EIGRP

Enhanced Interior Gateway Routing Protocol (EIGRP) är ett tidigare Cisco-propertiärt routingprotokoll som främst används i miljöer som enbart består av Cisco-produkter. Då protokollet tidigare var Cisco propertiärt har andra leverantörer ej implementerat protokollet. Protokollet ersatte Interior Gateway Routing Protocol år 1993.

EIGRP

Grunder

EIGRP transporteras ej med hjälp av UDP och TCP utan är ett eget protokoll med protokollnummer 88.
Reliable Transport Protocol (RTP) används för att säkerställa att paket har kommit fram till grannar korrekt.
Administrative Distance (AD) är 90.
Protokollet använder sig av en metric som enligt standard är baserad på bandwidth och ackumulerad delay för att välja bästa väg till destination.
Hello timers används för hur ofta en EIGRP-router ska annonsera sin närvaro. Hold timers används för att berätta för en granne hur länge de ska vänta på Hello paket innan ett grannskap anses vara dött.
Multicast används för att skapa grannskap. Multicastadresserna är 224.0.0.10 och FF02::A.
Hela EIGRP databasen utbyts när ett grannskap etableras. Därefter utbyts enbart förändringar.
Protokollet stöder autentisering via MD5 och SHA.
EIGRP kan tagga rötter och filtrera baserat på distribute-lists och route-maps.
Protokollet kan annonsera rötter med annat next-hop än sig själv.
Det går att manuellt summera rötter var som helst inom routingdomänen.

Loopförhindrande

För att ej skapa loopar i ett EIGRP nät som har EIGRP något som kallas för Feasability Conidition. Om en rot har en viss Feasable Distance , ex. 2000, så måste en annan route nå en lägre Received Distance (advertised distance + interfacet egna distance) för att anse vara en bättre väg utan loopar. Är Received Distance lägre än nuvarande Successor-routen (den bästa routen) så kommer routen ej att installeras i routing-tabellen.

Även Split Horizon och Poisoned Reverse är aktiverat per default i EIGRP. Split Horizon innebär att en route annonseras ej via samma interface som det togs emot på. Split Horizon kan behöva stängas av på interface som ansluter mot flera routrar, ex. på hub:en i en hub-and-spoke topologi. Stängs av per interface med `no ip/ipv6 split horizon eigrp` i classic mode och `no split-horizon` i named mode.

Poisoned Reverse är ett "tillägg" till Split Horizon, de arbetar ihop. Poisoined Reverse annonserar tillbaka mottagna rötter på samma interface som det tas emot på men med en såpass hög metric att nätverket ej är nåbart.

Metrics

För att räkna ut den bästa routen så används metrics. Metrics kommer i två varianter, classic och wide. De olika värdena hänvisas till som K-värden.

Classic metrics

Metric	**Förklaring**
Bandwidth (K1)	Statiskt värde som konfigureras med *bandwidth* kommandot i interface-konfigurationsläge. Finns inget konfigurerat så autoskapas ett. Vid fysiska interface så används den sanna hastigheten enligt länken. EIGRP använder sig av den lägsta bandwidth:en enligt hela roten. Enbart ett värde används alltså längst hela vägen oavsett hur många routrar som finns.
Delay (K2)	Statiskt värde som konfigureras med *delay* kommandot i interface-konfigurationsläge. Värdet 123 blir 1230 mikrosekunder. EIGRP använder sig av den totala ackumulerade delay-värdet enligt hela routingvägen.
Reliability (K3)	Ett värde mellan 1 och 255. 255 innebär 100 procentlig reliability. 230 innebär 90 procentlig reliability. Det minsta värdet av reliability används längs routingvägen.
Load (K4)	Ett värde mellan 1 och 255. 1 är ett tomt interface och 255 är ett fullt. Den maximala Txload:en (ej Rxload) används för att räkna ut Load-metric.
MTU (K5)	Den minsta MTU:n enligt hela routingvägen.
Hop count (K6)	Värde mellan 1 och 255. Enligt EIGRPs defaultkonfiguration är dock det maximala hopp-räknaren 100.

Komposit-metricen av K-värdena räknas ut genom den här formeln:

image.png

MTU (K5) och Hop count (K6) är inte med i formeln.

