IP-címe, amely az R1.
* **41024256** - Az R2 új legkisebb távolsága a 192.168.1.0/24 hálózat felé, ha az R1 lesz az új legjobb útvonal. Ez az új mérték jelenik meg az irányítótáblában.
* **2170112** - A második legjobb útvonalhoz (FS) tartozó jelentett távolság (RD), azaz az R1-hez tartozó mérték a célhálózat eléréséhez. Az alkalmassági feltétel (FC) teljesüléséhez az RD értéknek kisebbnek kell lennie az aktuális legkisebb távolság (FD) 3012096 értékénél.
* **Serial 0/0/0** - A tartalék útvonalhoz tartozó kimenő interfész, amennyiben ez a forgalomiránytó legjobb útvonallá válik.

# Topológiatábla: Nincs tartalék útvonal

Az 1. ábrán látható konvergált hálózat és az R1 irányítótáblájának vizsgálatával megfigyelhető, hogy miként használja a DUAL algoritmus a legjobb és második legjobb útvonalakat.

A 2. ábra a **show ip route** parancs kimenetének egy részletét mutatja az R1-en. A 192.168.1.0/24 hálózat útvonalánál látható, hogy az R3 a legjobb útvonal a 192.168.10.6 következő ugrás címmel és a 2170112 legkisebb távolság értékkel.

Az irányítótábla csak a legjobb útvonalakat tartalmazza. Tartalék útvonalak kereséséhez meg kell vizsgálnunk az EIGRP topológiatáblát. A 3. ábrán lévő topológiatábla csak a 192.168.10.6 következő ugráshoz, vagyis az R3-hoz tartozó legjobb útvonalat mutatja. Tartalék útvonalak nincsenek. A tényleges fizikai topológiát vagy hálózati diagramot megvizsgálva viszont nyilvánvalóvá válik, hogy létezik tartalék útvonal a 192.168.1.0/24 hálózathoz az R2-n keresztül. Az R2 nem lehet tartalék útvonal (FS), mivel nem teljesíti az alkalmassági feltételt (FC). A topológiát megnézve viszont egyértelmű, hogy az R2 egy tartalék útvonal, tehát az EIGRP nem rendelkezik a teljes topológiát tartalmazó térképpel. Az EIGRP egy távolságvektor alapú irányító protokoll és a távoli hálózatokról kizárólag a szomszédoktól szerez információkat.

A DUAL algoritmus nem tárol R2-n keresztül vezető útvonalat a topológiatáblában. Az összes kapcsolat megjeleníthető a **show ip eigrp topology all-links** paranccsal. A parancs attól függetlenül megjeleníti az összes kapcsolatot, hogy azok teljesítik-e az alkalmassági feltételt.

A 4. ábrán látható **show ip eigrp topology all-links** parancs megjeleníti a hálózathoz tartozó összes lehetséges útvonalat, beleértve a legjobb útvonalakat, valamint a tartalék és nem tartalék útvonalakat is. Az R1 legkisebb távolság értéke a 192.168.1.0/24 hálózathoz 2170112 az R3 legjobb útvonalon keresztül. Ahhoz, hogy az R2-t tartalék útvonalnak tekinthessük, teljesítenie kell az alkalmassági feltételt. A hirdetett távolság értékének, amelyet az R2 a 192.168.1.0/24 hálózatról hirdet R1-nek, kisebbnek kell lennie, mint az R1 jelenlegi legkisebb távolsága. Az ábra szerint az R2 hirdetett távolsága 3012096, ami nagyobb, mint az R1 aktuális legkisebb távolsága, azaz 2170112.

Annak ellenére, hogy az R2 a 192.168.1.0/24 hálózat felé valódi tartalék útvonalnak néz ki, az R1 nem tudja, hogy az útvonal esetleg nem egy önmagába visszatérő hurok útvonal. Az EIGRP egy távolságvektor alapú irányító protokoll, amely nem rendelkezik azzal a képességgel, hogy lássa a hálózat teljes, hurokmentes topológia térképét. A DUAL algoritmus úgy biztosítja a szomszédok útvonalainak hurokmentességét, hogy a szomszéd mértékének teljesítenie kell az alkalmassági feltételt. Azzal, hogy a szomszéd hirdetett távolsága kisebb, mint a forgalomirányító saját legkisebb távolsága, az eszköz feltételezheti, hogy a szomszédos forgalomirányító nem része az ő hirdetett útvonalának, és így a lehetséges hurkok mindig elkerülhetővé válnak.

Ha az R3 elérhetetlenné válik, akkor az R2 használható legjobb útvonalként. Az irányítótáblához történő hozzáadás előtt azonban van még egy hosszabb késleltetés. Mielőtt az R2 legjobb útvonallá válhatna, a DUAL algoritmusnak további számításokat kell elvégeznie.

# DUAL véges állapotú automata (Finite State Machine, FSM)

Az EIGRP központi eleme a DUAL algoritmus, amely az EIGRP útvonalszámító motorja. A technológia teljes neve DUAL véges állapotú automata (Finite State Machine, FSM). Az FSM tartalmazza mindazt a logikát, amely egy EIGRP hálózat útvonalainak kiszámításához és összehasonlításához szükséges. Az ábrán a DUAL FSM egyszerűsített változata látható.

A véges állapotú automata egy elvonatkoztatott értelemben vett gép, és nem valamiféle mozgó alkatrészekkel rendelkező szerkezet. Az FSM meghatározott állapothalmazzal rendelkezik, és pontosan megadja, hogy melyik állapotot milyen esemény idéz elő, illetve hogy ezekből az állapotokból milyen események következnek. A tervezők különféle készülékek, számítógépes programok vagy irányító algoritmusok meghatározott bemenetre adott válaszát szokták véges állapotú automatákkal modellezni.

A véges állapotú automata nem része a kurzus anyagának. Magát a fogalmat azonban használjuk, amikor az EIGRP véges állapotú automata kimenetét vizsgáljuk a **debug eigrp fsm** paranccsal. Ezzel a paranccsal vizsgálhatjuk meg a DUAL algoritmus viselkedését, ha egy útvonal eltávolításra kerül az irányítótáblából.

# DUAL: Tartalék útvonal (FS)

Az R2 jelenleg az R3-at használja a 192.168.1.0/24 hálózat legjobb útvonalaként. Továbbá az R2 tartalék útvonalként tartja nyilván az R1-et, ahogy ez az 1. ábrán látható.

A 2. ábrán az R2-n kiadott **show ip eigrp topology** parancs kimenete azt mutatja,hogy az R3 a legjobb útvonal, és az R1 a tartalék útvonal a 192.168.1.0/24 hálózat számára. Ahhoz, hogy megértsük, hogyan használja a DUAL algoritmus a tartalék útvonalat a legjobb útvonal elérhetetlenné válásakor, egy kapcsolati hibát szimulálunk az R2 és az R3 között.

A hiba szimulációjának megkezdése előtt engedélyezni kell az R2-n a DUAL algoritmus hibakeresését a **debug eigrp fsm** paranccsal, a 3. ábrán látható módon. A kapcsolati hiba szimulálását a **shutdown** parancs kiadásával végezhetjük el az R2 Serial 0/0/1 interfészén.

A **debug** parancs kimenetéből látható, hogy milyen tevékenységet végzett a DUAL algoritmus az összeköttetés lekapcsolásakor. Az R2 forgalomirányítónak minden EIGRP szomszédját értesítenie kell az elveszett kapcsolatról, valamint a saját irányító- és topológiatábláját is frissítenie kell. A példa csak egy kiválasztott részt mutat a **debug** parancs kimenetéből. Figyeljük meg, hogy a DUAL FSM tartalék útvonalat keres és talál is az EIGRP topológiatáblában.

Az R1 tartalék útvonal legjobb útvonallá válik és belekerül az irányítótáblába, mint az új legjobb útvonal a 192.168.1.0/24 hálózat felé. (Lásd 4. ábra) Ez a változás a tartalék útvonallal szinte azonnal megtörténik az irányítótáblában.

Az 5. ábrán látható, hogy az R2 topológiatáblájában az R1 lett a legjobb útvonal és nincsenek tartalék útvonalak. Ha az R2 és az R3 közötti kapcsolat újra elérhetővé válik, akkor megint az R3 lesz a legjobb útvonal és az R1 a tartalék útvonal.

# DUAL: Nincs tartalék útvonal

Esetenként előfordulhat, hogy az elsődleges útvonal elérhetetlenné válik és nincs tartalék útvonal. Ebben az esetben a DUAL algoritmus nem rendelkezik egy garantáltan hurokmentes tartalék útvonallal a hálózathoz, és így az útvonal nem is szerepel a topológiatáblában tartalék útvonalként. Ha a topológiatábla nem tartalmaz tartalék útvonalakat, a hálózat aktív állapotba kerül. A DUAL algoritmus ilyenkor a szomszédok aktív lekérdezésébe kezd új legjobb útvonalat keresve.

Az 1. ábrán látható, hogy az R1 jelenleg az R3-at használja a 192.168.1.0/24 hálózat legjobb útvonalaként, de nem használja R2-t mint tartalék útvonal, mivel R2 nem teljesíti az alkalmassági feltételt. Ahhoz, hogy megértsük a DUAL algoritmus működését, amikor új legjobb útvonalat keres, ha nincs tartalék útvonal, egy kapcsolati hibát kell szimulálnunk az R1 és az R3 között.

A hiba szimulációjának megkezdése előtt engedélyezni kell az R1-en a DUAL algoritmus hibakeresését a **debug eigrp fsm** paranccsal a 2. ábrán látható módon. A kapcsolati hiba szimulálását az R1 Serial 0/0/1 interfészén kiadott **shutdown** paranccsal végezhetjük el.

Ha a legjobb útvonal már nem áll rendelkezésre és nincs tartalék útvonal, akkor a DUAL algoritmus aktív állapotba helyezi az útvonalat, majd EIGRP kéréseket küld további forgalomirányítók számára a hálózathoz vezető útvonalat keresve. A forgalomirányítók EIGRP válaszokat küldenek vissza, melyben értesítik a feladót arról, hogy rendelkeznek-e a kért hálózathoz tartozó útvonallal. Amennyiben a válaszok egyike sem tartalmaz útvonalat ehhez a hálózathoz, akkor a kérés feladója nem rendelkezik útvonallal hozzá.

A 2. ábrán lévő hibakeresési kimenet alapján látható, hogy a 192.168.1.0/24 hálózat aktív állapotba került, és megtörtént az EIGRP kérések kiküldése a szomszédokhoz. Az R2 válasza tartalmaz egy útvonalat a hálózathoz, ami az új legjobb útvonal lesz és belekerül az irányítótáblába is.

Ha az EIGRP kérések feladója olyan EIGRP válaszokat kap, amelyek tartalmaznak a kért hálózathoz vezető útvonalat, akkor az adott útvonal legjobb útvonallá válik és belekerül az irányítótáblába. Ez a folyamat tovább tart annál, mint amikor a DUAL algoritmus rendelkezik tartalék útvonallal a topológiatáblában és képes gyorsan hozzáadni az új útvonalat az irányítótáblához. Figyeljük meg a 3. ábrán, hogy az R1 új útvonallal rendelkezik a 192.168.1.0/24 hálózathoz. Az új EIGRP legjobb útvonal az R2 forgalomirányító.

A 4. ábrán látható, hogy az R1 topológiatáblájában most már az R2 a legjobb útvonal és új tartalék útvonalak nincsenek. Ha az R1 és az R3 közötti kapcsolat újra elérhetővé válik, akkor megint az R3 lesz a legjobb útvonal. Az R2 azonban továbbra sem lehet tartalék útvonal, mivel nem teljesíti az alkalmassági feltételt.

# EIGRP IPv6 használatával

Az IPv4-es megfelelőjéhez hasonlóan, az EIGRP IPv6-os változata is forgalomirányítási információkat cserél, hogy távoli hálózatok előtagjaival hirdesse az irányítótáblát. Az EIGRP IPv6-os változata a Cisco IOS 12.4(6)T verziótól kezdődően érhető el.

**MEGJEGYZÉS:**IPv6 esetén a hálózati címet előtagnak (prefix), az alhálózati maszkot pedig előtag hossznak nevezik.

Az EIGRP IPv4-es változata az IPv4 hálózati réteg fölött fut, IPv4-es partnerekkel kommunikál és csak IPv4 útvonalakat hirdet. Az IPv6-os változat ugyanazokkal a funkciókkal rendelkezik, mint az IPv4-es verzió, de már az IPv6 hálózati rétegre épül, IPv6-os EIGRP partnerekkel kommunikál és IPv6 útvonalakat hirdet.

Az EIGRP IPv6-os változata szintén a DUAL algoritmust használja a hurokmentes és tartalék útvonalak számításának motorjaként az irányítási tartományban.

Ahogy az összes IPv6 irányító protokoll, az EIGRP IPv6-os változata is rendelkezik az IPv4-es megfelelőjétől független folyamatokkal. A folyamatok és a műveletek alapvetően megegyeznek az IPv4 forgalomirányító protokolléval, viszont attól függetlenül futnak. Az EIGRP IPv4-es és IPv6-os változatának is van külön szomszédtáblája, topológiatáblája és IP-irányítótáblája, ahogy azt az ábra is szemlélteti. Az EIGRP IPv6-os változata egy külön protokollfüggő modul (protocol-dependent modul, PDM).

Az EIGRP IPv6-os változatának konfigurációs és ellenőrző parancsai nagyon hasonlítanak az IPv4 esetén használtakhoz. Ezekről a parancsokról a fejezet későbbi részében lesz szó.

# Az EIGRP IPv4-es és IPv6-os változatának összehasonlítása

A következő összehasonlítás az EIGRP IPv4-es és IPv6-os változatának főbb jellemzőit veti össze:

* **Hirdetett útvonalak** - Az IPv4-es változat IPv4-hálózatokat, míg az IPv6-os IPv6-előtagokat hirdet.
* **Távolságvektor** - Az IPv4-es és IPv6-os változatok mindegyike fejlett távolságvektor alapú irányító protokoll. Mindkét protokoll azonos adminisztratív távolság értéket használ.
* **A konvergencia módja** - Az IPv4-es és az IPv6-os változat is a DUAL algoritmust használja. Mindkét protokoll ugyanazokat a DUAL módszereket és folyamatokat használja, ideértve például a legjobb útvonalat, a tartalék útvonalat (FS), a legkisebb távolságot (FD) és a hirdetett távolságot (RD) is.
* **Mérték** - Az EIGRP az IPv4 és az IPv6 esetében is a sávszélességet, a késleltetést, a megbízhatóságot és a terhelést használja az összetett mérték meghatározásához. Mindkét protokoll ugyanazt az összetett mértéket használja és alapértelmezetten csak a sávszélességet és a késleltetést veszi figyelembe.
* **Szállítási protokoll** - A megbízható szállítási protokoll (RTP) felelős az EIGRP IPv4-es és IPv6-os változatánál is a szomszédok közötti megbízható csomagtovábbításért.
* **Útvonalfrissítő üzenetek** - Mindkét változat részleges frissítéseket küld ki, ha egy cél állapota megváltozik. Mindegyik protokoll esetén a részleges és korlátozott kifejezéseket használjuk, amikor az útvonalfrissítésekre hivatkozunk.
* **A szomszédok felderítési mechanizmusa** - Az EIGRP IPv4-es és IPv6-os változata egyaránt a Hello mechanizmust használja a szomszédos forgalomirányítók felderítésére és a szomszédsági viszonyok kialakítására.
* **Forrás- és célcímek** - Az EIGRP IPv4-es változata a 224.0.0.10 csoportcímre küldi a csomagokat. Ezen üzenetek a kimenő interfész IPv4-címét használják forráscímként. IPv6 esetén az EIGRP az FF02::A csoportcímre küldi az üzeneteit. Ezen üzenetek forráscíme a kimenő interfész IPv6 link-local címével egyezik meg.
* **Hitelesítés** - IPv4 és IPv6 esetén is az MD5 (Message Digest 5) hitelesítést használja az EIGRP.
* **Forgalomirányító azonosító (Router ID)** - Az EIGRP IPv4-es és IPv6-os változata is egy 32-bites számot használ forgalomirányító azonosítóként. A 32-bites forgalomirányító azonosítót pontozott decimális jelöléssel ábrázoljuk és általában egy IPv4-címet értünk alatta. Ha az EIGRP-t futtató IPv6-os forgalomirányítón nincs IPv4-cím beállítva, akkor az **eigrp router-id** paranccsal kell beállítani a 32-bites forgalomirányító azonosítót. Az EIGRP forgalomirányító azonosító meghatározásának folyamata IPv4 és IPv6 esetén megegyezik.

**IPv6 link-local címek**

Dinamikus forgalomirányító protokollt, például EIGRP-t futtató forgalomirányítók a közös alhálózaton vagy összeköttetésen lévő szomszédjaikkal cserélnek üzeneteket. A forgalomirányítók ezért csak a hozzájuk közvetlenül kapcsolódó forgalomirányítókkal váltanak (küldenek és fogadnak) üzeneteket. Ezen üzenetek forráscíme mindig a küldő forgalomirányító IP-címe.

Az IPv6 link-local címek ideálisak erre a célra. Az IPv6 link-local címek lehetővé teszik, hogy az eszköz kommunikáljon a vele közös kapcsolaton lévő más IPv6-os eszközökkel, de csak ezen az egy kapcsolaton (alhálózaton) lévőkkel. Azok a csomagok, amelyeknek a forrása vagy célja link-local cím nem továbbíthatók a kiinduló kapcsolaton kívülre.

Az IPv6-os EIGRP üzenetek a következő címek használatával kerülnek elküldésre:

* **Forrás IPv6-cím** – A kimenő interfész IPv6 link-local címe.
* **Cél IPv6-cím** - Ha a csomagot egy csoportcímre kell küldeni, akkor a minden EIGRP forgalomirányítót tartalmazó FF02::A IPv6 csoportcímre küldik a link-local hatókörön belül. Ha a csomag egyedi címzéses üzenetként is továbbítható, akkor a szomszédos forgalomirányító link-local címére küldik.

**MEGJEGYZÉS:**Az IPv6 link-local címek az FE80::/10 tartományban vannak. A /10 azt jelenti, hogy az első 10 bit 1111 1110 10xx xxxx értékű, ami azt eredményezi, hogy az első hextet az 1111 1110 10**00 0000** (FE80) és az 1111 1110 10**11 1111** (FEBF) közé esik.

**IPv6 link-local címek**

Dinamikus forgalomirányító protokollt, például EIGRP-t futtató forgalomirányítók a közös alhálózaton vagy összeköttetésen lévő szomszédjaikkal cserélnek üzeneteket. A forgalomirányítók ezért csak a hozzájuk közvetlenül kapcsolódó forgalomirányítókkal váltanak (küldenek és fogadnak) üzeneteket. Ezen üzenetek forráscíme mindig a küldő forgalomirányító IP-címe.

Az IPv6 link-local címek ideálisak erre a célra. Az IPv6 link-local címek lehetővé teszik, hogy az eszköz kommunikáljon a vele közös kapcsolaton lévő más IPv6-os eszközökkel, de csak ezen az egy kapcsolaton (alhálózaton) lévőkkel. Azok a csomagok, amelyeknek a forrása vagy célja link-local cím nem továbbíthatók a kiinduló kapcsolaton kívülre.

Az IPv6-os EIGRP üzenetek a következő címek használatával kerülnek elküldésre:

* **Forrás IPv6-cím** – A kimenő interfész IPv6 link-local címe.
* **Cél IPv6-cím** - Ha a csomagot egy csoportcímre kell küldeni, akkor a minden EIGRP forgalomirányítót tartalmazó FF02::A IPv6 csoportcímre küldik a link-local hatókörön belül. Ha a csomag egyedi címzéses üzenetként is továbbítható, akkor a szomszédos forgalomirányító link-local címére küldik.

**MEGJEGYZÉS:**Az IPv6 link-local címek az FE80::/10 tartományban vannak. A /10 azt jelenti, hogy az első 10 bit 1111 1110 10xx xxxx értékű, ami azt eredményezi, hogy az első hextet az 1111 1110 10**00 0000** (FE80) és az 1111 1110 10**11 1111** (FEBF) közé esik.

# IPv6 link-local címek

Dinamikus forgalomirányító protokollt, például EIGRP-t futtató forgalomirányítók a közös alhálózaton vagy összeköttetésen lévő szomszédjaikkal cserélnek üzeneteket. A forgalomirányítók ezért csak a hozzájuk közvetlenül kapcsolódó forgalomirányítókkal váltanak (küldenek és fogadnak) üzeneteket. Ezen üzenetek forráscíme mindig a küldő forgalomirányító IP-címe.

Az IPv6 link-local címek ideálisak erre a célra. Az IPv6 link-local címek lehetővé teszik, hogy az eszköz kommunikáljon a vele közös kapcsolaton lévő más IPv6-os eszközökkel, de csak ezen az egy kapcsolaton (alhálózaton) lévőkkel. Azok a csomagok, amelyeknek a forrása vagy célja link-local cím nem továbbíthatók a kiinduló kapcsolaton kívülre.

Az IPv6-os EIGRP üzenetek a következő címek használatával kerülnek elküldésre:

* **Forrás IPv6-cím** – A kimenő interfész IPv6 link-local címe.
* **Cél IPv6-cím** - Ha a csomagot egy csoportcímre kell küldeni, akkor a minden EIGRP forgalomirányítót tartalmazó FF02::A IPv6 csoportcímre küldik a link-local hatókörön belül. Ha a csomag egyedi címzéses üzenetként is továbbítható, akkor a szomszédos forgalomirányító link-local címére küldik.