Wide metrics

Classic metric i EIGRP får problem när länkhastigheter överskrider 1 Gbit/s. Classic metrics ser ej skillnad på 10 gbit/s interface och snabbare interface. Wide metrics adresserar de här problemen.

Metric	**Förklaring**
Throughput	Likartad till Bandwidth. Räknas ut enligt formeln 65536×10 7 /Interface Bandwidth och hastigheten skrivs ut i Kbit/s. Uträkningen visar hur långsammare interfacet är jämfört med en 655.36 Tbit/s länk.
Latency	Likartad till Delay. Räknas ut enligt 65536×Interface Delay/10 6 och visas i pikosekunder.
Reliability	Identisk till Reliability i classic metrics.
Load	Identisk till Load i classic metrics.
MTU	Identisk till MTU i classic metrics. Annonseras men används ej.
Hop count	Identisk till Hop count i classic metrics. Annonseras men används ej i path selection. Används till att undvika eventuella routingloopar.
Extended Metrics	Placeholder för framtida användning. Inkluderar Jitter, Energy, och Quiescent Energy.

Uträkningen sker enligt denna formel:

image.png

Wide metrics används när EIGRP är konfigurerat med named mode. Routrar som kör wide metrics är bakåtkompatibla med classic metrics men föredrar att köra wide metrics.

Influera EIGRPs vägval

För att manueltl influera EIGRPs vägval kan man konfigurera bandwidth eller delay på interfacen. Dock så avråds det från att ändra bandwidth-värdet av flera anledningar:

- QoS kan använda sig av bandwidth värdet

- EIGRP begränsar sin trafik till att använda max 50% av bandwidth värdet, vid belastning kan alltså EIGRP-paket kastas istället för att skickas och därmed orsaka problem med grannskapet
- Ändrande av delay orsakar ej problemen beskrivna ovan

Alltså är rekommendationen att man enbart ska använda delay för att påverka EIGRPs vägval.

Det går att ändra hur många procent av ett interface bandbredd som EIGRP får använda med kommandot `bandwidth-percent x` i af-interface läge.

EIGRP Paket

EIGRP paket bärs direkt över IP, ingen UDP/TCP inblandad, med protokollnummer 88. Den maximala storleken på EIGRP paket tas ifrån interfacets maximala MTU.

Varje EIGRP paket innehåller en 20 byte lång header följd av innehåll i form av TLV:er, Type Length Value triplets. TLV:erna innehåller information om EIGRP och TLV versioner, K-värden, Hold timers, kontrollinformation för att främja säker multicasting och route reachability informationen. Det finns ingen dedikerat RTP (Reliable Transport Protocol) header. Istället används Flags, Sequence Number och Acknowledgement för den mesta av RTP funktionaliteten i EIGRP. Viss funktionalitet är via TLV:er.

image.png

Fält	**Förklaring**
Version	Ett 4 bitar stort värde för EIGRPs protokollversion. EIGRP protokollet har inte ändrats sedan dess skapande och är alltid satt till 2.
Opcode	4 bitar stort värde som specifiserar EIGRP pakettyp. 1 = Update, 3 = Query, 4 = Reply, 5 = Hello/Ack, 10 = SIA Query, 11 = SIA Reply.
Checksum	24 bitar sort värde som används för att utföra en sanity check på EIGRP paketet. Fältet baseras på EIGRP paketet utan dess IP header.
Flags	32 bitar stort värde som indikerar specifika flaggor. 0x1 = Init, 0x2 = Conditional Receive, 0x4 = Restart, 0x8 = End-of-Table
Sequence	32 bitar stort värde som innehåller ett sekvensnummer som används av RTP. Används för att säkerställa säkert leverans av paket.