**MEGJEGYZÉS:**Az IPv6 link-local címek az FE80::/10 tartományban vannak. A /10 azt jelenti, hogy az első 10 bit 1111 1110 10xx xxxx értékű, ami azt eredményezi, hogy az első hextet az 1111 1110 10**00 0000** (FE80) és az 1111 1110 10**11 1111** (FEBF) közé esik.

# EIGRPv6 hálózati topológia

Az 1. ábrán látható topológiát fogjuk használni az EIGRP IPv6-os változatának konfigurálásához. Ha a hálózat kettős protokollkészlettel működik, vagyis minden eszköz IPv4 és IPv6-címeket is használ, akkor az EIGRP IPv4-es és IPv6-os változata is konfigurálható a forgalomirányítók mindegyikén. Ebben a fejezetben a hangsúly kizárólag az EIGRP IPv6-os változatán lesz.

A forgalomirányítókon csak IPv6 globális egyedi címek vannak konfigurálva.

A 2., 3. és a 4. ábrákon az egyes forgalomirányítók kezdeti interfész beállításai láthatók. Figyeljük meg az interfész sávszélességek értékeit a korábbi IPv4-es EIGRP konfigurációkban. Mivel az EIGRP IPv4-es és IPv6-os változata is ugyanazt a mértéket használja, ezért a sávszélesség értékének megváltoztatása mindkét protokollra hatással van.

# IPv6 link-local címek beállítása

Ha egy interfészhez IPv6 globális egyedi címet rendelünk, akkor a link-local címek automatikusan létrejönnek. Az interfésznek nem szükséges globális egyedi címmel rendelkeznie, az IPv6 link-local címek viszont elengedhetetlenül fontosak.

Hacsak azt kézzel be nem állítjuk, a Cisco forgalomirányítók az FE80::/10 előtaggal és az EUI-64 folyamattal hoznak létre link-local címeket, ahogy ez az 1. ábrán is látható. Az EUI-64 a 48 bites Ethernet MAC-cím közepére FFFE-t illesztve és a hetedik bit invertálásával állítja elő a cím interfész részét. A soros interfészekhez a Cisco az Ethernet interfészek MAC-címét használja. Több soros interfésszel rendelkező forgalomirányító valamennyi interfészéhez akár ugyanazt az IPv6 link-local címet is rendelhetjük, mivel a link-local címeknek csak a kapcsolaton belül kell egyedinek lenniük.

Az EUI-64 formátumú link-local címek, vagy egyes esetekben a véletlenszerű interfész-azonosítók használata megnehezítik a címek felismerését és megjegyzését. Mivel az IPv6 forgalomirányító protokollok IPv6 link-local címeket használnak az egyedi címzéshez és az irányítótábla következő ugrás értékeiként, ezért általános gyakorlat, hogy azokat könnyen felismerhetőnek és megjegyezhetőnek választjuk. A link-local címek manuális konfigurálása lehetővé teszi, hogy olyan címet alkossunk, ami felismerhető és könnyen megjegyezhető.

A link-local címeket ugyanazzal az interfész konfigurációs módbeli paranccsal állíthatjuk be manuálisan, mint amivel IPv6 globális egyedi címet adunk meg, csak más paraméterekkel:

Router(config-if)# **ipv6 address** *link-local-cím* **link-local**

A link-local címek előtagja az FE80 és FEBF tartományba esik. Ha egy cím ezzel a hextettel (16 bites bitcsoport) kezdődik, akkor a címet a **link-local** kulcsszónak kell követnie.

A 2. ábra a link-local cím konfigurációját mutatja az **ipv6 address** interfész konfigurációs paranccsal. Azért használjuk az FE80::1 link-local címet, hogy könnyebben megjegyezhessük, hogy az R1 forgalomirányítóhoz tartozik. Ugyanezt az IPv6 link-local címet beállíthatjuk az R1 összes interfészén is. Sőt akár minden kapcsolatra is beállítható lenne ugyanez az FE80::1 cím, mert csak a kapcsolaton belül kell egyedinek lennie.

Az R1-hez hasonlóan az R2 minden interfészére beállítjuk az FE80::2 IPv6 link-local címet, ez látható a 3. ábrán.

A 4. ábra parancsszimulátorában állítsuk be az FE80::3 link-local címet az R3 minden interfészén.

Az 5. ábrán látható, hogy a **show ipv6 interface brief** parancs használható az interfészek IPv6 link-local és globális egyedi címeinek ellenőrzésére.

# Az EIGRP IPv6-os forgalomirányítási folyamatának beállítása

Az **ipv6 unicast-routing** globális konfigurációs módbeli parancs engedélyezi az IPv6 forgalomirányítást a forgalomirányítón. A parancs kiadására bármely IPv6 irányító protokoll konfigurálása előtt szükség van. Az interfészek IPv6-címeinek konfigurálása ezen parancs kiadása nélkül is megoldható, ahhoz viszont szükséges, hogy az IPv6 forgalomirányítás engedélyezve legyen a forgalomirányítón.

**EIGRP IPv6-hálózatban**

Az EIGRP IPv6-os változata esetén a forgalomirányító módba történő belépéshez a következő globális konfigurációs módbeli parancs használható:

Router(config)# **ipv6** **router eigrp** *autonóm\_rendszer\_azonosító*

Az EIGRP IPv4-es változatához hasonlóan az *autonóm\_rendszer\_azonosító* értékének az irányítási tartomány minden forgalomirányítóján egyeznie kell. Az 1. ábrán látható, hogy IPv6 esetén az EIGRP forgalomirányítási folyamata nem konfigurálható, amíg az IPv6 forgalomirányítás engedélyezésre nem kerül az **ipv6 unicast-routing** globális konfigurációs módbeli paranccsal.

**Forgalomirányító azonosító (Router ID)**

Ahogy a 2. ábrán látható, a forgalomirányító azonosítót az **eigrp router-id** parancs használatával lehet beállítani. Az EIGRP IPv6-os változata egy 32 bites számot használ forgalomirányító azonosítóként. Ahhoz, hogy megkapja ezt az értéket, ugyanazt a folyamatot használja, mint IPv4 esetében. Az **eigrp router-id** parancs elsőbbséget élvez bármely loopback vagy fizikai interfész IPv4-címével szemben. Ha egy EIGRP-t futtató IPv6-os forgalomirányító nem rendelkezik egyetlen olyan aktív interfésszel sem, amely IPv4-címmel konfigurált, akkor az **eigrp router-id** parancsra van szükség.

Az azonosítónak egy egyedi 32 bites számnak kell lennie az EIGRP irányítási tartományon belül, különben irányítási ellentmondások fordulhatnak elő.

**MEGJEGYZÉS:**Az **eigrp router-id** parancs az EIGRP forgalomirányító azonosító beállítására használható. Az IOS egyes verziói elfogadják a **router-id**parancsot az **eigrp**előtag nélkül is. Az aktív konfigurációban viszont megjelenik az **eigrp router-id** sor, függetlenül attól, hogy melyik utasítást használtuk.

Alapértelmezés szerint az EIGRP irányítási folyamata IPv6 esetén lekapcsolt állapotban van. A **no shutdown** paranccsal lehet az EIGRP IPv6-os irányítási folyamatot aktiválni. (Lásd 3. ábra) A parancs használata az EIGRP IPv4-es változatánál nem szükséges. Annak ellenére, hogy az EIGRP IPv6-os változata engedélyezett, a szomszédsági viszonyok kialakítása, valamint az irányítási frissítések küldése és fogadása mindaddig nem lehetséges, amíg az EIGRP-t nem aktiváljuk a megfelelő interfészeken.

Mind a **no shutdown** parancs, mind pedig a forgalomirányító azonosító megadása szükséges ahhoz, hogy egy forgalomirányító szomszédsági viszonyokat építsen ki.

A 4. ábra az EIGRP IPv6-os változatának teljes konfigurációját mutatja az R2 forgalomirányítón.

Használjuk az 5. ábrán lévő parancsszimulátort az EIGRP IPv6-os irányítási folyamatának beállításához az R3 forgalomirányítón.

**Az ipv6 eigrp interfész konfigurációs parancs**

Az EIGRP IPv6-os változata eltérő módszert használ az EIGRP interfészek engedélyezésére. Ahelyett, hogy a **network** forgalomirányító konfigurációs módbeli paranccsal adnánk meg a megfelelő interfész címeket, közvetlenül az interfészen konfiguráljuk az EIGRP IPv6-os változatát.

A következő interfész konfigurációs módbeli paranccsal engedélyezhető az EIGRP IPv6-os változata egy interfészen:

Router(config-if)# **ipv6 eigrp** *autonóm\_rendszer\_azonosító*

Az *autonóm\_rendszer\_azonosító* értékének meg kell egyeznie az EIGRP irányítási folyamat engedélyezésénél használt autonóm rendszer azonosítóval. Hasonlóan az EIGRP IPv4-es változatánál használt **network** parancshoz, a **ipv6 eigrp interface** parancsra is igaz, hogy:

* Lehetővé teszi az interfész számára, hogy szomszédsági viszonyt alakítson ki, valamint EIGRP frissítéseket küldjön és fogadjon.
* Hozzárendeli az EIGRP IPv6-os irányítási frissítésekhez az interfészhez tartozó előtagot (hálózatot).

Az 1. ábrán az EIGRP IPv6-os változatának engedélyezése látható az R1 és az R2 forgalomirányítók interfészein. Figyeljük meg, hogy az R2 Serial 0/0/0 interfészén a következő üzenet jelenik meg:

%DUAL-5-NBRCHANGE: EIGRP-IPv6 2: Neighbor FE80::1 (Serial0/0/0) is up: new adjacency

Ez az üzenet jelzi, hogy az R2 EIGRP IPv6-os szomszédsági viszonyt alakított ki a szomszédjával az FE80::1 link-local címen. Mivel mindhárom forgalomirányítón a link-local címek beállítása manuálisan történt, így könnyű meghatározni, hogy ez a szomszédság az R1 forgalomirányítóval áll fenn (FE80::1).

A 2. ábra parancsszimulátorában engedélyezzük az EIGRP IPv6-os változatát az R3 interfészein.

**Passzív interfészek az EIGRP IPv6-os változata esetén**

Ugyanaz a **passive-interface** parancs, amit korábban az IPv4 esetén használtunk, alkalmazható egy EIGRP IPv6-os interfészt passzív állapotúvá tételére is. Ahogy a 3. ábrán is látható, a **show ipv6 protocols** paranccsal ellenőrizhető a konfiguráció.

**Az EIGRP IPv6-os változatának ellenőrzése: a szomszédság vizsgálata**

Az EIGRP IPv4-es változatához hasonlóan, IPv6 esetén is szomszédsági viszonyt kell létesíteniük egymással a forgalomirányítóknak az EIGRP frissítések küldése vagy fogadása előtt. (Lásd 1. ábra)

Használjuk a **show ipv6 eigrp neighbors** parancsot a szomszédtábla megtekintéséhez, valamint annak ellenőrzésére, hogy az EIGRP IPv6-os változata kiépítette-e a szomszédsági kapcsolatait a szomszédaival. A 2. ábrán lévő parancskimeneten a szomszéd IPv6 link-local címe, valamint az az interfész látható, amelyet a forgalomirányító az EIGRP szomszéd eléréséhez használ. A link-local címek ésszerű használatával egyszerűvé válik a szomszédok felismerése: az R2 forgalomirányító az FE80::2 címen, és az R3 forgalomirányító az FE80::3 címen.

A **show ipv6 eigrp neighbors** parancs kimenete a következőket tartalmazza:

* **H oszlop** - Kilistázza a szomszédokat a megismerésük sorrendjében.
* **Address** - A szomszéd IPv6 link-local címe.
* **Interface** - A helyi interfész, amelyen a Hello csomag érkezett.
* **Hold** - A megtartási idő aktuális értéke. Hello csomag érkezésekor az interfészen ez a számláló visszaáll a legmagasabb értékre, és onnan kezd el csökkenni. Amint eléri a nullát, a szomszédot a forgalomirányító elérhetetlennek minősíti.
* **Uptime** - A szomszédsági táblában létrejött bejegyzés óta eltelt idő.
* **SRTT** és **RTO** - Az RTP által a megbízható EIGRP csomagok kezelésére használt mezők.
* **Queue Count** - Az értékének mindig nullának kell lennie. Ha nagyobb mint nulla, akkor EIGRP csomagok vannak a várakozási sorban.
* **Sorszám** - Frissítő, lekérdező és válasz csomagok nyomon követésére használható.

A **show ipv6 eigrp neighbors** parancs az EIGRP IPv6-os változatának ellenőrzésénél és hibaelhárításánál nyújt segítséget. Amennyiben egy feltételezett szomszéd nem jelenik meg a listában, meg kell bizonyosodnunk róla, hogy a kapcsolat mindkét vége felkapcsolt állapotban van. Ehhez használjuk a **show ipv6 interface brief** parancsot. A szomszédsági viszonyok létrejöttének ugyanazok a feltételei az EIGRP IPv6-os és IPv4-es változata esetén is. Ha a kapcsolat mindkét végpontja felkapcsolt interfészekkel rendelkezik, ellenőrizzük, hogy:

* Mindkét forgalomirányító azonos autonóm rendszer azonosítóval lett konfigurálva?
* Az interfész a megfelelő autonóm rendszer azonosítóval lett engedélyezve az EIGRP IPv6-os változata számára?

# Az EIGRP IPv6-os változatának ellenőrzése: show ip protocols parancs

A **show ipv6 protocols** parancs a forgalomirányítón konfigurált IPv6 irányító protokoll folyamatainak aktuális állapotáról jelenít meg paramétereket és egyéb információkat. A **show ipv6 protocols** parancs az adott IPv6 forgalomirányító protokolltól függően eltérő kimenetet adhat.

Az ábra parancskimenetén számos korábban már tárgyalt EIGRP paraméter van megjelölve, többek között:

1. Az R1-en az 1-es autonóm rendszer azonosítóval konfigurált EIGRP IPv6-os változata az aktív dinamikus forgalomirányító protokoll.

2. Az EIGRP összetett mérték kiszámításához használt *k* értékek. Alapértelmezés szerint K1 és K3 értéke 1, K2, K4 és K5 értéke pedig 0.

3. Az R1 EIGRPv6 forgalomirányító azonosítója 1.0.0.0.

4. Az EIGRP IPv4-es változatához hasonlóan IPv6 esetén is az EIGRP-hez tartozó belső adminisztratív távolság 90, a külső pedig 170 (alapértelmezett értékek).

5. Az EIGRP IPv6-os változata számára engedélyezett interfészek.

A **show ipv6 protocols** parancs forgalomirányítási műveletek hibakeresésében nyújt segítséget. Az interfészek rész azt mutatja, hogy melyek az EIGRP IPv6-os változata számára engedélyezett interfészek. A parancs használatával ellenőrizhetjük, hogy az EIGRP a megfelelő interfészeken a megfelelő autonóm rendszer azonosítókkal lettek-e engedélyezve.

# Az EIGRP IPv6-os változatának ellenőrzése: Az IPv6 irányítótábla vizsgálata

Minden irányító protokoll esetén az cél, hogy az IP-irányítótábla tartalmazza a távoli hálózatokhoz tartozó bejegyzéseket, valamint ezen hálózatok eléréséhez használható legjobb útvonalakat. Ahogy az IPv4-nél, úgy IPv6 esetén is fontos az irányítótábla vizsgálata, valamint annak ellenőrzése, hogy a megfelelő útvonalak kerültek-e bele.

Az IPv6-irányítótábla vizsgálata elvégezhető a **show ipv6 route** paranccsal. Az IPv4-hez hasonlóan, az EIGRP IPv6-os változatának útvonalait is D betű jelöli az irányítótáblában.

Az 1. ábra azt mutatja, hogy az R1 irányítótáblája három EIGRP útvonalat tartalmaz távoli IPv6-hálózatok felé:

* A 2001:DB8:CAFE:2::/64 hálózatot az FE80::3 következő ugrás címmel (R3) és a Serial 0/0/1 kimenő interfésszel.
* A 2001:DB8:CAFE:3::/64 hálózatot az FE80::3 következő ugrás címmel (R3) és a Serial 0/0/1 kimenő interfésszel.
* A 2001:DB8:CAFE:A002::/64 hálózatot az FE80::3 következő ugrás címmel (R3) és a Serial 0/0/1 kimenő interfésszel.

Mindhárom útvonal az R3-at használja következő ugrás forgalomirányítóként (legjobb útvonal). Figyeljük meg, hogy az irányítótábla a link-local címet használja következő ugrás címként. Mivel minden egyes forgalomirányító interfész egyedi és megkülönböztethető link-local címmel rendelkezik, így könnyen azonosítható, hogy az FE80::3 című következő ugrás forgalomirányító az R3.

A 2. ábrán az R2 IPv6-irányítótáblája látható.

A 3. ábra az R3 irányítótábláját mutatja. Figyeljük meg, hogy az R3 két egyenlő költségű útvonallal rendelkezik a 2001:DB8:CAFE:A001::/64 hálózat felé. Az egyik útvonal az R1-en az FE80::1, a másik útvonal pedig az R2-n az FE80::2 címen keresztül halad.

# Packet Tracer - Az EIGRP alapszintű konfigurálása IPv6-hálózatban

**Háttér / Esetleírás**

Ebben a feladatban EIGRP alapú forgalomirányítást fogunk konfigurálni egy IPv6-hálózatban. Ezen kívül elvégezzük a forgalomirányító azonosító hozzárendelést, a passzív interfészek beállítását és a hálózati konvergencia ellenőrzését, valamint forgalomirányítási információkat jelenítünk meg **show** parancsok segítségével.

* Az EIGRP IPv6-os változata ugyanolyan általános érvényű működéssel és funkciókkal rendelkezik, mint az IPv4-es változat. Néhány jelentősebb különbség azért van köztük:
* Az EIGRP IPv6-os változata közvetlenül a forgalomirányító interfészeken van konfigurálva.
* Az EIGRP IPv6-os változata esetén minden forgalomirányítón szükség van forgalomirányító azonosítóra, különben az irányítási folyamat nem indul el.
* Az EIGRP IPv6-os forgalomirányítási folyamata egy kikapcsolási funkciót használ.

[Packet Tracer - Configuring Basic EIGRP with IPv6 Instructions](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/7.4.3.4%20Packet%20Tracer%20-%20Configuring%20Basic%20EIGRP%20with%20IPv6%20Routing%20Instructions.pdf)

[Packet Tracer - Configuring Basic EIGRP with IPv6 - PKA](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/7.4.3.4%20Packet%20Tracer%20-%20Configuring%20Basic%20EIGRP%20with%20IPv6%20Routing.pka)

# Összefoglalás

Az EIGRP (továbbfejlesztett belső átjáró irányító protokoll, Enhanced Interior Gateway Routing Protocol) egy osztály nélküli, távolságvektor alapú irányító protokoll. Az EIGRP egy másik, már elavultnak számító, Cisco fejlesztésű irányító protokoll, az IGRP (Interior Gateway Routing Protocol) továbbfejlesztése. Az 1992-es megjelenésekor az EIGRP a Cisco saját protokolljának számított, és kizárólag Cisco eszközökön volt elérhető. A Cisco 2013-ban az EIGRP alap funkcióit nyílt szabványúvá tette az IETF számára.

Az EIGRP a D betűt használja a forrás megjelöléseként az irányítótáblában, amely a DUAL algoritmusra utal. Az EIGRP belső útvonalaihoz az alapértelmezett 90-es adminisztratív távolság, a külső forrásokból importált útvonalakhoz, például az alapértelmezett útvonalakhoz pedig a 170-es érték van hozzárendelve.

Az EIGRP egy fejlett távolságvektor alapú irányító protokoll, amelyben olyan funkciók is megtalálhatók, amelyek más távolságvektor alapú protokolloknál, például a RIP-nél nincsenek. Ezek a funkciók a következők: szétszóró frissítő algoritmus (Diffusing Update Algorithm, DUAL), szomszédsági viszonyok kialakítása, megbízható szállítási protokoll (Reliable Transport Protocol, RTP), részleges és korlátozott frissítések, valamint egyenlő és nem egyenlő költségű terhelés elosztás.

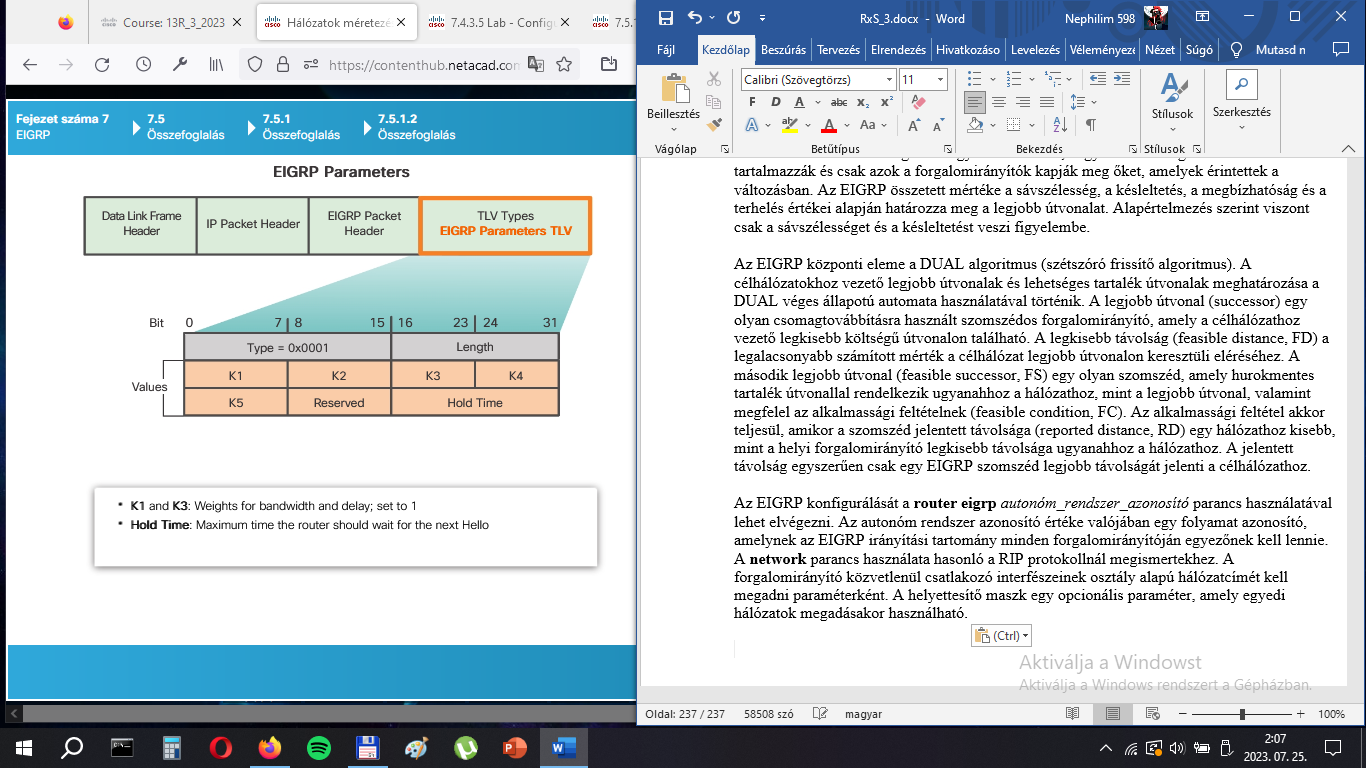
Az EIGRP protokollfüggő modulokat használ (Protocol Dependent Modules, PDM) azért, hogy különböző 3. rétegbeli protokollok, például az IPv4 és az IPv6 támogatására is képes legyen. Az EIGRP csomagok kézbesítését a szállítási rétegben található megbízható szállítási protokoll (Reliable Transport Protocol, RTP) végzi. Az EIGRP megbízható kézbesítést használ a frissítő, lekérdező és válaszcsomagok estén, és nem megbízható kézbesítést a Hello és nyugta üzeneteknél. Megbízható RTP esetén kötelező az EIGRP nyugták használata az üzenetváltás során.

Mielőtt megkezdődhetne az EIGRP frissítő csomagok küldése, a forgalomirányítónak fel kell fedeznie saját szomszédjait. Ezt EIGRP Hello csomagok segítségével valósítja meg. Két forgalomirányító közti szomszédsági viszony kialakításához a rajtuk beállított Hello és megtartási idők értékének nem kell megegyezni. A **show ip eigrp neighbors** parancs használható a szomszédtábla megtekintéséhez, valamint az EIGRP létrejött szomszédsági kapcsolatainak ellenőrzésre.

A RIP-pel ellentétben az EIGRP nem küld rendszeres időközönként útvonalfrissítéseket. Az elküldött frissítések részlegesek vagy korlátozottak, vagyis csak a megváltozott útvonalakat tartalmazzák és csak azok a forgalomirányítók kapják meg őket, amelyek érintettek a változásban. Az EIGRP összetett mértéke a sávszélesség, a késleltetés, a megbízhatóság és a terhelés értékei alapján határozza meg a legjobb útvonalat. Alapértelmezés szerint viszont csak a sávszélességet és a késleltetést veszi figyelembe.

Az EIGRP központi eleme a DUAL algoritmus (szétszóró frissítő algoritmus). A célhálózatokhoz vezető legjobb útvonalak és lehetséges tartalék útvonalak meghatározása a DUAL véges állapotú automata használatával történik. A legjobb útvonal (successor) egy olyan csomagtovábbításra használt szomszédos forgalomirányító, amely a célhálózathoz vezető legkisebb költségű útvonalon található. A legkisebb távolság (feasible distance, FD) a legalacsonyabb számított mérték a célhálózat legjobb útvonalon keresztüli eléréséhez. A második legjobb útvonal (feasible successor, FS) egy olyan szomszéd, amely hurokmentes tartalék útvonallal rendelkezik ugyanahhoz a hálózathoz, mint a legjobb útvonal, valamint megfelel az alkalmassági feltételnek (feasible condition, FC). Az alkalmassági feltétel akkor teljesül, amikor a szomszéd jelentett távolsága (reported distance, RD) egy hálózathoz kisebb, mint a helyi forgalomirányító legkisebb távolsága ugyanahhoz a hálózathoz. A jelentett távolság egyszerűen csak egy EIGRP szomszéd legjobb távolságát jelenti a célhálózathoz.

Az EIGRP konfigurálását a **router eigrp** *autonóm\_rendszer\_azonosító* parancs használatával lehet elvégezni. Az autonóm rendszer azonosító értéke valójában egy folyamat azonosító, amelynek az EIGRP irányítási tartomány minden forgalomirányítóján egyezőnek kell lennie. A **network** parancs használata hasonló a RIP protokollnál megismertekhez. A forgalomirányító közvetlenül csatlakozó interfészeinek osztály alapú hálózatcímét kell megadni paraméterként. A helyettesítő maszk egy opcionális paraméter, amely egyedi hálózatok megadásakor használható.



* [Fejezet száma 8 EIGRP haladó szintű konfiguráció és hibaelhárítás](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#8)
* [8.0 EIGRP haladó szintű konfiguráció és hibaelhárítás](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#8.0)
* [8.0.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#8.0.1)
* [8.0.1.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#8.0.1.1)

# Bevezetés

Az EIGRP egy sokoldalú irányító protokoll, ami többféle módon finomhangolható. A két legfontosabb ilyen képessége, hogy képes útvonalak összevonására és terheléselosztásra. Egyéb képességei közé tartozik, hogy hirdetni tud egy alapértelmezett útvonalat, finomhangolni az időzítőit, és hitelesíteni az EIGRP szomszédokat a biztonság növelése érdekében.

Ez a fejezet ezeket a további jellemzőket és a konfigurációs módbeli parancsokat tárgyalja, amikkel ezek a funkciók megvalósíthatók az IPv4 és az IPv6 alatt is.

# Csoportos feladat - EIGRP - Vissza a jövőbe!

**Feladat -** **EIGRP - Vissza a jövőbe!**

Ez a fejezet bemutatja, hogyan lehet karbantartani és befolyásolni az EIGRP hálózatokat úgy, hogy azt tegyék, amit szeretnénk. Az ebben a fejezetben szereplő EIGRP fogalmak a következők:

* Automatikus útvonal-összevonás
* Terheléselosztás
* Alapértelmezett útvonalak
* Megtartási időzítők
* Hitelesítés

Egy partnerünkkel írjunk 10 EIGRP ismétlő kérdést, az előző fejezet tananyagának tartalma alapján. Három kérdésnek a fentebb felsorolt tételekre kell fókuszálnia. Ideális esetben feleletválasztós, igaz / hamis vagy "üres helyek kitöltése" típusú feladatok készüljenek! Ahogy megtervezzük a kérdéseket, jegyezzük fel a tananyag vonatkozó részeit és oldalszámait, hogy szükség esetén hivatkozni tudjunk rájuk az ellenőrzésekor.

Mentsük el a munkánkat, majd egy másik csoportnak vagy az egész osztálynak adjuk fel a kérdéseinket!

[Class Activity - EIGRP - Back to the Future](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/8.0.1.2%20EIGRP%20-%20Back%20to%20the%20Future%20Instructions.pdf)

# Hálózati topológia

Mielőtt finomhangolnánk az EIGRP funkcióit, kezdjük az alapbeállításoknál.

Az 1. ábra mutatja az ebben a fejezetben használt hálózati topológiát.

A 2., 3. és 4. ábrákon az IPv4-es interfész-beállítások és az EIGRP megvalósítása látható az R1, R2 és R3 eszközön.

A soros interfészek típusa és sávszélessége nem szükségszerűen felel meg a napjainkban leginkább elterjedt kapcsolattípusoknak. A topológiában a soros vonalak sávszélességét a forgalomirányító protokoll mértékszámításának és a legjobb út kiválasztási folyamatának bemutatása céljából választottuk.

Vegyük észre, hogy a **bandwidth** parancs a soros interfészeken arra használatos, hogy módosítsuk vele az alapértelmezett 1544 kb/s értékű sávszélességet.

Ebben a fejezetben az ISP forgalomirányítóját használjuk átjáróként az internet felé. Mindhárom forgalomirányító a Cisco IOS 15.2 verzióját futtatja.

# EIGRP automatikus útvonal-összevonás

Az EIGRP egyik leggyakoribb hangolási módszere az automatikus útvonal-összevonás engedélyezése és letiltása. Útvonal-összevonáskor a forgalomirányító az összefüggő hálózatokat csoportosítja, és egyetlen nagy hálózatként hirdeti őket. Az útvonalak összevonása amiatt szükséges, hogy követhessük a hálózatok gyors növekedési ütemét.

A határ-forgalomirányító olyan eszköz, amely a hálózat peremén foglal helyet. Ennek képesnek kell lennie arra, hogy hirdesse az összes ismert hálózatát az irányítótábláján belül, egy kapcsolódó hálózati vagy ISP forgalomirányító felé. Ez akár rendkívül nagyméretű irányítótáblákat is eredményezhet. Képzeljük el, hogy egy forgalomirányítónak 10 különböző hálózata van és hirdetnie kell mind a 10 útvonal-bejegyzést egy összekötő eszköz felé. Mi van akkor, ha ez a kapcsolódó forgalomirányító is rendelkezik 10 hálózattal, és mind a 20 útvonalat hirdetnie kell egy ISP forgalomirányító felé? Ha minden vállalati forgalomirányító követné ezt a mintát, az irányítótábla az ISP forgalomirányítón hatalmasra duzzadna.

Az útvonal-összevonás csökkenti az útvonalfrissítésekben szereplő bejegyzések, és a helyi irányítótábla-bejegyzések számát is. Javítja az útvonalfrissítések sávszélesség-kihasználását és gyorsítja az irányítótáblában való keresést.

Az irányítási hirdetmények számának és az irányítótáblák méretének korlátozása miatt az irányítóprotokollok, köztük az EIGRP is, az automatikus összevonást használják az osztály alapú határokon. Ez azt jelenti, hogy az EIGRP az alhálózatokat egyetlen A, B vagy C osztályú hálózatként ismeri fel, és csak egy bejegyzést hoz létre az irányítótáblában az összevont útvonalnak. Ennek eredményeként minden, külön alhálózatnak szánt forgalom ezen az egy útvonalon halad.

Az ábra egy példát mutat az automatikus összevonás működésére. Az R1 és R2 forgalomirányítókat IPv4-es EIGRP-vel állították be, automatikus összevonással. A R1-nek három alhálózata van az irányítótáblájában: 172.16.1.0/24, 172.16.2.0/24 és 172.16.3.0/24. Az osztály alapú hálózati címzési architektúra alapján ezek az alhálózatok mind részei a nagyobb, B osztályú 172.16.0.0/16 hálózatnak. Mivel az EIGRP-t az R1 forgalomirányítón automatikus összevonásra állították be, ezért amikor elküldi az útvonalfrissítést az R2-nek, összefogja a három /24 alhálózatot egy egységes 172.16.0.0/16 hálózatba, ami csökkenti az elküldött útvonalfrissítések és az R2 forgalomirányító IPv4-es irányítótábla-bejegyzéseinek számát.

Minden, külön alhálózatnak szánt forgalom ezen az egy útvonalon halad. Az R2 nem tartja nyilván az egyes alhálózatokhoz vezető útvonalakat és nem tanul meg alhálózati információkat. Nagyvállalati hálózatokban az összevont útvonal nem feltétlenül a legjobb választás az egyes alhálózatokba címzett csomagok számára. Az egyetlen megoldás, mellyel a forgalomirányítók megtalálhatják az alhálózatokhoz vezető legjobb útvonalakat, hogyha alhálózati információt is küldenek a szomszédok. Ebben a helyzetben az automatikus összevonást le kell tiltani. Az automatikus összevonás letiltásakor a frissítések az alhálózati információkat is tartalmazzák.

# Az EIGRP automatikus útvonal-összevonásának beállítása

Az EIGRP IPv4-es verziójánál az automatikus összevonás alapértelmezés szerint le van tiltva a Cisco IOS 15.0(1)M és 12.2(33) verziótól kezdődően. Ezt megelőzően az automatikus összevonás alapértelmezés szerint engedélyezett volt. Ez azt jelentette, hogy az EIGRP automatikus összevonást végzett, valahányszor a EIGRP topológia átlépte két különböző osztály-alapú főhálózat határát.

Az 1. ábrán a **show ip protocols** parancs kimenete azt jelzi, hogy az R1-en az EIGRP automatikus útvonal-összevonása le van tiltva. Ez a forgalomirányító IOS 15.2 verziót futtat; ezért az EIGRP automatikus útvonal-összevonása alapértelmezés szerint le van tiltva. A 2. ábra mutatja az R3 jelenlegi irányítótábláját. Figyeljük meg, hogy az R3 forgalomirányító IPv4-es irányítótáblája tartalmazza az összes hálózatot és alhálózatot az EIGRP irányítási területéről.

Az automatikus útvonal-összevonás engedélyezése érdekében használjuk az **auto-summary** parancsot forgalomirányító-konfigurációs üzemmódban, ahogy a 3. ábrán látható:

R1(config)# **router eigrp** *AS-azonosító*

R1(config-router)# **auto-summary**

A **no** formában kiadott parancs kikapcsolja az automatikus összevonást.

Használjuk a 4. ábrán látható parancsszimulátort, hogy engedélyezzük az automatikus útvonal-összevonást az R3 számára.

**Az automatikus összevonás ellenőrzése: show ip protocols**

Az 1. ábrán látható, hogy az EIGRP tartomány három osztály-alapú hálózatot tartalmaz.

* a 172.16.0.0/16 B-osztályú hálózatot, amely a 172.16.1.0/24, 172.16.2.0/24 és 172.16.3.0/30 alhálózatokat tartalmazza
* a 192.168.10.0/24 C-osztályú hálózatot, amely a 192.168.10.4/30 és 192.168.10.8/30 alhálózatokat tartalmazza
* és a 192.168.1.0/24 C-osztályú hálózatot, ami nincs alhálózatokra bontva

Az R1-en kiadott **show ip protocols** parancs kimenete azt jelzi, hogy az R1-en az EIGRP automatikus útvonal-összevonása engedélyezve van. A kimenet jelzi az összevont hálózatokat és a kapcsolódó interfészeket is. Figyeljük meg, hogy az R1 két hálózatot von össze saját EIGRP útvonalfrissítéseiben:

* a 192.168.10.0/24-et a GigabitEthernet 0/0 és Serial 0/0/0 interfészeken hirdeti
* a 172.16.0.0/16-ot a Serial 0/0/1 interfészen hirdeti

Az R1 tartalmazza a 192.168.10.4/30 és 192.168.10.8/30 alhálózatokat az IPv4-es irányítótáblájában.

Amint azt a 3. ábra mutatja, az R1 összevonja a 192.168.10.4/30 és 192.168.10.8/30 alhálózatokat. A 192.168.10.0/24 összevont címet továbbítja a szomszédjai felé a Serial 0/0/0 és GigabitEthernet 0/0 interfészein keresztül. Mivel az R1-nek nincs EIGRP szomszédja a GigabitEthernet 0/0 interfészén, az összevont frissítést csak az R2 kapja meg.

Amint azt a 4. ábra mutatja, az R1 szintén tárolja a 172.16.1.0/24, 172.16.2.0/24 és 172.16.3.0/30 alhálózatokat az IPv4-es irányítótáblájában. Az R3 forgalomirányító az R1-et választja ki legjobb útvonalként a 172.16.0.0/16 felé, mivel kedvezőbb mértékkel rendelkezik. Az R3 forgalomirányító R1-hez csatlakozó S0/0/0 interfészének alapértelmezett sávszélessége 1544 kb/s. Az R3 kapcsolata az R2 felé magasabb költséggel rendelkezik, mivel az R3 forgalomirányító S0/0/1 interfészén alacsonyabb a beállított sávszélesség: 1024 kb/s.

Figyeljük meg, hogy a 172.16.0.0/16 összevont frissítést az R1 nem küldi ki a GigabitEthernet 0/0 és Serial 0/0/0 interfészein. Ez azért van, mert ez a két interfész ugyanannak a 172.16.0.0/16 B-osztályú hálózatnak a tagja. A nem összevont 172.16.1.0/24 frissítést az R1 forgalomirányító kiküldi az R2-nek. Az összevont frissítéseket csak különböző osztály-alapú főhálózathoz tartozó interfészeken küldi ki.

# Az automatikus összevonás ellenőrzése: a topológiatábla

Az 1. ábrán az R1 és R2 forgalomirányítók összevont frissítést küldenek az R3-nak a 172.16.0.0/16-ról. Az R1 és R2 irányítótáblái tartalmazzák a 172.16.0.0/16 hálózat alhálózatait; ezért mindkét eszköz kiküldi az összefogott frissítést a teljes főhálózatról az R3-nak.

A 2. ábra mutatja a **show ip eigrp topology all-links** parancs kimenetét, amivel meg lehet tekinteni az R3 teljes EIGRP topológiatábláját. Ez igazolja, hogy az R3 megkapta a 172.16.0.0/16 összevont útvonalat mind az R1-től a 192.168.10.5, mind az R2-től a 192.168.10.9 címen. Az első bejegyzés a 192.168.10.5-ön keresztül a legjobb, míg a második bejegyzés a 192.168.10.9-en át a második legjobb útvonal. Az R1 a legjobb útvonal, mert az 1544 kb/s kapcsolata az R3 számára jobb EIGRP költséget jelent a 172.16.0.0/16 felé, mint az R2, amely egy lassabb, 1024 kb/s kapcsolatot használ.

Az **all-links** opció attól függetlenül megjeleníti az összes beérkezett frissítést, hogy a kérdéses útvonal a második legjobb-e (FS). Ebben az esetben az R2 második legjobbnak minősül. Az R2 azért tekinthető második legjobb útvonalnak, mert a 2816 értékű jelentett távolsága (RD) kevesebb, mint a 2170112 értékű jelenlegi költség (FD) az R1-en keresztül.

**Az automatikus összevonás ellenőrzése: az irányítótábla**

Vizsgáljuk meg az irányítótáblát, hogy ellenőrizzük, az összevont útvonal megérkezett-e.

Az 1. ábrán látható az R3 irányítótáblája az automatikus útvonal-összevonás előtt, majd az automatikus összevonás engedélyezése után, az **auto-summary** parancs kiadását követően. Figyeljük meg, hogy ha az automatikus összevonás engedélyezett, az R3 irányítótáblája csak az egyszerű B-osztályú 172.16.0.0/16 hálózati címet tartalmazza. A legjobb útvonal, más szóval a következő ugrás az R1 forgalomirányítón, vagyis a 192.168.10.5-ön keresztül vezet.

**MEGJEGYZÉS:**Az automatikus összevonás csak az IPv4-es EIGRP alatti opció. Osztály-alapú címzés nem létezik az IPv6 esetében; ezért nincs is szükség automatikus útvonal-összevonásra az IPv6-os EIGRP protokollban.

Ha bekapcsoljuk az automatikus összevonást, arra is szükség van, hogy megértsük a Null interfész működését. A 2. ábra mutatja az R1 irányítótábláját. Figyeljük meg, hogy a két kiemelt bejegyzés kimenő interfészként a Null0-t használja. Az EIGRP automatikusan beszúrt egy összevont útvonalat a Null0-hoz két osztály-alapú hálózat (192.168.10.0/24 és 172.16.0.0/16) számára.

A Null0 egy virtuális IOS interfész, amely nem vezet sehova, közismert nevén a "bit bucket". Azokat a csomagokat, amelyek célútvonalának a Null0 a kimenő interfésze, eldobjuk.

Az IPv4-es EIGRP automatikusan beszúr egy összevont útvonalat a Null0-hoz, ha a következő feltételek adottak:

* Van legalább egy alhálózat, amit EIGRP-n keresztül tanult meg.
* Két vagy több, EIGRP-re vonatkozó **network** forgalomirányító-konfigurációs parancs van érvényben.
* Az automatikus összevonás engedélyezve van.

A Null0 összevont útvonal célja, hogy megelőzze az irányítási hurkok létrejöttét olyan célhálózatok felé, amelyek szerepelnek ugyan az összevont címben, de valójában nem léteznek az irányítótáblában.

# Összevont útvonal

Az ábra egy olyan esetleírást mutat be, ahol egy irányítási hurok alakulhat ki:

1. Az R1 alapértelmezett útvonallal (0.0.0.0/0) rendelkezik az ISP forgalomirányítón keresztül.

2. Az R1 útvonalfrissítést küld az R2-nek, ami tartalmazza az alapértelmezett útvonalat.

3. Az R2 beilleszti az alapértelmezett útvonalat a saját IPv4-es irányítótáblájába.

4. Az R2 irányítótáblája tartalmazza a 172.16.1.0/24, 172.16.2.0/24 és 172.16.3.0/24 alhálózatokat.

5. Az R2 összevont frissítést küld az R1-nek az 172.16.0.0/16 hálózatról.

6. Az R1 beilleszti az összevont útvonalat a 172.16.0.0/16 felé az R2-n keresztül.

7. Az R1 kap egy csomagot a 172.16.4.10 célcímmel. Mivel az R1 rendelkezik útvonallal a 172.16.0.0/16 felé az R2 forgalomirányítón keresztül, továbbítja a csomagot az R2-nek.

8. Az R2 fogadja a 172.16.4.10 célcímmel rendelkező csomagot az R1-től. A csomag nem egyezik egyik konkrét útvonalával sem, így az alapértelmezett útvonalat használva az R2 a csomagot visszaküldi az R1-nek.

9. A 172.16.4.10-nek szóló csomag hurokba kerül az R1 és R2 között, amíg a TTL le nem jár, és a csomag eldobásra nem kerül.

# Manuálisan összevont útvonalak

Az EIGRP beállítható manuális útvonal-összevonásra is, akár be van kapcsolva az automatikus összevonás (**auto-summary**), akár nincs. Mivel az EIGRP osztály nélküli irányító protokoll, és küldi az alhálózati maszkot az útvonalfrissítésekben, a manuális összevonás magában foglalhatja a szuperhálózati útvonalakat is. Emlékezzünk rá, hogy a szuperhálózat több osztály-alapú főhálózat összevonása.

Az 1. ábrán még két hálózatot adtunk az R3 forgalomirányítóhoz loopback interfészek alkalmazásával, ezek: 192.168.2.0/24 és 192.168.3.0/24. Bár a loopback interfészek virtuális interfészek, ebben a példában arra használjuk őket, hogy képviseljék a fizikai hálózatokat.

A 2. ábra megjeleníti azon parancsokat, amelyekkel beállítható a két loopback interfész, valamint az EIGRP mindkét interfészen történő engedélyezése.

Annak ellenőrzésére, hogy az R3 küldött-e EIGRP frissítéseket az R1 és R2 számára, vizsgáljuk meg az irányítótáblákat mindkét eszközön.

A 3. ábrán csak a vonatkozó útvonalak láthatók. Az R1 és R2 irányítótáblái mutatják ezeket a további hálózatokat: a 192.168.2.0/24-et és a 192.168.3.0/24-et. Ahelyett, hogy három különálló hálózatot hirdetne, az R3 egyetlen útvonalként vonja össze a 192.168.1.0/24, 192.168.2.0/24 és 192.168.3.0/24 hálózatokat.

# Az EIGRP manuális útvonal-összevonásának beállítása

**Az összevont EIGRP útvonal meghatározása**

Az 1. ábra az R3 forgalomirányítón beállított két manuális összevont útvonalat jeleníti meg. Ezeket az összevont útvonalakat a Serial 0/0/0 és 0/0/1 soros interfészeken küldi ki az R3 az EIGRP szomszédainak.

A három hálózat összevont címének meghatározására ugyanazt a módszert használjuk, mint amivel meghatározzuk a statikus összevont útvonalakat, ahogy a 2. ábrán látható:

**1. lépés** Az érintett hálózati címeket írjuk fel bináris formában.

**2. lépés:** Az összevonás alhálózati maszkjának meghatározásához kezdjük a bal szélső bittel.

**3. lépés:** Balról jobbra haladva számoljuk meg az összes címben az egyező biteket.

**4. lépés:** Ha olyan bitoszlophoz érünk, amelyek elemei nem egyeznek, álljunk meg. Ez az összevonás határa.

**5. lépés:** Számoljuk meg a bal szélen egyező biteket, ami ebben a példában 22. Ez a szám határozza meg az összevont útvonal alhálózati maszkját (/22 vagy 255.255.252.0).

**6. lépés:** Az összevont hálózati cím meghatározásához a 22 egyező bitet egészítsük ki 0 bitértékekkel 32 bit hosszúságig.

Az eredmény az alábbi összevont hálózati cím és a maszk: 192.168.0.0/22.

**Az EIGRP manuális útvonal-összevonásának beállítása**

Az EIGRP manuális útvonal-összevonásának beállításához egy adott EIGRP interfészen az alábbi interfészkonfigurációs parancsot használjuk:

Router(config-if)# **ip summary-address eigrp** *as-szám hálózati-cím alhálózati-maszk*

A 2. ábra mutatja a manuális útvonal-összevonás beállítását az R3 forgalomirányító Serial 0/0/0 interfészén. Mivel az R3-nak két EIGRP szomszédja van, a manuális összevonást mind a Serial 0/0/0, mind a Serial 0/0/1 interfészen be kell állítani.

Használjuk a 3. ábrán látható parancsszimulátort ugyanannak a manuális összevont útvonalnak az R3 Serial 0/0/1 interfészén való beállításához.

# Manuálisan összevont útvonalak ellenőrzése

Az ábra azt mutatja, hogy miután az összevont útvonalat beállítottuk, az R1 és R2 irányítótáblája nem tartalmazza a különálló 192.168.1.0/24, 192.168.2.0/24 és 192.168.3.0/24 hálózatokat. Ehelyett egyetlen összevont útvonalat tartalmaznak, ez a 192.168.0.0/22. Az összevont útvonalak csökkentik a teljes útvonalak számát az irányítótáblákban, ami a bennük zajló keresési folyamatot hatékonyabbá teszi. Az összevont útvonalak emellett kisebb sávszélességet igényelnek a frissítésekhez, mert csak egy útvonalat kell küldeni, nem pedig több egyedit.

# IPv6-os EIGRP: manuálisan összevont útvonalak

Bár az automatikus összevonás nem áll rendelkezésre az EIGRP IPv6-os verziójában, itt is be lehet állítani manuálisan összevont útvonalakat.

Az 1. ábra egy EIGRP IPv6-os topológiát mutat, az R3 forgalomirányítón beállított négy visszacsatolási címmel. Ezeket a virtuális címeket használjuk a fizikai hálózatok szimulálására a R3 IPv6-os irányítótáblájában. Ezek a hálózatok manuálisan összevonhatók.

A 2. ábra mutatja az IPv6-os loopback címek konfigurációját az R3-on. Csak az R3 forgalomirányítón beállított négy loopback cím jelenik meg a topológiában; a példában azonban azt feltételezzük, hogy az összes 2001: DB8: ACAD ::/48 alhálózat az R3-on keresztül elérhető.

Az IPv6-os EIGRP manuális útvonal-összevonásának beállításához egy adott EIGRP interfészen az alábbi interfészkonfigurációs parancsot használjuk:

Router(config-if)# **ipv6 summary-address eigrp** *as-szám előtag/előtag-hossza*

A 3. ábra mutatja a manuális útvonal-összevonás konfigurációját az R1 és R2 felé a 2001:DB8:ACAD::/48 előtaggal. Az EIGRP IPv4-es verziójához hasonlóan az R3 is tartalmaz egy összevont útvonalat a Null0 felé, hurok-megelőzési mechanizmusként.

A manuálisan összevont útvonal fogadása ellenőrizhető az irányítótáblák vizsgálatával az EIGRP tartomány többi forgalomirányítóján. A 4. ábra mutatja a 2001: DB8: ACAD::/48 útvonalat az R1 forgalomirányító IPv6-os irányítótáblájában.

# Statikus alapértelmezett útvonal terjesztése

**Statikus alapértelmezett útvonal terjesztése**

A 0.0.0.0/0 statikus útvonal alapértelmezett útvonalként való használata nem irányító protokoll-függő. A "négy nullás" statikus alapértelmezett útvonal használható bármilyen jelenleg támogatott irányító protokollal. A statikus alapértelmezett útvonalat általában azon a forgalomirányítón állítjuk be, aminek van hálózati kapcsolata az EIGRP irányítási területen kívülre; például egy ISP felé.

Az 1. ábrán az R2 az átjáró eszköz, ami összeköti az EIGRP tartományt az internettel. Amikor a statikus alapértelmezett útvonalat konfiguráljuk, szükséges, hogy az egész EIGRP tartományban hirdessük ezt az útvonalat, a 2. ábrán látható módon.

Az egyik módszer a statikus alapértelmezett útvonal terjesztésére a **redistribute static** parancs használata. A **redistribute static** parancs utasítja az EIGRP-t, hogy felvegye a statikus útvonalakat is a többi eszköznek küldött frissítések közé. A 3. ábra mutatja a statikus alapértelmezett útvonal konfigurációját, és a **redistribute static** parancsot az R2 forgalomirányítón.

A 4. ábra igazolja, hogy az alapértelmezett útvonalat fogadta az R2 és beillesztette az IPv4-es irányítótáblába.

Az 5. ábrán a **show ip protocols** paranccsal ellenőrizzük, hogy az R2 újrahirdeti a statikus útvonalakat az EIGRP irányítási területen.

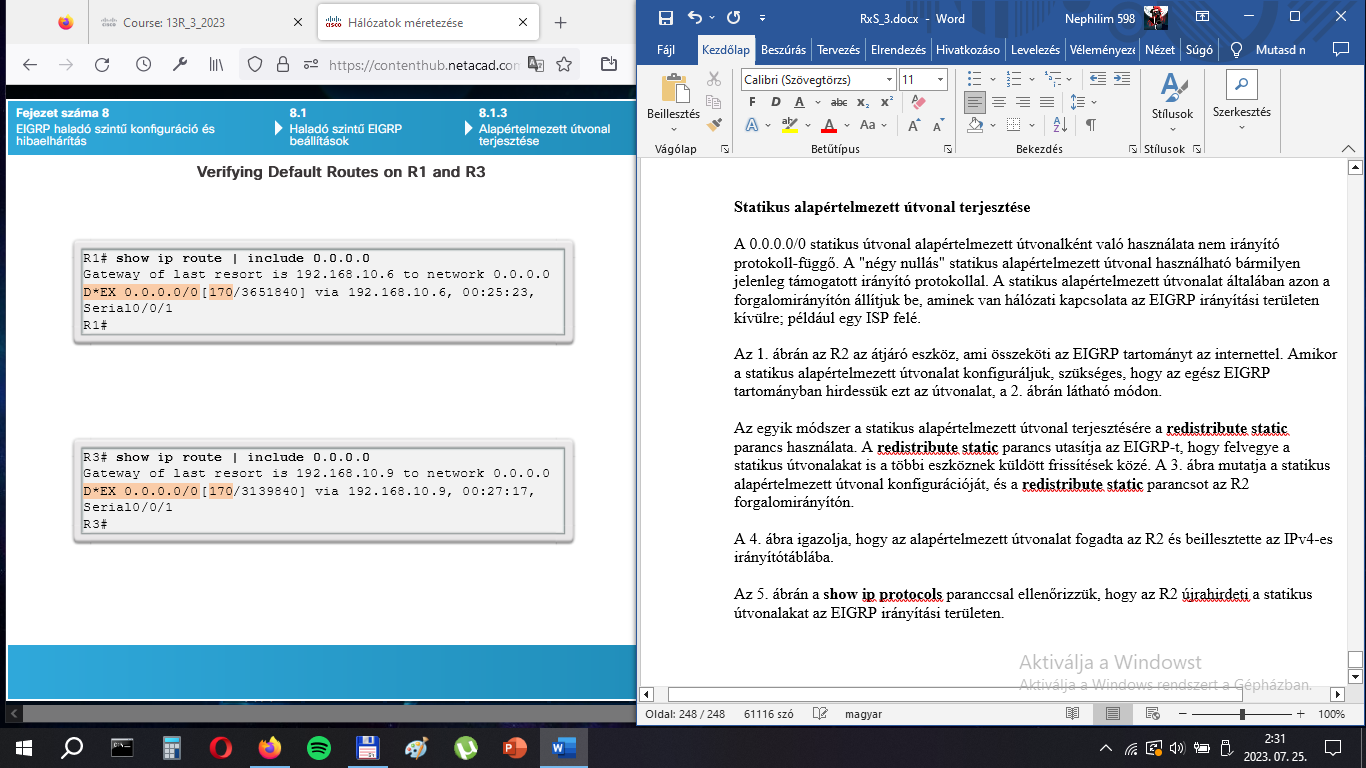
**A terjesztett alapértelmezett útvonalak ellenőrzése**

Az ábra az R1 és az R3 irányítótábláinak egy részét jeleníti meg.

Figyeljük meg az ezekben szereplő, az EIGRP segítségével tanult új alapértelmezett útvonal-forrás és adminisztratív távolság értékeit. Az így megtanult alapértelmezett útvonalat a következők azonosítják:

* **D** - Ezt az útvonalat EIGRP frissítésből tanulta meg.
* **\*** - Az útvonal egy kijelölt alapértelmezett útvonal.
* **EX** - Az útvonal egy külső EIGRP útvonal, ebben az esetben egy statikus útvonal az EIGRP tartományon kívül.
* **170** - Ez a külső EIGRP útvonal adminisztratív távolsága.

Figyeljük meg, hogy az R1 az R3 forgalomirányítót választja ki legjobbként az alapértelmezett útvonal felé, mert kisebb költségmértéke van. Az alapértelmezett útvonalak egy irányadó útvonalat jelentenek a forgalomirányítási tartományon kívülre, és mint az összevont útvonalak, minimalizálják a bejegyzések számát az irányítótáblában.



# IPv6-os EIGRP: Alapértelmezett útvonal

Emlékezzünk vissza, hogy az EIGRP külön táblázatokat tart fenn az IPv4 és az IPv6 számára; ezért az alapértelmezett IPv6-os útvonalat külön kell terjeszteni, amint az 1. ábrán is látható. Hasonlóan az IPv4-es EIGRP protokollhoz, az alapértelmezett statikus útvonal be van állítva az átjáró forgalomirányítón (R2), amint azt a 2. ábra mutatja:

R2(config)# **ipv6 route ::/0 serial 0/1/0**

A ::/0 előtag és az előtag hossza megegyezik az IPv4 esetében használt 0.0.0.0 0.0.0.0 címmel és alhálózati maszkkal. Mindkettő a csupa nullákból álló címet jelenti nulla hosszúságú előtaggal.

Az IPv6-os alapértelmezett statikus útvonalat az EIGRP tartományban terjeszteni ugyanazzal a **redistribute static** paranccsal lehet, mint amit az IPv4 esetében használtunk.

**MEGJEGYZÉS:**Néhány IOS igényelheti, hogy a **redistribute static** parancs tartalmazza a EIGRP mérték paramétereket, hogy terjeszthessük a statikus útvonalat.

**Az alapértelmezett útvonal terjesztésének ellenőrzése**

Az IPv6-os statikus alapértelmezett útvonal terjesztését le lehet ellenőrizni az R1 forgalomirányító IPv6-os irányítótáblájának vizsgálatával, a **show ipv6 route** parancs használatával, a 3. ábrán látható módon. Figyeljük meg, hogy a legjobb útvonal, vagyis a következő ugrás címe nem az R2, hanem az R3. Ez azért van, mert az R3 jobb útvonalat, alacsonyabb költségmértéket biztosít R2 forgalomirányítónak, mint az R1.

# EIGRP sávszélesség-kihasználás

**EIGRP sávszélesség-kezelés az IPv4 esetében**

Alapértelmezés szerint az EIGRP csak 50 százalékát használja egy interfész sávszélességének az irányítási információk átvitelére. Ez megakadályozza, hogy a EIGRP folyamat túlzott mértékben használja fel a kapcsolatot, és ne engedjen elegendő sávszélességet az útvonalon a szokásos forgalom számára.

Az **ip bandwidth-percent eigrp** parancs segítségével állíthatjuk be azt a százalékban értendő sávszélességet, amit az EIGRP használhat egy adott interfészen.

Router(config-if)# **ip bandwidth-percent eigrp** *as-szám százalék*

Az 1. ábrán az R1 és R2 forgalomirányítók osztoznak egy nagyon lassú, 64 kb/s-os kapcsolaton. Annak konfigurálása, hogy hogyan korlátozzuk az EIGRP által felhasználható sávszélességet, a 2. ábrán látható. Az **ip bandwidth-percent eigrp** parancs a beállított sávszélesség értékét (vagyis az alapértelmezett sávszélességet) használja fel a százalék kiszámítása során, amit az EIGRP használhat. Ebben a példában az EIGRP-t a kapcsolat sávszélességének legfeljebb 40 százalékára korlátozzuk. Emiatt az EIGRP nem használ fel 32 kb/s-nál többet a kapcsolat sávszélességéből a saját csomagjainak forgalmazására.

Az alapértelmezett érték visszaállítására a parancs **no** előtaggal ellátott változata szolgál.

A 3. ábrán lévő parancsszimulátorban korlátozzuk az EIGRP által használt sávszélességet az R2 és R3 között úgy, hogy 75 százalékát használhassa a kapcsolat sávszélességének!

**EIGRP sávszélesség-kezelés az IPv6 esetében**

A EIGRP IPv6-os verziója által felhasználható százalékos sávszélesség egy adott interfészen való konfigurálásához használjuk az **ipv6 bandwidth-percent eigrp** parancsot interfész konfigurációs módban. Az alapértelmezett érték visszaállítására a parancs **no** előtaggal ellátott változata szolgál.

Router(config-if)# **ipv6 bandwidth-percent eigrp** *as-szám százalék*

A 4. ábra mutatja az R1 és R2 közötti interfészek konfigurációját, amivel korlátozzuk az IPv6-os EIGRP által használt sávszélességet.

# Hello és megtartási időzítők

**EIGRP Hello intervallumok és megtartási időzítők az IPv4 esetében**

Az EIGRP egy egyszerű Hello protokollt használ a szomszédokkal való kapcsolat állapotának létrehozása és nyomon követése céljából. A megtartási időzítő (Hold timer) annak a legnagyobb időértéknek a meghatározására szolgál, amennyit a forgalomirányítónak várnia kell a következő Hello csomagra, mielőtt a szomszédot elérhetetlennek minősítené.

A Hello időközök és megtartási idők interfészenként külön konfigurálhatók, és nem kell megegyezniük más forgalomirányítók értékeivel a velük való szomszédság létrehozásához vagy fenntartásához. Az alapértelmezettől eltérő Hello intervallum beállítása:

Router(config-if)# **ip hello-interval eigrp** *as-szám másodperc*

Ha a Hello intervallum megváltozik, győződjünk meg arról, hogy a megtartási idő értéke vele egyenlő, vagy nagyobb nála. Ellenkező esetben a szomszédsági viszony lebomlik a megtartási idő lejárta után, mielőtt a következő Hello csomag megérkezne. Használjuk a következő parancsot más megtartási idő beállításához:

Router(config-if)# **ip hold-time eigrp** *as-szám másodperc*

A *másodperc* érték mind a Hello, mind a megtartási időintervallumok esetében 1-től 65535-ig terjedhet.

Az 1. ábra mutatja az R1 konfigurációját, amin egy 50 másodperces Hello intervallumot és 150 másodperces megtartási időt állítottunk be. A parancsok **no** formáját lehet használni az alapértelmezett értékek visszaállításához.

A Hello időközök és megtartási idők nem kell, hogy megegyezzenek két forgalomirányítón a köztük lévő szomszédság létrehozásához.

Használjuk a 2. ábrán látható parancsszimulátort, hogy ugyanazokat az értékeket állítsuk be az R2 szomszédos interfészén, mint az R1 forgalomirányítón.

**Hello intervallumok és megtartási időzítők az IPv6-os EIGRP esetében**

Az EIGRP IPv6-os verziója ugyanazon Hello intervallumot és megtartási időt használja, mint az IPv4-es protokoll. Az interfész konfigurációs parancsok hasonlóak az IPv4 esetében:

Router(config-if)# **ipv6 hello-interval eigrp** *as-szám másodperc*

Router(config-if)# **ipv6 hold-time eigrp** *as-szám másodperc*

A 3. ábra a Hello intervallum és a megtartási időzítő konfigurációját mutatja az R1 és R2 eszközön.

# Terheléselosztás IPv4 alatt

Az egyenlő költségű útvonalak közti terheléselosztás egy forgalomirányító azon képessége, amikor szét tudja osztani a kimenő forgalmat az összes interfészén, amelyek ugyanazzal a költségmértékkel rendelkeznek a célhálózat felé. A terheléselosztás a hálózati szegmensek és a sávszélesség jobb kihasználását eredményezi. Az IP protokoll esetében a Cisco IOS szoftver a terheléselosztást legfeljebb négy egyenlő költségű útvonal között végzi alapértelmezés szerint.

Az 1. ábra egy EIGRP IPv4-es hálózati topológiát szemléltet. Ebben a topológiában az R3 két egyenlő költségű EIGRP útvonallal rendelkezik az R1 és R2 közötti hálózat, a 172.16.3.0/30 felé. Az egyik útvonal az R1-en a 192.168.10.4/30, a másik útvonal pedig R2-n a 192.168.10.8/30 hálózaton keresztül halad.

A **show ip protocols** paranccsal lehet ellenőrizni a jelenleg konfigurált egyenlő költségű útvonalak számát az eszközön. A kimenet a 2. ábrán látható, eszerint az R3 az alapértelmezett négy egyenlő költségű útvonalat használja.

Az irányítótáblában mindkét útvonal benne van. A 3. ábra azt mutatja, hogy az R3 két egyenlő költségű EIGRP útvonallal rendelkezik a 172.16.3.0/30 hálózat felé. Az egyik útvonal az R1-en a 192.168.10.5, a másik útvonal pedig az R2-n a 192.168.10.9 címen keresztül halad. Az 1. ábra topológiáját szemügyre véve úgy tűnhet, mintha az R1 forgalomirányítón keresztüli útvonal jobb lenne, mivel ott egy 1544 kb/s-os kapcsolat van az R3 és R1 között, míg az R2 felé menő kapcsolat csak 1024 kb/s. Az EIGRP azonban csak a legkisebb sávszélességértéket használja az összetett mértékében, amely az R1 és R2 közötti 64 kb/s-os kapcsolat. Mindkét útvonalon ugyanaz a 64 kb/s-os kapcsolat a legkisebb sávszélesség, ez azt eredményezi, hogy mindkét útvonal egyenlő költségű.

Ha a folyamatkapcsolás van engedélyezve, az egyenlő költségű útvonalak közti terheléselosztás csomagonként történik. Amikor a csomagokat gyorskapcsolással továbbítjuk, a terheléselosztás célcím alapján működik. A Cisco Express Forwarding (CEF) képes mind csomagonkénti, mind célállomás alapú terheléskiegyenlítésre.

A Cisco IOS alapértelmezés szerint lehetővé teszi a legfeljebb négy egyenlő költségű útvonal közti terheléselosztást; ez azonban módosítható. A **maximum-paths** forgalomirányító konfigurációs paranccsal akár 32 egyenlő költségű útvonal is megjelenhet az irányítótáblában.

Router(config-router)# **maximum-paths** *érték*

Az *érték* argumentum jelenti azon útvonalak számát, amit felhasználhatunk a terhelés kiegyenlítésére. Ha a beállított érték **1**, a terheléselosztás le van tiltva.

**Az irányító protokollok hitelesítésének áttekintése**

**Az irányító protokollok hitelesítése**

A hálózati rendszergazdáknak tisztában kell lenniük azzal, hogy a forgalomirányítók ugyanúgy ki vannak téve a támadásoknak, mint a végfelhasználói eszközök. Bárki, aki (a Wireshark programhoz hasonló) csomagfigyelőt használ, információt képes kiolvasni a forgalomirányítók közötti forgalomból. A forgalomirányító rendszerek általában a szomszédos eszközök kapcsolatának megszakítása vagy az forgalomirányítási információk meghamisítása révén támadhatóak.

A kapcsolatok megszakítása kevésbé kritikus a két támadástípus közül, mivel az irányító protokollok képesek gyógyítani magukat, így a zavar csak egy kicsivel hosszabb ideig tart, mint maga a támadás.

Az irányítási információk meghamisítása a támadások kifinomultabb osztálya, ami az irányítóprotokoll által szállított információt célozza meg. Ennek következményei az alábbiakban foglalhatók össze:

* forgalom átirányítása irányítási hurkok létrehozásával
* a forgalom nem biztonságos vonalra történő átirányítása megfigyelés céljából
* forgalom átirányítása annak eldobása érdekében

Az irányítási információk védelmének egyik módszere, hogy a protokoll csomagjainak hitelesítése a hálózaton a Message Digest 5 (MD5) algoritmussal történik. Az MD5 segítségével a forgalomirányító képes az elvileg megegyező digitális aláírások összehasonlítására: ez megerősíti, hogy hiteles forrásból származnak.

Egy ilyen rendszer az alábbi három komponensből áll:

* Titkosítási algoritmus, ami általában mindenki által ismert
* A titkosítási algoritmushoz használt kulcs, ami egy forgalomirányítók között megosztott titkos információ a csomagok hitelesítésére
* Maga a csomag tartalma

Kattintsunk az ábrán a lejátszás gombra annak megtekintéséhez, hogy az egyes forgalomirányítók miként hitelesítik az információkat. Általában az információk forrásaként szolgáló eszköz készít egy aláírást a kulcs és a küldendő irányítási adatok segítségével, a titkosítási algoritmus bemeneteként. A fogadó eszköz ezután megismétli a folyamatot ugyanazzal a kulccsal, és ugyanazzal a kapott irányítási információval. Ha a vevő által számított aláírás megegyezik az elküldöttel, a feladó által küldött frissítés hitelesített és megbízhatónak tekinthető.

Az irányító protokollok, mint a RIPv2, EIGRP, OSPF, IS-IS és BGP mindegyike támogatja az MD5 hitelesítés különböző formáit.

# Az EIGRP beállítása MD5 hitelesítés használatára

Az EIGRP üzenethitelesítés biztosítja, hogy a forgalomirányítók csak akkor fogadjanak el üzeneteket más eszközöktől, ha megegyznek az előre megosztott kulcsaik. Ha nincs hitelesítés beállítva, egy illetéktelen beállíthat egy másik forgalomirányítót, hogy eltérő vagy egymásnak ellentmondó útvonal-információkat terjesszen a hálózaton, így a legitim forgalomirányítók irányítótáblái hamissá válhatnak és DoS támadás következhet be. Így az EIGRP üzeneteket küldő forgalomirányítók közötti hitelesítés beállításával megakadályozzuk, hogy valaki szándékosan vagy véletlenül hozzáadhasson egy új eszközt a hálózathoz és problémát okozzon ezáltal.

Az EIGRP az MD5 típusú irányítóprotokoll hitelesítést támogatja. Az EIGRP üzenethitelesítésének konfigurációja két lépésből áll: egy kulcslánc (kulcstartó) és egy kulcs létrehozásából, valamint az EIGRP hitelesítés konfigurációjából, hogy használja is ezt a kulcsot.

**1. lépés: A kulcslánc és a kulcs létrehozása**

A hitelesítéshez egy kulcsláncra és rajta egy kulcsra van szükség. Mielőtt a hitelesítést engedélyeznénk, hozzunk létre egy kulcsláncot és legalább egy kulcsot.

a. Globális konfigurációs módban hozzuk létre a kulcsláncot. Bár több kulcsot is be lehet állítani, ez a rész csak egy kulcs használatára összpontosít.

Router(config)# **key chain** *lánc-neve*

b. Adjuk meg a kulcsazonosítót (key ID). A kulcsazonosító száma egy adott hitelesítési kulcs azonosítására szolgál a kulcsláncon belül. A kulcssorszámok tartománya 0-tól 2147483647-ig terjed. Ajánlott, hogy a kulcs azonosítója azonos legyen minden forgalomirányító konfigurációjában.

Router(config-keychain)# **key** *kulcsazonosító*

c. Adjuk meg a kulcshoz tartozó karakterláncot. A karakterlánc egy jelszóhoz hasonló. A hitelesítési kulcsot cserélő eszközöket azonos karakterlánc használatára kell beállítani.

Router(config-keychain-key )# **key-string** *kulcs-karakterlánc*

**2. lépés: Az EIGRP hitelesítés konfigurálása kulcslánc és kulcs használatával**

Állítsuk be az EIGRP-t az üzenethitelesítés elvégzésére a korábban létrehozott kulcs segítségével. Végezzük el ezt a beállítást minden olyan interfészen, amin engedélyeztük az EIGRP-t.

a. Globális konfigurációs módban adjuk meg az interfészt, amelyiken konfigurálni akarjuk az EIGRP üzenethitelesítést.

Router(config)# **interface** *típus szám*

b. Engedélyezzük az üzenethitelesítést. Az **md5** kulcsszó jelzi, hogy MD5 kivonatot (hash-t) kell használni a hitelesítéshez.

Router(config-if)# **ip authentication mode eigrp** *AS-azonosító* **md5**

c. Adjuk meg a hitelesítéshez használni kívánt kulcsláncot. A *lánc-neve* argumentum adja meg a kulcsláncot, ami az 1. lépésben jött létre.

Router(config-if)# **ip authentication key-chain eigrp** *AS-szám lánc-neve*

Mindegyik kulcsnak saját, helyben tárolt kulcsazonosítója van. A kulcs azonosítójának és az üzenethez tartozó interfésznek a kombinációja egyedileg azonosítja a hitelesítési algoritmust és a használatban lévő MD5 hitelesítési kulcsot. A kulcslánc és a frissítések feldolgozása az MD5 algoritmus felhasználásával történik, így áll elő az egyedi aláírás.

# Példa az EIGRP hitelesítés használatára

Az irányítási frissítések hitelesítésére minden EIGRP-kompatibilis interfészt be kell beállítani úgy, hogy támogassa a hitelesítést. Az 1. ábra mutatja az IPv4-es topológiát, valamint a hitelesítés használatára konfigurált interfészeket.

A 2. ábra mutatja az R1 konfigurációját, benne az **EIGRP\_KEY** kulcslánccal és a **cisco123** karakterlánccal. Miután az R1 beállítása megtörtént, a többi eszköz hitelesített irányítási frissítéseket kap. A szomszédságok elvesznek, amíg a szomszédok beállítása meg nem történik a protokoll-hitelesítés használatára.

A 3. ábra egy hasonló konfigurációt mutat az R2 forgalomirányítón. Figyeljük meg, hogy ugyanannak a **cisco123**karakterláncnak a segítségével hitelesítjük az R1 és az R3 eszközöktől származó információkat.

Használjuk a 4. ábrán látható parancsszimulátort az EIGRP hitelesítés R3 forgalomirányítón történő beállításához.

**Az IPv6-os EIGRP hitelesítés beállítása**

A hitelesítési algoritmusok és az IPv6-os EIGRP üzenetek hitelesítésének beállítása megegyezik az IPv4-es verzióval. Az egyetlen különbség az interfész konfigurációs módban az **ipv6**parancs használata az **ip**helyett.

Router(config-if)# **ipv6 authentication mode eigrp** *AS-azonosító* **md5**

Router(config-if)# **ipv6 authentication key-chain eigrp** *AS-szám lánc-neve*

Az 5. ábra látható parancsokkal állíthatjuk be az IPv6-os EIGRP hitelesítést az R1 forgalomirányítón, az **EIGRP\_IPV6\_KEY** kulcslánc és a **cisco123** karakterlánc használatára. Hasonló konfigurációk adhatók meg az R2 és R3 eszközön is.

# A hitelesítés ellenőrzése

Miután beállítottuk az EIGRP üzenethitelesítést egy forgalomirányítón, a hitelesítéshez még nem beállított egyetlen szomszédos eszköz sem lehet EIGRP szomszéd. Például, ha az R1 forgalomirányító Serial 0/0/0 interfészét beállítottuk az MD5 hitelesítésre, de az R2 még nem lett beállítva, a következő IOS üzenet jelenik meg az R1-en:

%DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 172.16.3.2 (Serial0/0/0) is down: authentication mode changed

Ha a szomszédos Serial 0/0/0 interfész beállítása az R2-n megtörtént, a szomszédság helyreáll, és a következő IOS üzenet jelenik meg az R1-en:

%DUAL-5-NBRCHANGE: EIGRP-IPv4 1: Neighbor 172.16.3.2 (Serial0/0/0) is up: new adjacency

Hasonló üzenetek jelennek meg az R2-n is.

Szomszédságok csak akkor alakulnak ki, ha mindkét összekötött eszközön be van állítva a hitelesítés, amint az 1. ábrán is látható. Annak ellenőrzésére, hogy a megfelelő EIGRP szomszédságok alakultak ki a hitelesítés konfigurálása után, használjuk a **show ip eigrp neighbors** parancsot minden forgalomirányítón. A 2. ábra azt mutatja, hogy mind a három eszközön helyreállt a szomszédsági viszony, miután beállítottuk őket az EIGRP hitelesítés használatára.

Az IPv6-os EIGRP szomszédsági viszonyainak ellenőrzésére használjuk a **show ipv6 eigrp neighbors** parancsot.

**Alapvető EIGRP hibaelhárítási parancsok**

Az EIGRP-t gyakran használják nagyvállalati hálózatokban. Az irányítási információk cseréjével kapcsolatos hibaelhárítási problémák megoldása fontos készség egy hálózati rendszergazda számára. Ez különösen igaz azokra a rendszergazdákra, akik részt vesznek az EIGRP-t mint belső átjáró protokollt (IGP) használó nagy, irányított vállalati hálózatok kivitelezésében és karbantartásában. Számos parancs létezik, amely hasznos lehet, ha egy EIGRP hálózat hibaelhárítására kerül sor.

A **show ip eigrp neighbors** parancs ellenőrzi, hogy a forgalomirányító felismeri-e a szomszédait. Az 1. ábra kimenete azt mutatja, hogy az R1-nek két működő EIGRP szomszédja van.

A 2. ábrán a **show ip route** parancs ellenőrzi, hogy a forgalomirányító megtanulta-e a útvonalat a távoli hálózat felé az EIGRP révén. A kimenet azt jelzi, hogy az R1 négy távoli hálózatot tanult meg az EIGRP segítségével.

A 3. ábrán a **show ip protocols** parancs kimenete látható. Ez a parancs ellenőrzi, hogy a EIGRP megjeleníti-e az aktuálisan beállított értékeket, illetve bármely engedélyezett irányító protokoll különböző tulajdonságait.

**IPv6-os EIGRP**

Hasonló parancsok és hibaelhárítási szempontok vonatkoznak az EIGRP IPv6-os verziójára is.

A következő egyenértékű parancsok használhatók ebben az esetben:

* Router# **show ipv6 eigrp neighbors**
* Router# **show ipv6 route**
* Router# **show ipv6 protocols**

**Összetevők**

Az ábra egy folyamatábrát mutat az EIGRP kapcsolódási problémáinak diagnosztizálásához.

Miután beállítottuk az EIGRP-t, első lépésként teszteljük a kapcsolatot a távoli hálózat felé. Ha a pingelés sikertelen, ellenőrizzük a EIGRP szomszédsági viszonyokat. A szomszédság számos okból nem jöhet létre, többek között:

* A készülékek közötti interfész nem működik.
* A két forgalomirányítón nem egyeznek az EIGRP autonóm rendszer azonosítók (folyamatazonosítók).
* A megfelelő interfészeken nincs engedélyezve az EIGRP folyamat.
* Az interfész passzívnak van beállítva.

Eltekintve ezektől, számos más, komolyabb probléma is eredményezheti, hogy a szomszédság nem alakítható ki. Két példa erre a rosszul konfigurált EIGRP hitelesítés, illetve a rosszul beállított K-értékek, amelyek az EIGRP mértékszámításához használatosak.

Ha az EIGRP szomszédság kialakult a két forgalomirányító között, de még mindig csatlakozási probléma van, előfordulhat, hogy ez egy irányítási probléma. Kapcsolódási problémát okozhatnak az alábbi esetek:

* A távoli forgalomirányító nem hirdeti a megfelelő hálózatokat.
* Egy nem megfelelően konfigurált passzív interfész vagy ACL blokkolja a távoli hálózatokról szóló hirdetményeket.
* Az automatikus útvonal-összevonás nem egységes útválasztást okoz egy nem-folytonos hálózatban.

Ha az összes szükséges útvonal megjelenik az irányítótáblában, de az útvonal, amin a forgalom halad, mégsem helyes, ellenőrizzük az interfészek sávszélesség értékeit.

# 3. rétegbeli kapcsolat

Két közvetlenül kapcsolódó forgalomirányító közötti szomszédsági viszony kialakulásának előfeltétele a 3. rétegbeli kapcsolat megléte. Megvizsgálva a **show ip interface brief** parancs kimenetét, a hálózati rendszergazda ellenőrizheti az összekötő interfészek működőképes állapotát és vonali protokollját. Az egyik forgalomirányítóról egy másik, közvetlenül kapcsolódó eszközre pingelve ellenőrizni lehet a két eszköz közötti IPv4-es kapcsolatot. Az ábra mutatja a **show ip interface brief** parancs kimenetét az R1 forgalomirányítón. Az R1 kapcsolatot mutat az R2 felé és a ping sikeres.

Ha a ping sikertelen, ellenőrizzük a kábelezést, és hogy a csatlakoztatott eszközök interfészei egy közös alhálózaton vannak-e. Egy rendszernapló üzenet is mutatja, hogy az EIGRP szomszédok nincsenek közös hálózaton ( not on common subnet ). Ez azt jelzi, hogy hibás IPv4-es cím van beállítva a két EIGRP szomszéd közül az egyik interfészén.

**IPv6-os EIGRP**

Hasonló parancsok és hibaelhárítási szempontok vonatkoznak az EIGRP IPv6-os verziójára is.

A következő egyenértékű parancs használható ebben az esetben: **show ipv6 interface brief**.

# EIGRP paraméterek

Amikor egy EIGRP hálózat hibaelhárítását végezzük, az egyik első dolog, hogy ellenőrizzük minden EIGRP hálózatban részt vevő forgalomirányítón, hogy azonos autonóm rendszer azonosítók vannak-e beállítva. A **router eigrp** *AS-azonosító* parancs elindítja az EIGRP folyamatot, amit egy szám követ, ez az autonóm rendszer azonosítója. Az *AS-azonosító* paraméterben megadott értéknek azonosnak kell lennie minden forgalomirányítón az EIGRP irányítási területen.

Az 1. ábra azt mutatja, hogy minden forgalomirányító az 1-es számú autonóm rendszerben található. A 2. ábrán a **show ip protocols** parancs ellenőrzi, hogy az R1, R2 és R3 mindegyike ugyanazt az autonóm rendszer azonosítót használja.

**IPv6-os EIGRP**

Hasonló parancsok és hibaelhárítási szempontok vonatkoznak az EIGRP IPv6-os verziójára is.

A következő egyenértékű parancsok használhatók ebben az esetben:

* Router(config)# **ipv6 router eigrp** *AS-azonosító*
* Router# **show ipv6 protocols**

**MEGJEGYZÉS:**A kimenet tetején látható szövegben ("IP Routing is NSF aware") az NSF kifejezés a Nonstop Forwarding (folyamatos továbbítás) rövidítése. Ez a képesség lehetővé teszi az EIGRP partnerek számára, hogy megtartsanak egy meghibásodott forgalomirányító által hirdetett információkat, és hogy továbbra is használják ezeket, amíg a hibás eszköz visszaáll normál üzemmódra és képes lesz újra megosztani irányítási adatokat. További információkért lásd: <http://www.cisco.com/en/US/docs/ios-xml/ios/iproute_eigrp/configuration/15-mt/eigrp-nsf-awa.html>

**EIGRP interfészek**

Amellett, hogy megbizonyosodunk az autonóm rendszer azonosító helyességéről, azt is meg kell vizsgálni, hogy az összes interfész részt vesz-e az EIGRP hálózatban. Az EIGRP irányítási folyamat beállításánál kiadott **network** parancs megmutatja, hogy mely interfészek vesznek részt az EIGRP folyamatban. Ez a parancs az interfész osztály-alapú hálózati címére, helyettesítő maszk használata esetén pedig egy alhálózat címére vonatkozik.

Az 1. ábrán a **show ip eigrp interfaces** parancs megjeleníti, hogy mely interfészeken van engedélyezve az EIGRP az R1 forgalomirányítón. Amennyiben a csatlakoztatott interfészeken nincs engedélyezve az EIGRP, a szomszédos eszközök nem lépnek szomszédsági viszonyba.

A 2. ábrán a "Routing for Networks" rész a **show ip protocols** parancs kimenetéből jelzi, hogy mely hálózatok vannak beállítva; az e hálózatokba eső minden interfész részt vesz az EIGRP irányításban.

Ha a hálózat nem szerepel ebben a részben, használjuk a **show running-config** parancsot, hogy meggyőződjünk arról, hogy a **network** parancs be van-e állítva.

A 3. ábrán a kimenet a **show running-config** parancsból megerősíti, hogy az ezekkel a címekkel vagy valamely alhálózatukkal rendelkező interfészen engedélyezve van az EIGRP.

**IPv6-os EIGRP**

Hasonló parancsok és hibaelhárítási szempontok vonatkoznak az EIGRP IPv6-os verziójára is.

A következő egyenértékű parancsok használhatók ebben az esetben:

* Router# **show ipv6 protocols**
* Router# **show ipv6 eigrp interfaces**

**Passzív interfész**

Annak egyik oka, hogy az irányítótáblák nem mutatják a helyes útvonalakat, a **passive-interface** parancs lehet. Ha EIGRP protokollt használunk a hálózaton, a **passive-interface** parancs leállítja mind a kimenő, mind a bejövő útvonalfrissítéseket. Emiatt a forgalomirányítók nem válnak szomszédokká.

Annak ellenőrzésére, hogy bármilyen interfész passzívként van-e beállítva, használjuk a **show ip protocols** parancsot privilegizált EXEC módban. Az 1. ábra azt mutatja, hogy az R2 forgalomirányító GigabitEthernet 0/0 interfésze passzív interfészként van beállítva, mivel nincsenek szomszédok azon a kapcsolaton.

Amellett, hogy szomszédokhoz nem kapcsolódó interfészeken állítjuk be, a passzív módot engedélyezni lehet biztonsági okokból is. A 2. ábrán figyeljük meg, hogy az EIGRP irányítási tartomány árnyékolt része különbözik a korábbi topológiáktól. A 209.165.200.224/27 hálózatot is bevontuk az R2-n az EIGRP frissítések közé. Biztonsági okokból azonban a hálózati rendszergazda nem akarja, hogy az R2 forgalomirányító EIGRP szomszédságra lépjen az ISP forgalomirányítóval.

A 3. ábra mutatja a 209.165.200.224/27 hálózat hozzáadását a **network** paranccsal az R2 forgalomirányítón. Az R2 most már hirdeti ezt a hálózatot a többi eszköz felé az EIGRP irányítási területen.

A **passive-interface** forgalomirányító konfigurációs parancsot beállítottuk a Serial 0/1/0 interfészen annak megelőzésére, hogy az R2 forgalomirányító EIGRP frissítéseket küldjön az ISP forgalomirányítónak. A **show ip eigrp neighbors** parancs az R2-n igazolja, hogy az eszköz nem hozott létre szomszédsági viszonyt az ISP-vel.

A 4. ábra azt mutatja, hogy az R1 rendelkezik egy EIGRP útvonallal a 209.165.200.224/27 hálózat felé az IPv4-es irányítótáblájában (az R3-nak is lesz egy útvonala ehhez a hálózathoz). Az R2-nek viszont nincs szomszédsági viszonya az ISP forgalomirányítóval.

**IPv6-os EIGRP**

Hasonló parancsok és hibaelhárítási szempontok vonatkoznak az EIGRP IPv6-os verziójára is.

A következő egyenértékű parancsok használhatók ebben az esetben:

* Router# **show ipv6 protocols**
* Router(config-rtr)# **passive-interface** *típus szám*

**Hiányzó network parancs**

Az 1. ábra azt mutatja, hogy az R1 forgalomirányító GigabitEthernet 0/1 interfészén már beállították az 10.10.10.1/24 címet, és aktív is.

Az R1 és R3 még szomszédsági viszonyban van, de a ping teszt az R3-ról az R1 forgalomirányító G0/1 interfész 10.10.10.1 címére sikertelen. A 2. ábra egy sikertelen kapcsolatvizsgálatot mutat az R3 forgalomirányítóról a 10.10.10.0/24 cél hálózat felé.

A 3. ábrán a **show ip protocols** parancs az R1-en azt mutatja, hogy a 10.10.10.0/24 hálózat nincs hirdetve az EIGRP szomszédok felé.

Amint a 4. ábrán is látható, az R1 forgalomirányítón futó EIGRP folyamat úgy lett beállítva, hogy tartalmazza a 10.10.10.0/24 hálózat hirdetését.

Az 5. ábra mutatja, hogy már megjelent egy útvonal az R3 irányítótáblájában a 10.10.10.0/24 hálózathoz, és az elérhetőségét is ellenőrizték az R1 forgalomirányító GigabitEthernet 0/1 interfészének megpingelésével.

**IPv6-os EIGRP**

Hasonló parancsok és hibaelhárítási szempontok vonatkoznak az EIGRP IPv6-os verziójára is.

A következő egyenértékű parancsok használhatók ebben az esetben:

* Router# **show ipv6 protocols**
* Router# **show ipv6 route**

A hiányzó hálózat hozzáadásához IPv6 esetében használjuk az **ipv6 eigrp** *autonóm-rendszer* parancsot interfész konfigurációs módban: Router(config-if)# **ipv6 eigrp** *autonóm-rendszer*

**MEGJEGYZÉS:**a hiányzó útvonal egy másik esetben eredhet abból is, hogy szűrik a bejövő vagy kimenő útvonalfrissítéseket. Az ACL-ek különböző protokollokhoz biztosítanak szűrést, és ezek az ACL-ek befolyásolhatják az irányítóprotokoll üzeneteinek cseréjét, így bizonyos útvonalak hiányozhatnak az irányítótáblából. A **show ip protocols** parancs megmutatja, hogy van-e olyan ACL, ami az EIGRP-re vonatkozik.

# Automatikus útvonal-összevonás

Egy másik dolog, ami problémákat okozhat a hálózati rendszergazda számára, az EIGRP automatikus útvonal-összevonása.

Az 1. ábrán egy, a fejezetben eddig használt hálózati topológiától eltérő topológia látható. Nincs összeköttetés az R1 és az R3 között. Az R1 forgalomirányító LAN-ja a 10.10.10.0/24 hálózati címmel, míg az R3 forgalomirányító LAN-ja a 10.20.20.0/24 címmel rendelkezik. A soros kapcsolat mindkét forgalomirányító és az R2 között azonos, 1024 kb/s sávszélességű.

Az R1 és R3 forgalomirányítók LAN és soros interfészein engedélyezve van az EIGRP, amint a 2. ábrán látható. Mindkét eszköz automatikus összevonást végez.

Az IPv4-es EIGRP protokollt be lehet úgy állítani, hogy automatikusan vonja össze az útvonalakat az osztály-alapú határokon. Ha nem folytonos hálózatok vannak, az automatikus összevonás nem egységes forgalomirányítást eredményez.

A 3. ábrán az R2 irányítótáblája azt mutatja, hogy nem kap külön útvonalakat a 10.10.10.0/24 és 10.20.20.0/24 alhálózatok felé. Az R1 és R3 mindegyike automatikusan összevonja ezeket az alhálózatokat a 10.0.0.0/8 osztály-alapú határon, amikor az EIGRP frissítéseket küldi az R2-nek. Az eredmény az, hogy az R2 forgalomirányító két egyenlő költségű útvonalat lát a 10.0.0.0/8 felé az irányítótáblájában, ami azt eredményezheti, hogy pontatlan útvonalat használ és csomagvesztés lép fel. Attól függően, hogy csomagonkénti, cél szerinti vagy CEF terheléskiegyenlítést használ, a csomagokat vagy továbbítani lehet a megfelelő interfészen át, vagy nem.

A 4. ábrán a **show ip protocols** paranccsal ellenőrizhető, hogy az automatikus összevonás működik mind az R1-en, mind az R3-on. Figyeljük meg, hogy a két eszköz ugyanazzal a költségmértékkel vonja össze a 10.0.0.0/8 hálózatot.

Az **auto-summary** parancs alapértelmezés szerint le van tiltva a Cisco IOS szoftver 15-ös és a 12.2(33)-nál újabb verzióiban. A régebbi szoftververziókban az automatikus útvonal-összevonás alapértelmezés szerint engedélyezett. Ahhoz, hogy kikapcsoljuk az automatikus összevonást, adjuk ki a **no auto-summary** parancsot a **router EIGRP** konfigurációs módban.

A probléma kijavításához az R1-en és az R3-on is le kell tiltani az automatikus összevonást:

R1(config)# **router eigrp 1**

R1(config-router)# **no auto-summary**

R3(config)# **router eigrp 1**

R3(config-router)# **no auto-summary**

Miután az automatikus összevonást letiltottuk az R1 és R3 forgalomirányítókon, az R2 irányítótáblája azt jelzi, hogy megkapta a 10.10.10.0/24 és 10.20.20.0/24 alhálózatokat külön az R1-től és az R3-tól, ahogy az 5. ábrán látható. A pontos forgalomirányítás és a kapcsolat mindkét alhálózat felé helyreállt.

**IPv6-os EIGRP**

Osztály-alapú hálózatok nem léteznek IPv6 alatt; ezért az EIGRP IPv6-os verziója nem támogatja az automatikus összevonást. Minden összevonást az EIGRP manuálisan összevont útvonalaival kell megvalósítani.

# Csoportos feladat - EIGRP trükkök

**EIGRP trükkök**

E tevékenység célja, hogy áttekintsük az EIGRP irányítóprotokoll finomhangolásával kapcsolatos fogalmakat.

Tervezzünk meg egy EIGRP topológiát az egyik társunkkal közösen. Ez a topológia lesz az alapja a tevékenység két részének. Az első az alapértelmezett beállításokat használja minden konfigurációhoz, a második pedig foglaljon magába legalább hármat az alábbi EIGRP finomhangolási lehetőségekből:

* Manuálisan összevont útvonalak
* Alapértelmezett útvonalak
* Alapértelmezett útvonal terjesztése
* Hello időzítő beállításai

Nézzük át a laborgyakorlatokat, Packet Tracer és interaktív feladatokat, amik segíthetnek ezen modellezési tevékenységben.

A PDF-fájlban útmutatások szerepelnek ehhez a feladathoz. Mutassuk be az elvégzett munkát egy másik csoportnak! Érdemes lehet másolatot készíteni erről a tevékenységről egy portfólióba.

[Class Activity - Tweaking EIGRP](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/course/files/8.3.1.1%20Tweaking%20EIGRP%20Instructions.pdf)

# Összefoglalás

Az EIGRP egyike a nagyvállalati hálózatokban általánosan használt irányítóprotokolloknak. Az EIGRP funkciók módosítása és hibaelhárítása az egyik legfontosabb készsége egy hálózati mérnöknek, aki részt vesz olyan nagyvállalati hálózatok megvalósításában és karbantartásában, ahol EIGRP protokollt használnak.

Az útvonal-összevonás csökkenti az útvonalfrissítésekben szereplő bejegyzések, valamint a helyi irányítótábla bejegyzéseinek számát is. Javítja az útvonalfrissítések sávszélesség-kihasználását és gyorsítja az irányítótáblában való keresést. Az EIGRP IPv4-es verziójánál az automatikus összevonás alapértelmezés szerint le van tiltva a Cisco IOS 15.0(1)M és 12.2(33) verziótól kezdődően. Ezt megelőzően az automatikus összevonás alapértelmezés szerint engedélyezett volt. Az automatikus útvonal-összevonás engedélyezése érdekében használjuk az **auto-summary** parancsot forgalomirányító konfigurációs üzemmódban. Használjuk a **show ip protocols** parancsot az automatikus összevonás állapotának ellenőrzésére. Vizsgáljuk meg az irányítótáblát, és ellenőrizzük, hogy az összevonás működik-e.

Az EIGRP automatikusan hozzárendeli a Null0 interfészhez azokat az összevont útvonalakat, amelyek szerepelnek ugyan az összevont címben, de valójában nem léteznek az irányítótáblában, annak megelőzésére, hogy irányítási hurkok alakuljanak ki. A Null0 egy virtuális IOS interfész, amely nem vezet sehova, közismert nevén a "bit bucket". Azokat a csomagokat, amelyek célútvonalának a Null0 a kimenő interfésze, eldobjuk.

Az EIGRP manuális útvonal-összevonásának beállításához egy adott EIGRP interfészen az alábbi interfészkonfigurációs parancsot használjuk:

Router(config-if)# **ip summary-address eigrp** *as-szám hálózati-cím alhálózati-maszk*

Az IPv6-os EIGRP manuális útvonal-összevonásának beállításához egy adott EIGRP interfészen az alábbi interfészkonfigurációs parancsot használjuk:

Router(config-if)# **ipv6 summary-address eigrp** *as-szám előtag/előtag-hossza*

Az egyik módszer a statikus alapértelmezett útvonal terjesztésére az EIGRP irányítási tartományon belül a **redistribute static** parancs. Ez a parancs arra utasítja az EIGRP-t, hogy felvegye ezt a statikus útvonalat is a többi eszköznek küldött frissítések közé. A **show ip protocols** parancs ellenőrzi, hogy a statikus útvonalakat az EIGRP tartományban terjesztjük-e.

Az **ip bandwidth-percent eigrp** *as-szám százalék* interfész konfigurációs parancs segítségével állíthatjuk be azt a százalékban értendő sávszélességet, amit az EIGRP használhat egy adott interfészen.

A EIGRP IPv6-os verziója által felhasználható százalékos sávszélesség egy adott interfészen való konfigurálásához használjuk az **ipv6 bandwidth-percent eigrp** parancsot interfész konfigurációs módban. Az alapértelmezett érték visszaállítására a parancs **no** előtaggal ellátott változata szolgál.

A Hello időközök és megtartási idők interfészenként külön konfigurálhatók, és nem kell megegyezniük más forgalomirányítók értékeivel a velük való szomszédság létrehozásához vagy fenntartásához.

Az IP protokoll esetében a Cisco IOS szoftver a terheléselosztást legfeljebb négy egyenlő költségű útvonal között végzi alapértelmezés szerint. A **maximum-paths** forgalomirányító konfigurációs paranccsal akár 32 egyenlő költségű útvonal is megjelenhet az irányítótáblában.

Az EIGRP az MD5 típusú irányítóprotokoll hitelesítést támogatja. A hitelesítési algoritmusok és az IPv4-es EIGRP üzenetek hitelesítésének beállítása megegyezik az IPv6-os verzióval. Az egyetlen különbség az interfész konfigurációs módban az **ip**parancs használata az **ipv6**helyett.

Router(config-if)# **ipv6 authentication mode eigrp** *AS-azonosító* **md5**

Router(config-if)# **ipv6 authentication key-chain eigrp** *AS-szám lánc-neve*

Annak ellenőrzésére, hogy a megfelelő EIGRP szomszédságok alakultak ki a hitelesítés konfigurálása után, használjuk a **show ip eigrp neighbors** parancsot minden forgalomirányítón.

A **show ip route** parancs ellenőrzi, hogy a forgalomirányító megtanulta-e az EIGRP útvonalat. A **show ip protocols** parancs használható annak ellenőrzésére, hogy az EIGRP megjeleníti-e az aktuálisan beállított értékeket.

# 

* [Fejezet száma 9 IOS képfájlok és licencelés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#9)
* [9.0 IOS képfájlok és licencelés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#9.0)
* [9.0.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#9.0.1)
* [9.0.1.1 Bevezetés](https://contenthub.netacad.com/legacy/CCNA/ScaN/5.03/hu/index.html#9.0.1.1)

# Bevezetés

A Cisco IOS (korábban Internetwork Operating System) a legtöbb Cisco forgalomirányítón és kapcsolón használt szoftver. Az IOS olyan integrált, többfeladatos operációs rendszer, mely forgalomirányítási, kapcsolási és biztonsági technológiákat valósít meg.

A Cisco IOS képfájlok különböző változatai a legkülönbözőbb technológiai és képességbeli elvárásoknak is megfelelnek. A megfelelő IOS képfájlt az általa támogatott protokollok és szolgáltatások alapján érdemes kiválasztani. A szervezetünk számára legmegfelelőbb IOS verzió kiválasztásához, érdemes alaposan tanulmányozni a Cisco által kialakított tulajdonságkészletek portfólióját.

Az IOS 12.4-ről 15.0-ra történő verzióváltáskor lényeges újításokat vezetett be a Cisco az IOS képfájlok összeállítása és licencelése terén. A fejezet részletesen bemutatja az IOS 12.4 és 15.0 verzióinak névhasználati szokásait és tartalmi jellegzetességeit. Az IOS 15-ös verziójával kezdődően új összeállítási és licencelési elveket követ a Cisco. A fejezet részletesen bemutatja a Cisco IOS 15 szoftver licencek beszerzési, telepítési és menedzselési lehetőségeit.

**MEGJEGYZÉS**: Az IOS 12.4-es verzióját a 15.0 verzió követi. 13-as és 14-es verziót nem adtak ki az IOS-ból.

**Cisco IOS szoftvercsaládok (Families) és sorozatok (Trains)**

A Cisco IOS szoftver sokat fejlődött első megjelenése óta: egyplatformos, célzottan forgalomirányításra kidolgozott operációs rendszerből olyan, fejlett operációs rendszerré nőtte ki magát, amely számos előremutató technológiát támogat a VoIP-tól a NetFlow-n át az IPSec-ig. Annak érdekében, hogy a piaci elvárások legszélesebb körét tudja megfelelően kiszolgálni, a szoftver egyes kiadásait családokba (family) és sorozatokba (train) rendezik.

Azokat az IOS szoftver verziókat tekintjük egy családba tartozónak, melyek:

* kódbázisa közös
* összetartozó hardverplatformokon alkalmazhatók
* a támogatási időszakuk átfedésben van (amikor az egyik operációs rendszer kifut, bevezetésre kerül a következő és annak támogatása folyamatban van)

Ennek megfelelően egy IOS családba tartozik a 12.3, 12.4 verziószámú kiadás és a 15.0, valamint a 15.1 kiadás.

Az egyes szoftver kiadások mindig új képességeket vagy hibajavításokat tartalmazó verziókat jelentenek. Az IOS esetében ezeket a verziókat hívjuk sorozatoknak (trains).

A Cisco IOS sorozatok mindig egy közös kódbázis adott platformra és képességhalmazra vonatkozó kiadásait tartalmazzák. Így tehát egy adott sorozat több kiadásból állhat, melyek mindegyike az adott közös kódbázis - kiadás pillanatában érvényes - pillanatfelvétele. Mivel az egyes szoftver családok különböző platformokra, vagy piaci szegmensekre alkalmazhatóak, így egy időben több érvényes sorozat is lehet.

A fejezet részletesen tárgyalja az IOS 12.4 és a 15-ös sorozat jellegzetességeit.

# Cisco IOS 12.4 fővonalbeli és T sorozat

**12.4 sorozat**

Az ábrán a 12.3 és a 12.4 kiadás közti váltás látható. Egy szoftver kiadási családon belül több, szorosan kapcsolódó, aktív sorozat is lehet. A Cisco IOS szoftver 12.4 kiadási családjában például 2 sorozat található: a 12.4 fővonalbeli sorozat és a 12.4T sorozat.

A jelölés nélküli Cisco IOS 12.4 sorozat számít fővonalbeli sorozatnak. A fővonalbeli sorozathoz általában csak - a szoftver minőségét emelő - hibajavítási frissítéseket adnak ki. A fővonalbeli sorozat kiadásait szoktuk MD kiadásnak is nevezni (Maintenance Deployment release - karbantartási kiadás).

A fővonalbeli sorozathoz mindig kiadnak egy technológiai sorozatot (T train) is. A T sorozat, mint például a 12.4T, megkap minden fővonalbeli hibajavítást, ugyanakkor ehhez adják ki az új hardver- és szoftverösszetevőket támogató frissítéseket is. A Cisco IOS 12.4T sorozatba tartozó kiadásokat szokák ED kiadásnak (Early Deployment - korai kiadás) is nevezni.

A kiadási családtól függően lehetnek további sorozatok is. Ilyen további sorozat jellemzően a szolgáltatói sorozat (Service provider train - S train) is. Az S sorozat jellemzően a hálózati szolgáltatók igényeihez kialakított szolgáltatásokat tartalmaz.

Az adott fővonalbeli sorozathoz tartozó leszármazott sorozatokat általában a nevükben található nagybetűvel írt típusazonosítóról (T, S stb.) lehet felismerni.

Fővonalbeli sorozat = 12.4

T sorozat = 12.4T (12.4 verzió + új szoftver- és hardverösszetevőket támogató tulajdonságok)

A Cisco IOS korábbi változatai esetében, egészen a 12.4 családdal bezárólag, a fővonalbeli és a T sorozat szigorúan különvált egymástól. Másként fogalmazva a fővonalhoz képest a T sorozat markánsan eltért, olyan egyedi kódbázissal rendelkezett, amely képes volt az új hardver és szoftverösszetevőket támogatni. Lényegében az újabb fővonalbeli sorozat mindig egy kiadott T sorozatból lett kialakítva, és a ciklus kezdődött elölről. A sorozatok ezen kiadási logikáját változtatta meg a Cisco IOS 15-ös verzió.

Az ábrán a Cisco IOS 12.4 fővonalbeli sorozatának és a kapcsolódó 12.4T sorozatnak a viszonya látható.

# Cisco IOS 12.4 fővonalbeli és T sorozatának számozása

Az IOS kiadások számozása úgy van kialakítva, hogy könnyen felismerhető legyen a verzió beleértve a hibajavításokat és az új szoftver tulajdonságokat is. Az ábrán a fővonalbeli és a T sorozat számozási sémája látható konkrét példán keresztül.

* A fővonalbeli szoftververzió számozási sémája a sorozatszámból (train number), a karbantartási azonosítóból (maintenance identifier) és a fordítási azonosítóból (rebuild identifier) áll. A 12.4(21a) Cisco IOS kiadás, például egy fővonalbeli sorozathoz tartozik. A T sorozat kiadásainak számozása a sorozatszámból (train number), a karbantartási azonosítóból (maintenance identifier), a sorozatazonosítóból (train identifier) és a fordítási azonosítóból (rebuild identifier) áll. A 12.4(20)T1 Cisco IOS kiadás, például a 12.4T sorozathoz tartozik.
* Minden egyes karbantartási azonosító a Cisco IOS 12.4 fővonalbeli sorozat új szoftverkomponensekkel bővített vagy hibajavításokat tartalmazó verzióját jelöli, például 12.4(7). Ezt a váltást mindig a zárójelben szereplő szám jelöli. Az összes azonos (például Cisco IOS 12.4(20)T) karbantartási kiadáshoz tartozó verziója a 12.4T sorozatnak ugyanazokat a szoftverjavításokat, új komponenseket és hardvertámogatásokat tartalmazza.
* A Cisco újrafordításokkal oldja meg a lényeges problémák javításainak integrálását a kiadásokba. Ezáltal csökkenthető a kockázata a váltásoknak azon ügyfelekre, akik már üzembe helyeztek egy tanúsított szoftverkiadást. Az egyes újrafordítások általában korlátozott számú szoftverhiba (úgynevezett kifogásolt rész - caveat) javítását tartalmazzák. Az újrafordításokat fővonalbeli sorozatok esetén a zárójelen belüli kisbetűk jelölik, más sorozatok esetén pedig az utolsó szám. Például a 12.4(21) verzióhoz kiadtak egy újrafordítást néhány kifogásolt rész javítására, ez a verzió a 12.4(21a) számot viseli. Hasonlóképpen a 12.4(15)T8 verzió a 12.4(15)T nyolcadik újrafordítása. Minden egyes újrafordítás eggyel növeli a fordítási azonosítót és lehetővé teszi, hogy az ütemezett kiadások előtt gyors javításokat adjanak ki. Az újrafordítások módosításai szigorú szabályokhoz kötöttek.

A 12.4 Cisco IOS sorozaton belül minden egyes kiadást egyedi szám azonosít. A 12.4 családon belül az egyes kiadási azonosítókat egyaránt következetesen alkalmazzák a 12.4 karbantartási kiadáson és a 12.4 T sorozaton belül. A 12.4(6)T kiadást a 12.4(7)T követte, majd azt a 12.4(8)T. Ez teszi lehetővé, hogy a rendszergazdák követni tudják az egyes kódváltozatokban bevezetett újításokat.

**MEGJEGYZÉS**: Minden T sorozatba beépített javításnak be kell kerülnie a következő fővonalbeli kiadásba.

# Cisco IOS 12.4 rendszer képfájl csomagok

A 15.0 IOS szoftververzió előtt a forgalomirányítókhoz az ábrán látható nyolc különböző csomag közül lehetett a megfelelőt kiválasztani. Ezt a csomagsémát a 12.3 fővonalbeli változattal vezették be, és azóta használták más sorozatoknál is. A nyolc rendelkezésre álló IOS csomag közül három számít prémium csomagnak.

Az öt nem prémium csomag:

* **IP Base - IP alap** - az IP Base a belépő szintű Cisco IOS képfájl
* **IP Voice - IP hang** - adatok és hangok együttes továbbítása, VoIP, VoFR és IP telefon
* **Advanced Security - haladó biztonság** - biztonsági és VPN funkciók, mint Cisco IOS tűzfal, IDS/IPS, IPSec, 3DES és VPN
* **SP (Service Provider)- hálózati szolgáltatói változat** - SSH/SSL, ATM, VoATM és MPLS to IP Voice elemekkel bővített változat
* **Enterprise Base - nagyvállalati alap** - nagyvállalati protokollok: Appletalk, IPX, és IBM támogatás

**MEGJEGYZÉS**: A Cisco IOS 12.4 változattól kezdődően minden csomagban elérhető az SSH.

A három további prémium csomag összetettebb hálózati elvárásoknak megfelelő funkciókkal gazdagítja a lehetőségeket. Az Advanced Enterprise Services (haladó nagyvállalati szolgáltatások) csomag egyesíti az összes funkciót. Ebben a csomagban megtalálható az összes hangátvitelhez, biztonsághoz és VPN-hez kapcsolódó protokoll támogatása.

* **Advanced Enterprise Services - Haladó nagyvállalati szolgáltatások** - az összes Cisco IOS szoftver funkció
* **Nagyvállalati szolgáltatások** - az Enterprise Base és az SP verziók kombinációja
* **Fejlettebb IP szolgáltatások** - az Advanced Security és az SP csomagok kombinációja IPv6 támogatással

**MEGJEGYZÉS**: a Cisco Feature Navigator (Cisco funkciókereső) eszköz segítségével könnyen kiválasztható a szükséges tulajdonságokkal rendelkező Cisco operációs rendszer verzió.

# Cisco IOS 15.0 M és T sorozatok

A Cisco IOS 12.4(24)T kiadás után a 15.0 verzió következett.

Az IOS 15.0 számos újítást vezetett be:

* Új funkciók és hardverek támogatása
* Más IOS kiadásokkal összehangoltabb működés
* Kiszámíthatóbbá vált az új funkciók kiadásának és az újrafordításoknak az ütemezése.
* Proaktív egyedi kiadástámogatási eljárásrend
* Egyszerűsített kiadásszámozási séma
* Egyértelmű szoftver alkalmazási és áttelepítési irányelvek

Az ábrán látható módon a Cisco IOS 15.0 új kiadási logikát követ a korábbi 12.4-nél alkalmazott elkülönült fővonalbeli és T sorozatos kiadási logika helyett. A Cisco IOS 15.0 ahelyett, hogy külön sorozatokban adná ki a fővonalbeli és T verziókat, két kiterjesztett karbantartási (EM extended maintenance) kiadás között sztenderd karbantartási kiadások (T release - standard maintenance release) formájában jelenteti meg a frissítéseket. A Cisco IOS 15 kiadási modelljében a fővonalbeli kiadásokat M sorozatoknak (M trains) hívjuk.

A 15.0 verziótól kezdődően az új kiadások T sorozatbeli formában, kiszámíthatóan 2-3 alkalommal jelennek meg évente. Az EM kiadások 16-20 havonta várhatóak. A T sorozatbeli kiadások teszik lehetővé, hogy a következő esedékes EM kiadás előtt gyorsan lehessen új Cisco funkciókat beépíteni.

Az EM kiadás minden esetben magába foglalja az összes előtte kiadott T kiadás hardvertámogatási és szoftveres újításait. Ezáltal az EM verzió kiadásának pillanatában az adott sorozat elérhető teljes funkcionalitását tartalmazza.

Összegezve az új Cisco IOS kiadási modell előnyeit:

* - örökölte a Cisco IOS 12.4T és 12.4 fővonalbeli kiadások elérhető képességeit
* - az évente 2-3 alkalommal kiadásra kerülő frissítések egy sorozatot alkotnak
* - 16-20 havonta tervezhetően megjelenik az új EM kiadás, mely tartalmazza a legújabb funkciókat
* - az esedékes EM kiadás előtt a legfrissebb újdonságokat és hardvertámogatást tartalmazó T kiadások a Cisco.com weboldalon érhetők el
* - az M és T sorozatok karbantartási újrafordításai mindig csak hibajavításokat tartalmaznak

# A Cisco IOS 15 sorozat számozási sémája

Az IOS 15 számozása - hasonlóan a korábbi sorozatokhoz - egyértelműen azonosítja az adott IOS kiadást, legyen szó hibajavításról vagy új funkciók bevezetéséről. Az ábrán 1-1 példa látható a séma alkalmazására M és T sorozat esetében.

**Kiterjesztett karbantartási kiadás (Extended Maintenance Release - ME)**

Az EM kiadás lehetővé teszi, hogy az ügyfelek hosszú távon tervezhessenek, kipróbált verziót alkalmazhassanak kiszámítható időtávon. A fővonalbeli sorozat elérhetővé teszi a korábbi kiadások funkcionalitását, miközben inkrementálisan beépülnek az új funkciók és az időközben megjelent hardverösszetevők támogatása.

A 15.0(1)M kiadás első karbantartási újrafordítása (maintenance rebuild), ami csak hibajavítást tartalmazhat, nem lehet benne új funkció, sem új hardvertámogatás, a 15.0(1)M1 sorszámot viseli. A következő karbantartási kiadások egyesével növelik a sorszám végén található újrafordítási azonosítót (M2, M3 és így tovább).

**Sztenderd karbantartási kiadás (Standard Maintenance Release)**

A T kiadás hivatott arra, hogy az esedékes EM kiadásig új funkcionalitás és hardvertámogatás beépítését tegye lehetővé . A T kiadások tartalmazzák a rendszeres hibajavítást tartalmazó karbantartási újrafordításokat, valamint az olyan hálózatot veszélyeztető kritikus hibák javításait, mint a termékbiztonsági eseményfigyelő csapat (PSIRT - Product Security Incident Report Team) által azonosított esetek javításait.

Az első menetrend szerint kiadott, új funkcionalitást biztosító 15 T kiadás a 15.1(1)T sorszámot viseli. A 15.1(1)T kiadást követő első karbantartási újrafordítás (amely csak hibajavításokat tartalmaz, új funkciókat és új hardverelemek támogatását nem) a 15.1(1)T1 sorszámot kapja. A következő kiadások egyesével növelik a sorszám végén található újrafordítási azonosítót (T2, T3 és így tovább).

# IOS 15 rendszer képfájl csomagok kialakítása

Az 1900, 2900 és 3900 sorozathoz tartozó második generációs, integrált szolgáltatású forgalomirányítók (ISR G2) a megfelelő szoftverlicencek használatával tudják biztosítani a szükséges szolgáltatásokat. Az igény szerinti szolgáltatás koncepció (Services on Demand) megkönnyíti az ügyfelek számára a szoftvervásárlás és menedzselés megvalósítását. Minden új ISR G2 eszköz ugyanazzal az univerzális Cisco IOS képfájllal érkezik, melyen az egyes funkciócsoportok a megfelelő licenc alkalmazásával engedélyezhetők (lásd 1. ábra).

Az ISR G2 termékcsalád alapvetően kétféle univerzális képfájltípust támogat:

* **A fájlnévben "universalk9" elnevezést tartalmazó univerzális képfájlokat** - Ez az univerzális képfájl tartalmazza a Cisco IOS szoftver összes képességét beleértve az erős forgalomtitkosítási képességet, mint az IPsec VPN, SSL VPN és Biztonságos Egységesített Kommunikáció (Secure Unified Communications).
* **A fájlnévben "universalk9\_npe" elnevezést tartalmazó univerzális képfájlokat** - a Cisco Software Activation rendszer által továbbfejlesztett titkosítási képességek kielégítik a titkosítási képességek exportja iránti igényeket. Ugyanakkor egyes országok importszabályai előírják, hogy a hardverelemek ne tartalmazzanak erős titkosítási képességeket, mint amilyen a forgalomtitkosítás. Az ilyen importszabályok kielégítése érdekében az npe univerzális képfájlok egyáltalán nem támogatják a forgalom erős titkosítását.

Az ISR G2 eszközökkel lényegesen egyszerűbbé vált a megfelelő IOS képfájl kiválasztása, hiszen az univerzális képfájlok minden funkciót tartalmaznak. Az egyes funkciók a megfelelő licenc használatával aktiválhatók. Minden eszköz az univerzális képfájllal érkezik. Az univerzális képfájlon belül az egyes technológiai csomagok (IP Base, Data, UC - Unified Communication és SEC - Security) a megfelelő Cisco szoftver aktiváló licenc kulccsal engedélyezhetők. A licenckulcs az adott eszközre nézve egyedi. A kulcsot a forgalomirányító termékazonosítójának, illetve a termék aktiválási kulcsnak (PAK - Product Activation Key) a megadásával lehet a Cisco-tól igényelni. A termékaktiválási kulcsot (PAK) a Cisco a szoftver vásárlásakor adja. Alapértelmezésben az IP Base csomag kerül telepítésre.

A 2. ábrán látható, hogy milyen IOS 15 (Simplified Packiging) csomagra érdemes váltani az IOS 12 (IOS Reformation Packiging) után a következő generációs ISR forgalomirányító beszerzése esetén.

**IOS képfájlok fájlnevei**

Fontos a megfelelő képességeket nyújtó IOS verzió kiválasztása, ha új forgalomirányítót választunk, vagy a jelenlegit fejlesztjük. A Cisco IOS képfájlok egységes elnevezési sémát követnek. A Cisco IOS képfájlok állományneve több részből áll, melyek mindegyikének meg van a maga jelentése. A megfelelő IOS szoftver kiválasztása érdekében fontos, hogy tisztában legyünk ezzel a nevezéktannal.

Ahogy azt az 1. ábra is mutatja, a **show flash** parancs a flash memória tartalmát listázza ki, beleértve az Cisco IOS fájlnevét is.

A 2. ábrán egy tipikus 12.4 verziójú IOS állománynév látható.

* **Képfájl neve (c2800nm)** - Azonosítja a platformot, amin a képfájl fut. Ebben a példában ez egy hálózati modult tartalmazó Cisco 2800-as sorozatú forgalomirányító.
* **advipservicesk9** - Azonosítja a forgalomirányító által támogatott képességek halmazát. Ebben a példában az advipservicesk9 a továbbfejlesztett IP szolgáltatások halmazára utal, mely tartalmazza mind a továbbfejlesztett biztonsági csomagot, a szolgáltatói csomagot, valamint az IPv6 támogatást.
* **mz** - Ebből a részből látszik, hogy hol fut a képfájl, illetve, hogy a képfájl tömörített-e. Ebben a példában az mz azt jelenti, hogy a képfájl a RAM-ból fut és tömörített.
* **124-6.T** - A képfájl kiadásának azonosítója fájlnév alakban: 12.4(6)T. Azaz a sorozatszám (train number), a karbantartási kiadás száma (maintenance release number), végül a sorozat azonosítója (train identifier).
* **bin** - A fájl kiterjesztése. A kiterjesztésből látszik, hogy ez egy futtatható, bináris állomány.

A 3. ábrán egy ISR G2 forgalomirányítón futó IOS 15 rendszerfájl nevének egyes tagjai láthatók.

* **Képfájl neve (c1900)** - Azonosítja a platformot, amin a képfájl fut. Ebben a példában ez egy Cisco 1900-as sorozatú forgalomirányító.
* **universalk9** - Ez a rész azonosítja, hogy milyen piacra szánt verzióval van dolgunk. Az ISR G2 forgalomirányítók alapvetően kétféle csomaggal érkezhetnek: universalk9 vagy universalk9\_npe. Az universalk9\_npe csomag nem tartalmaz erős titkosítást, olyan országokba szánják, ahol a titkosítás használata szigorúan szabályozott. Az elérhető funkciókat a licencek szabályozzák. Ez alapján négy technológiai csomagot különböztetünk meg: IP Base, Security, Unified Communications és Data.
* **mz** - Ebből a részből látszik, hogy hol fut a képfájl, illetve, hogy a képfájl tömörített-e. Ebben a példában az mz azt jelenti, hogy a képfájl a RAM-ból fut és tömörített.
* **SPA** - A fájl rendelkezik a Cisco digitális aláírásával.
* **152-4.M3** - A képfájl azonosítójának - 15.2(4)M3 - fájlnévben használatos alakja. Ez az IOS verziószáma, ami tartalmazza a főverzió számát (major release), az alverzió számát (minor release) valamint a karbantartási újrafordítás számát (maintenance rebuild). Az M jelzi, hogy ez egy kiterjesztett karbantartási kiadás (EM).
* **bin** - A fájl kiterjesztése. A kiterjesztésből látszik, hogy ez egy futtatható, bináris állomány.

A futtatási hely és tömörítés legjellemzőbb formája az mz alak. Ezen a tagon belül az első karakter jelöli a képfájl futtatási helyét, ami lehet:

* **f** - flash memória
* **m** - RAM
* **r** - ROM
* **l** - áthelyezhető.

A tömörítés típusa kétféle lehet: a z jelöli a zip tömörítést, az x az mzip tömörítést. A Cisco zip tömörítést alkalmaz egyes RAM-ból futtatható képfájloknál, ezáltal csökkentve a képfájl méretét. Ez alapvetően önkicsomagoló formát jelent, vagyis a képfájl először kicsomagolásra kerül, mielőtt betöltődne futtatásra a RAM-ba.

**MEGJEGYZÉS**: A Cisco fenntartja a jogot az elnevezési séma, az egyes részek jelentése, a képfájlok tartalma és további részletek megváltoztatására.

**Memóriakövetelmények**

A legtöbb Cisco forgalomirányítón, beleértve az integrált szolgáltatású forgalomirányítókat is, az IOS-t a flash memória tárolja, onnan kerül betöltésre a DRAM-ba a rendszerindítás során. A Cisco 1900 és 2900 sorozatú ISR forgalomirányítókhoz elérhető Cisco IOS Software Release 15.0 képfájlok memóriaigénye 256 MB flash és 512 MB RAM. A 3900 sorozatú ISR forgalomirányítók esetén ez 256 MB flash memória és 1 GB RAM. Ebbe nem számít bele a további menedzsment eszközök (mint a Cisco Configuration Professional - Cisco CP) memóriaigénye. A pontos részletek megtalálhatók az adott forgalomirányító adatlapjában.

# Biztonsági mentés TFTP kiszolgálóra

Amikor egy hálózat egyre nagyobbá válik, érdemes lehet a Cisco IOS képfájlokat és a konfigurációs fájlokat egy központi TFTP kiszolgálón tárolni. Ezáltal kordában tartható a szükséges konfigurációs fájloknak, IOS képfájloknak és azok frissítéseinek darabszáma.

Az összekapcsolt hálózatok általában nagy területeket fednek le, és számos forgalomirányítót tartalmaznak. Függetlenül a hálózat jellegétől érdemes a Cisco IOS képfájlokból biztonsági másolatot készíteni arra az esetre, ha valamilyen okból megsérül, vagy törlődik a forgalomirányítón futó rendszerfájl.

Ezeknek a földrajzilag elosztott forgalomirányítóknak szükségük van egy forrás- vagy tartalékhelyre, ahol a szoftverkódot tárolják. TFTP kiszolgálóval lehetővé válik a képfájlok és konfigurációs fájlok fel- és letöltése a hálózaton keresztül. A TFTP kiszolgáló lehet egy másik forgalomirányító, egy munkaállomás vagy egy kiszolgáló.

# Cisco IOS képfájl biztonsági mentése

A rendszerleállási idő minimalizálása érdekében fontos, hogy kidolgozott eljárásrendje legyen a Cisco IOS képfájlok helyreállításának. Így a hálózati rendszergazda gyorsan fel tudja tölteni a forgalomirányítóra a rendszerfájl biztonsági másolatát, ha az megsérül, vagy törlődik.

Az 1. ábrán látható példában a rendszergazda biztonsági másolatot szeretne készíteni az aktuális rendszerfájlról (c1900-universalk9-mz.SPA.152-4.M3.bin) a 172.16.1.100 címen elérhető TFTP kiszolgálóra.

A következő három lépés végrehajtásával készíthetünk biztonsági másolatot az IOS rendszerfájlról egy TFTP kiszolgálóra:

**1. lépés** Ellenőrizzük a TFTP kiszolgáló hálózati elérhetőségét. A TFTP kiszolgáló elérhetőségét ping segítségével ellenőrizzük, ahogy az a 2. ábrán látható.

**2. lépés** Ellenőrizzük, hogy van-e elegendő tárhely a TFTP kiszolgálón a Cisco IOS képfájl számára. Használjuk a **show flash0:** parancsot a Cisco IOS képfájl méretének meghatározására. A példában szereplő képfájl mérete 68831808 bájt.

**3. lépés** Másoljuk fel a rendszerfájlt a TFTP kiszolgálóra a **copy** *forrás-url cél-url* paranccsal, ahogy az a 3. ábrán látható.

Miután kiadtuk a parancsot az ábrán látható forrás és cél url-ek használatával, a rendszer kérni fogja a forrásállomány nevét, a célhely IP-címét és a célfájl nevét, ebben a sorrendben. Ezután elkezdődik az adatátvitel.

Használjuk a parancsszimulátort a 4. ábrán, hogy az R2 forgalomirányítóról TFTP kiszolgálóra másoljuk az IOS fájlt!

# Cisco IOS képfájl visszatöltése

A Cisco IOS szoftver folyamatos frissítései biztosítják a hibajavításokat és az új funkciók beépítését. A példában IPv6 környezetet használunk az átvitel során, bemutatva, hogy a TFTP IPv6 hálózatokban is használható.

Az 1. ábrán látható a Cisco IOS képfájl visszatöltése egy TFTP kiszolgálóról. A 2001:DB8:CAFE:100::99 címen elérhető TFTP kiszolgálóról töltünk fel egy új képfájlt (c1900-universalk9-mz.SPA.152-4.M3.bin) a forgalomirányítóra.

A forgalomirányítón futó szoftver frissítéséhez kövessük az alábbi lépéseket:

**1. lépés** Válasszuk ki a hardverkörnyezetnek, funkcionális és szoftveres elvárásoknak megfelelő Cisco IOS képfájlt. Töltsük le a megfelelő képfájlt a cisco.com weboldalról, és töltsük fel a TFTP kiszolgálónkra.

**2. lépés** A TFTP kiszolgáló elérhetőségének a vizsgálata. Ping segítségével ellenőrizzük, hogy elérhető-e a TFTP kiszolgáló a forgalomirányítóról. A 2. ábrán megjelenő parancskimenetből látszik, hogy a TFTP kiszolgáló elérhető.

**3. lépés** Ellenőrizzük, hogy a frissítendő forgalomirányítón rendelkezésre áll-e a megfelelő mennyiségű tárhely a flash memóriában. A szabad flash memória mérete a **show flash0:** paranccsal ellenőrizhető. Hasonlítsuk össze a szabad tárhely méretét az új képfájl méretével. A 3. ábrán a **show flash0:** paranccsal ellenőrizzük a szabad flash tárhelyet. A példában 182 394 880 bájtnyi szabad hely van a flash memóriában.

**4. lépés:** Másoljuk a TFTP kiszolgálóról a forgalomirányítóra az IOS képfájlt a 4. ábrán látható **copy** paranccsal. Miután kiadtuk a parancsot az ábrán látható forrás és cél url-ek használatával, a rendszer kérni fogja a távoli kiszolgáló IP-címét, a forrásállomány nevét és a célfájl nevét, ebben a sorrendben. Ezután elkezdődik az adatátvitel.

**A rendszer újraindítása**

Miután feltöltöttük az új IOS képfájlt a forgalomirányító flash memóriájába, még módosítani kell az alábbiak szerint a konfigurációt, hogy következő rendszerindításkor betöltődjön a **boot system** paranccsal. Mentsük el a konfigurációt! Indítsuk újra a forgalomirányítót, hogy betöltődjön az új rendszerfájl. Az újraindítást követően ellenőrizzük, hogy valóban az új képfájl töltődött be a **show version** paranccsal!

A rendszerindítás során a rendszerindító program az NVRAM-ban tárolt konfigurációs fájlban a **boot system** utasításokat keresi, mivel ezek határozzák meg, hogy honnan és milyen nevű Cisco IOS képfájlt kell betölteni. Több **boot system** parancsot is megadhatunk, így kialakítható egy hibatűrő rendszerindítási terv.

Az 1. ábrán látható **boot system** parancs egy globális konfigurációs parancs, mely lehetővé teszi a rendszerindítás során betöltendő Cisco IOS képfájl helyének meghatározását. Néhány az elérhető konfigurációs lehetőségek közül:

* Beállítható, hogy a flash memória legyen a Cisco IOS képfájl forrása.

Router(config)# **boot system flash0://c1900-universalk9-mz.SPA.152-4.M3.bin**

* Beállítható, hogy egy TFTP kiszolgáló legyen a Cisco IOS képfájl forrása.

Router(config)# **boot system tftp://c1900-universalk9-mz.SPA.152-4.M3.bin**

Amennyiben a konfigurációs fájlban nem szerepel **boot system** parancs, akkor alapértelmezésként a flash memóriában található első érvényes Cisco IOS állomány töltődik be és kerül futtatásra.

A 2. ábrán látható **show version** paranccsal ellenőrizhetjük, hogy melyik rendszerfájl került betöltésre.

**Licencelés - áttekintés**

A Cisco IOS 15.0 kiadással megváltozott a technológiai újítások IOS funkciók közé beépítésének folyamata. A Cisco IOS 15.0 kiadás platformközi halmazokba rendezetten nyújtja az elérhető funkciókat, ezáltal könnyebbé vált a megfelelő képfájl kiválasztása. A platformhatárok elmosódnak a hasonló elérhető funkcióknak köszönhetően. Minden eszköz egy univerzális képfájllal érkezik. Az univerzális képfájlon belül az egyes technológiai csomagok a megfelelő Cisco szoftver aktiváló licenc kulccsal engedélyezhetők. A felhasználók a Cisco IOS szoftveraktivációs funkción keresztül tudják engedélyezni a licencelt funkciókat és tudnak újabb licenceket regisztrálni. A Cisco IOS szoftveraktiváció lényegében olyan eljárások és komponensek gyűjteménye, melyekkel Cisco szoftverlicenceket szerezhetünk be és érvényesíthetünk, s így aktiválhatjuk a megfelelő Cisco IOS funkciócsomagokat.

Az 1. ábrán láthatók az elérhető technológiai csomagok:

* IP Base - IP alapkészlet
* Data - Adat
* Unified Communications (UC) - Egységesített kommunikáció
* Security (SEC) - Biztonság

A 2. ábrán lévő gombokra kattintva részletes leírást kaphatunk az egyes technológiai csomagokról.

**MEGJEGYZÉS**: Az IP Base csomag szükséges a Data, Security és UC licencek telepítéséhez. Nem érhetők el univerzális képfájlok a Cisco IOS 15.0 verziót támogató korábbi forgalomirányító típusokhoz. Ezekhez a forgalomirányítókhoz külön képfájlok formájában kell a megfelelő szolgáltatáskészletet tartalmazó állományt letölteni.

**Technológiai csomag licencek**

A technológiai csomag licenceket a Cisco ISR G2 platformok (Cisco 1900, 2900 és 3900 sorozat forgalomirányítói) támogatják. Az univerzális Cisco IOS képfájlok egyben tartalmazzák az összes csomagot. Minden egyes csomag egy adott technológiához kapcsolódó funkciók gyűjteménye. A Cisco 1900, 2900 és 3900 sorozatú ISR platformokon egyszerre több technológiai csomag is aktiválható.

**MEGJEGYZÉS**: A **show license feature** paranccsal ellenőrizhetjük, hogy a forgalomirányítón melyik technológiai csomagok és funkciócsomagok támogatottak.

# A licencelés folyamata

Amikor új forgalomirányítót vásárolunk, akkor az a kiválasztott és megvásárolt csomagoknak és funkcióknak megfelelő képfájlt és az állandó licenceket előtelepítve tartalmazza.

Mindemellett az adott forgalomirányító által támogatott további csomagok és funkciók ideiglenes (kipróbálási) licencei is megtalálhatók rajta. Így az ügyfelek a megfelelő próbalicenc aktiválásával új szoftvercsomagokat és funkciókat is kipróbálhatnak. Amennyiben az ügyfelek szeretnék valamelyik funkciót véglegesen aktiválni, akkor meg kell vásárolniuk a hozzá tartozó új licencet.

Az ábrán egy új szoftvercsomag vagy funkció engedélyezésének 3 lépéses folyamata látható.

# 1. lépés A telepítendő szoftvercsomag vagy funkció megvásárlása.

**1. lépés** **A telepítendő szoftvercsomag vagy funkció megvásárlása.**

Első lépésként meg kell vásárolnunk azt a szoftvercsomagot vagy funkciót, amire szükségünk van. Ez azzal járhat, hogy az IP Base csomag mellé meg kell vásárolni egy másik csomagot, mondjuk a Security csomagot.

A szoftveraktiválást igénylő licencek esetében szoftverigény tanúsítványt (Software Claim Certificates) használ a Cisco. Ezen a tanúsítványon található meg a licenc termék aktivációs kulcsa (Product Activation Key - PAK), valamint a végfelhasználói licencszerződéssel (Cisco End User License Agreement - EULA) kapcsolatos minden fontos információ. Az esetek többségében a vásárlás pillanatában már a Cisco vagy valamelyik Cisco viszonteladó már aktiválta a megrendelt licenceket, ezért szoftverigény tanúsítvány már nem kerül kiállításra.

Az ügyfél mindkét esetben megkapja a PAK kulcsot a vásárláshoz. A PAK egyben nyugtaként is szolgál, és azzal lehet megszerezni a licencet. A PAK egy 11 karakterből álló alfanumerikus kulcs, melyet a Cisco állít elő. A PAK mindig egyértelműen meghatározza a hozzá tartozó funkciókészletet. A PAK mindaddig nem kötődik egy adott eszközhöz, amíg a licencet létre nem hozták. A PAK megvásárlásakor meghatározható, hogy hány licenc létrehozásához lehet felhasználni. Ahogy az az ábrán látható, minden egyes csomag (IP Base, Data, UC és SEC) külön licencet igényel.

# 2. lépés Licenc beszerzése

**2. lépés** **Licenc beszerzése**

A következő lépés a licenc beszerzése, ami igazából nem más, mint egy licencfájl. A licencfájl, más néven a szoftver aktivációs licenc, beszerzésére több lehetőségünk is van:

* **Cisco License Manager (CLM) - Cisco licencmenedzser** - Ez egy ingyenesen letölthető alkalmazás, mely a <http://www.cisco.com/go/clm>weboldalon érhető el. A CLM egy olyan Cisco fejlesztésű önálló alkalmazás, mellyel a hálózati rendszergazdák gyorsan tudnak egyszerre több Cisco licencet is aktiválni a hálózat eszközein. A Cisco License Manager alkalmazás képes felderíteni a hálózati eszközöket, le tudja kérdezni a licencinformációikat és segít az új licencek beszerzésében és alkalmazásában. Az alkalmazás grafikus felülete leegyszerűsíti és segít automatizálni a licencek beszerzési folyamatát, továbbá lehetővé teszi a többszörös licencelési folyamatok központosítását. A CLM ingyenesen letölthető a CCO-ról.
* **Cisco License Registration Portal - Cisco licenc regisztrációs portál** - Ez az a webes felület, ahol egyedi szoftver licenceket vásárolhatunk és regisztrálhatunk. <http://www.cisco.com/go/license>.

Mindkét esetben szükségünk lesz a PAK kulcsra és az egyedi eszközazonosítóra (Unique Device Identifier - UDI).

A PAK kulcsot vásárláskor kapjuk meg.

Az UDI a termékkulcs (Product ID - PID), a sorozatszám (Serial Number - SN) és a hardververzió kombinációja. Az SN egy 11 számjegyes azonosító, mely egyértelműen azonosít egy eszközt. A PID az eszköz típusát azonosítja. A licenc létrehozásához csak a PID és az SN szükséges. Az UDI megjeleníthető a **show license udi** paranccsal, lásd. 1. ábra. Ugyanez az információ megtalálható az eszköz hátulján kihúzható címketartón is. A 2. ábrán egy Cisco 1941-es forgalomirányító kihúzható címketartója látható.

A megfelelő adatok megadását követően az ügyfél elektronikus levélben kapja meg a licencfájl telepítéséhez szükséges információkat. A licence fájl egy egyszerű szöveges xml állomány .lic kiterjesztéssel.

Használjuk a 3. ábrán lévő parancsszimulátort az R2 forgalomirányító UDI azonosítójának meghatározására.

# 3. lépés A licenc telepítése

**3. lépés A licenc telepítése**

A licenc megvásárlását követően az ügyfél megkapja a licencfájlt. A végleges licenc telepítése 2 lépésből áll:

**1. lépés** A **license install** *stored-location-url* privilegizált módbeli paranccsal telepíthető a licencfájl.

**2. lépés** A forgalomirányítót ezután újra kell indítani a privilegizált módbeli **reload**paranccsal. Amennyiben volt aktív próbalicenc, akkor nem kell újraindítani a forgalomirányítót.

Az 1. ábrán a Security csomag végleges licencének telepítése folyamata látható.

**MEGJEGYZÉS**: Az 1941-es sorozatú forgalomirányítók nem támogatják a Unified Communications csomagot.

A végleges licencek nem járnak le. Ha egyszer telepítettünk egy végleges licencet, az a forgalomirányító teljes életciklusában elérhetővé teszi az adott funkciókat, akkor is, ha új IOS verziót telepítünk. Például, ha telepítettük egy forgalomirányítón az UC, a SEC vagy a Data licencet, a megfelelő funkciók akkor is elérhetők lesznek, ha új IOS verziót telepítünk. A végleges licenc a leggyakrabban alkalmazott licencfajta.

**MEGJEGYZÉS**: A gyártó előtelepíti a szükséges végleges licenceket a megrendelt eszközre a megvásárolt funkciókészletnek megfelelően. Az új hardveren a vásárlónak nincs további teendője a Cisco IOS szoftveraktivációs folyamatban.

Használjuk a 2. ábrán lévő parancsszimulátort az R2 forgalomirányítón a végleges licenc telepítéséhez.

**Licencek ellenőrzése**

Új licenc telepítése után újra kell indítani a forgalomirányítót a **reload** paranccsal. Ahogy azt az 1. ábra is mutatja, a forgalomirányító újraindítása után a **show version** paranccsal ellenőrizhetjük, hogy sikeres volt-e a licenc telepítése.

A **show license** paranccsal, a 2. ábrán látható módon, további részleteket jeleníthetünk meg a Cisco IOS licencekről. A parancs kimenete igen hasznos lehet, ha Cisco IOS licencekkel kapcsolatos hibákat kell meghatározni. A kimenetben az összes telepített licenc információi megjelennek. A példában az IP Base és a Security licencek lettek telepítve. A parancs kimenetében megjelennek a még elérhető, de egyelőre nem telepített funkciókészletek is, mint a példában a Data csomag. A kimenet a licencek tárolása szerint csoportosított.

A kimenetben a következők láthatók:

* **Jellemző** - A funkciócsomag neve.
* **Licenctípus** - A licenc típusa, például végleges (permanent) vagy próba (evaluation).
* **Licenc állapot** - A licenc állapota, például aktív (Active) vagy használatban levő (in use).
* **Licencek száma** - A még felhasználható és a használatban levő licencek száma (amennyiben releváns). Ha ebben a mezőben a non-counted (nem számozott) áll, akkor korlátlan licencről van szó. Ha ebben a mezőben a non-counted (nem számozott) áll, akkor korlátlan licencről van szó.
* **Licenc prioritás** - A licenc prioritása, például magas (high) vagy alacsony (low).

**MEGJEGYZÉS**: A **show license** a parancs részletes leírása megtalálható a Cisco IOS 15 parancsútmutatóban (command reference guide).

**RTU licencek aktiválása**

A próbalicencek kezelését 3 alkalommal vizsgálták felül az ISR G2 eszközök esetében. A legutolsó felülvizsgálat szerint - mely a Cisco IOS 15.0(1)M6, 15.1(1)T4, 15.1(2)T4, 15.1(3)T2 és 15.1(4)M kiadások óta érvényes - a próbalicencek (Evaluation) 60 nap elteltével Evaluation Right-To-Use (RTU - azaz használati jogosultsági alapot biztosító) licencekké változnak. A próba (Evaluation) licencek 60 napos kipróbálási periódust tesznek lehetővé. A 60 nap elteltével automatikusan átalakulnak RTU licenccé. Az átalakult licencek elérhetők maradnak a rendszerben, de az ügyfélnek el kell fogadnia az EULA (végfelhasználói licencszerződés) feltételeit. Az EULA feltételei automatikusan érvényesek az összes Cisco IOS szoftverlicencre.

A **license accept end user agreement** globális paranccsal egy lépésben elfogadhatjuk az EULA feltételeit az összes Cisco IOS szoftvercsomagra és funkciókészletre. A parancs kiadása és az EULA elfogadása után az EULA automatikusan érvényes lesz az összes Cisco IOS szoftverünkre, ezért a rendszer a licencek telepítése során már nem fog többet az EULA elfogadására felszólítani.

A 1. ábrán az EULA egyszeri elfogadásának konfigurációja látható:

Router(config)# **license accept end user agreement**

Az 1. ábra második részében pedig, egy RTU licenc aktiválásához használható parancs látható:

Router# **license boot module** *modul-neve* **technology-package** *csomag-neve*

Az argumentumok helyén a **?** használatával lekérdezhetjük, hogy az adott forgalomirányítón melyik modulnevek és szoftvercsomagok érhetők el. A Cisco ISR G2 platformokon az alábbi technológiai csomagnevek használhatóak:

* ipbasek9 - IP Base csomag
* securityk9 - Security csomag
* datak9 - Data csomag
* uck9 - Unified Communications csomag (az 1900-as sorozat eszközein nem érhető el)

**MEGJEGYZÉS**: A szoftvercsomag aktiválásához ki kell adni a **reload** parancsot.

A próbalicencek időkorlátosak, és arra valók, hogy ki lehessen próbálni az egyes funkciókészleteket az új hardvereszközökön. A kipróbálási idő általában 60 nap.

Miután sikeresen telepítettünk egy licencet, újra kell indítani a forgalomirányítót a **reload** paranccsal. A **show license** paranccsal ellenőrizhetjük, hogy sikerült-e telepíteni a licencet.

A 3. ábrán található parancsszimulátor segítségével fogadjuk el az EULA feltételeit, majd aktiváljunk egy RTU licenccel rendelekező Data csomagot az 1900-as forgalomirányítón!

# Licencek biztonsági mentése

A **license save** paranccsal másolatot készíthetünk az eszközön levő összes licencről, és eltárolhatjuk azokat a tárhelynek megfelelő formátumban. A licencek biztonsági másolatát a **license install** paranccsal állíthatjuk vissza.

Az adott eszközön található licencek biztonsági mentéséhez adjuk ki az alábbi parancsot:

Router# **license save** *mentés-helye:lic-fájl-neve*

Majd a **show flash0:** paranccsal ellenőrizhetjük, hogy sikerült-e elmentenünk a licenceket (lásd 1. ábra).

A licencmásolat tárolási helye lehet egy mappa vagy egy URL, ami egy tetszőleges fájlrendszerre mutat. A **?** használatával lekérdezhetjük, hogy milyen tárolási helyeket támogat az adott rendszer.

Használjuk a 2. ábrán lévő parancsszimulátort az R2 forgalomirányítón telepített licenc mentéséhez!

**Licencek eltávolítása**

A Cisco 1900, 2900 vagy 3900 sorozat forgalomirányítóin az alábbi lépések végrehajtásával törölhetünk egy korábban telepített aktív végleges licencet:

**1. lépés: Technológiai csomag letiltása.**

* Tiltsuk le az aktív licencet az alábbi parancs kiadásával:

Router(config)# **license boot module** *modul-neve* **technology-package** *csomag-neve* **disable**

* Majd indítsuk újra a forgalomirányítót a **reload** paranccsal. A szoftvercsomag az újraindítást követően lesz inaktív.

**2. lépés: Licenc törlése**

* Töröljük a technológiai csomag licencét a licenctárból.

Router# **license clear** *jellemző-neve*

* Vonjuk vissza a **license boot module** *modul-neve* **technology-package** *csomag-neve* **disable** parancsot, mellyel letiltottuk az aktív licencet:

Router(config)# **no license boot module** *modul-neve* **technology-package** *csomag-neve* **disable**

**MEGJEGYZÉS**: Egyes licenceket, mint például a beépített (built-in) licenceket, nem lehet törölni. Csak az olyan licenceket lehet eltávolítani, amelyeket a **license install** paranccsal lettek telepítve. A próbalicenceket nem lehet eltávolítani.

Az 1. ábrán egy aktív licenc eltávolítása látható.

Használjuk a 2. ábrán lévő parancsszimulátort az R2 forgalomirányítón a Security licenc eltávolításához.

**Összefoglalás**

A Cisco IOS szoftvernek sok verziója van, például 12.3, 12.4, 15.0 vagy 15.1. Az egyes szoftver kiadások mindig új képességeket vagy hibajavításokat tartalmazó verziókat jelentenek.

A Cisco IOS 12.4 verzió, például tartalmazza a Cisco IOS 12.3T sorozat által bevezetett összes új funkciót és hardvertámogatást, valamint további hibajavításokat. A fővonalbeli kiadások (ezeket hívjuk karbantartási kiadásnak is) nevében nincs nagybetűs névrészlet, és öröklik az összes kisebb sorszámú T kiadásban megjelent Cisco IOS funkcionalitást és hardvertámogatást. A 12.4 verzióig (azt is beleértve) a fővonalbeli M sorozat mindig csak hibajavításokat tartalmazott. Ugyanekkor a T sorozat ,a hibajavítások mellett, új funkciókat és hardvertámogatást is tartalmazott. Azaz a 12.4T sorozat olyan funkcióbeli, technológiai és hardvertámogatási újításokat vezet be, melyek a fővonalbeli 12.4 Cisco IOS szoftverben nem érhetők el.

A Cisco IOS szoftver 15.0-ás változata ehelyett új stratégiát vezetett be. A Cisco IOS 15.0 kiadástól kezdve a szoftvercsalád nem válik szét M és T sorozatra, az M és a T kiadások innentől egy sorozatot alkotnak. Ennek megfelelően a Cisco IOS 15.0 család első tagja a 15.0(1)M, ahol az M azt jelöli, hogy ez egy kiterjesztett karbantartási (extended maintenance) kiadás. A kiterjesztett karbantartási kiadás legfőbb előnye, hogy hosszútávon lehet vele tervezni. Természetesen a Cisco IOS 15.0 szoftvercsalád elemei nem csupán kiterjesztett karbantartási kiadások lesznek. Várhatóan lesznek normál karbantartási (standard maintenance) kiadások is, melyek új funkciókat és hardvertámogatási elemeket vezetnek be. A normál karbantartási kiadások nevében szerepel a nagy T betű.

Fontos a megfelelő képességeket nyújtó IOS verzió kiválasztása, ha új forgalomirányítót választunk, vagy a jelenlegit fejlesztjük. A Cisco IOS képfájlok egységes elnevezési sémát követnek. A Cisco IOS képfájlok állományneve több részből áll, melyek mindegyikének meg van a maga jelentése. Például: c1900-universalk9-mz.SPA.152-4.M3.bin

A flash memória tartalmának frissítéséhez és ellenőrzéséhez használjuk a megfelelő parancsokat. A **show flash** parancs a flash memória tartalmát listázza ki, beleértve a rendszerfájl nevét is. Ugyanezzel a paranccsal ellenőrizhetjük a flash memóriában rendelkezésre álló szabad tárhely méretét is . A **boot system** parancs egy globális konfigurációs parancs, mely lehetővé teszi a rendszerindítás során betöltendő Cisco IOS képfájl helyének meghatározását.

TFTP kiszolgálóval lehetővé válik a képfájlok és konfigurációs fájlok fel- és letöltése a hálózaton keresztül. A TFTP kiszolgáló lehet egy másik forgalomirányító, egy munkaállomás vagy egy kiszolgáló.

A Cisco IOS 15.0 kiadással megváltozott a technológiai újdonságok IOS funkciók közé beépítésének folyamata. Minden eszköz egy univerzális képfájllal érkezik. Az univerzális képfájlon belül az egyes technológiai csomagok (IP Base, Data, UC és SEC) a megfelelő Cisco szoftver aktiváló licenc kulccsal engedélyezhetők. A licenckulcs az adott eszközre nézve egyedi. A kulcsot a forgalomirányító termékazonosítójának, illetve a termék aktiválási kulcsnak (PAK - Product Activation Key) a megadásával lehet a Cisco-tól igényelni.

A rendeléshez gyárilag előtelepített licenceket nem kell használatba vétel előtt külön aktiválni. Az IP Base csomag minden ISR G2 eszközön végleges licencként rendelkezésre áll. A további három technológiai csomag (Data, Security és Unified Communications) alapértelmezésként próba verziókként érhető el szállításkor az eszközökön, a végleges licencet ezek esetében külön lehet megvásárolni.

A végleges licencek nem járnak le. Például, ha telepítettük egy forgalomirányítón az UC, a SEC vagy a Data licencet, a megfelelő funkciók akkor is elérhetők lesznek, ha új IOS verziót telepítünk.

**Licencek telepítése**

Előfeltételek:

* A megfelelő PAK beszerzése, ami egy levélben vagy elektronikusan beszerezhető, 11 karakteres azonosító.
* Rendelkezni kell érvényes Cisco felhasználói névvel és jelszóval.
* Le kell kérdezni a sorozatszámot és a PID-et a **show license udi** paranccsal, vagy meg kell nézni azt a forgalomirányító címketartó tálcáján.

A **show version** paranccsal ellenőrizhetjük - a forgalomirányító újraindítását követően -, hogy sikerült-e telepíteni a licencet.

A **show license** paranccsal további részleteket jeleníthetünk meg a Cisco IOS licencekről.

A **license accept end user agreement** globális paranccsal egy lépésben fogadhatjuk el az EULA feltételeit az összes Cisco IOS szoftvercsomagra és funkciókészletre nézve.

A cisco.com weboldalon megtalálhatók az IOS 15.0 további előnyei és részletes információi.