Acknowledgement	32 bitar stort värde som används för RTP. Det senaste sekvensnummret i sequence emottaget från granne placeras här.
Virtual router ID	16 bitar stort värde som identifierar den virtuella routern paketet är associerat med. 0x1 = Unicast Address Family, 0x2 = Multicast Address Family, 0x8000 = Unicast Service Address Family
Autonomous System Nunber	16 bitar stort värde som identifierar EIGRP domänen.
Type-Length-Value (TLV)	Används för att bära route-information och DUAL information. Följande TLV:er finns: 0x0001 EIGRP Parameters (General TLV Types) 0x0002 Authentication Type (General TLV Types) 0x0003 Sequence (General TLV Types) 0x0004 Software Version (General TLV Types) 0x0005 Next Multicast Sequence (General TLV Types) 0x0102 IPv4 Internal Routes (IP-Specific TLV Types) 0x0103 IPv4 External Routes (IP-Specific TLV Types) 0x0402 IPv6 Internal Routes (IP-Specific TLV Types) 0x0403 IPv6 External Routes (IP-Specific TLV Types) 0x0602 Multi Protocol Internal Routes (AFI-Specific TLV Types) 0x0603 Multi Protocol External Routes (AFI-Specific TLV Types)

TLV:er är ett specifikt sätt att lagra och skicka olika typer av information i ett enda datagram. En TLV innehåller:

- Type, en numerisk kod som indikerar vad för information som finns i Value-fältet
- Length, den total storleken av TLV fälten. Notera att vissa protokoll, ex. EIGRP, enbart sparar längden av Value-fältet i Length
- Value, bytes av olika storlek som innehåller den intressanta informationen

I EIGRP så innehåller varje Internal och External Route TLV ett enda route-värde. Update, Query, Reply, SIA-Query och SIA-Reply paketen innehåller minst en sådan TLV för att annonsera ett specifikt nätverk eller för att fråga om det.

EIGRP Pakettyper

Sju olika pakettyper finns i EIGRP. Dessa syns i Opcode värdet i EIGRP headern. Följande pakettyper finns:

1. Hello
2. Acknowledgement
3. Update
4. Query
5. Reply

6. SIA-Query
7. SIA-Reply

Update, Query, Reply, SIA-Query och SIA-Reply kallas för *reliable packets* för att EIGRP ser till att de levereras i rätt ordning med RTP.

Vad för typ av paket som har skickats kan kontrolleras med kommandot `show ip eigrp traffic`. Används EIGRP i en VRF så är kommandot `show ip eigrp vrf VRF-NAMN traffic`.

```
c6807#show ip eigrp vrf VRF-NAMN traffic
EIGRP-IPv4 VR(EIGRP-1) Address-Family Traffic Statistics for AS(1)
        VRF(VRF-NAMN)
  Hellos sent/received: 1474623/2949935
  Updates sent/received: 4878/5343
  Queries sent/received: 83/0
  Replies sent/received: 0/83
  Acks sent/received: 5051/4617
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
  Hello Process ID: 1258
  PDM Process ID: 1155
  Socket Queue: 0/10000/4/0 (current/max/highest/drops)
  Input Queue: 0/10000/4/0 (current/max/highest/drops)
```

Hello

EIGRP skickar Hello paket ut på interface som har lagts till i EIGRP processen och som inte är passiva. Hello meddelanden används för att identifiera grannar, verifiera att grannarna är kompatibla och som keepalive när ett grannskap är etablerat. Hello paket skickas via multicast till 224.0.0.10 eller FF02::A beroende på IP-version. Om statiska grannar är konfigurerade så skickas Hello paketen som unicast till den konfigurerade grannen. Intervallen som Hello-paketen skickas kan modifieras och är default 5 sekunder. På NBMA interface eller med bandbredd 1544 kbit/s eller mindre är default-intervallen 60 sekunder. Paketen har Opcode 5 och kommer inte att få en Acknowledgement tillbaka via RTP.

Acknowledgement

Acknowledgement (ACK) paket skickas för att bekräfta vissa EIGRP-paket för att säkerställa leverans. ACK Skicaks som svar till Update, Query, Reply, SIA-Query och SIA-Reply paket och skickas som unicast till sändaren. De använder samma Opcode som Hello, Opcode 5. Det ser i princip ut som ett Hello paket men innehåller inga TLV:er. Fältet Acknowledgement Number i EIGRP paket som skickas innehåller värdet från Sequence number i mottaget paket.

Update

EIGRP Update paket innehåller uppdateringar om routing information. De kan skickas både som multicast och unicast. På interface där det enbart finns en granne så skickas paketen som unicast. På interface där och när flera nya grannar märks samtidigt (ex. DMVPN) så används multicast. Efter full synkning av databasen så skickas nya uppdateringar som multicast. Routern förväntar sig ett Ack-svar på Update paket Om en granne ej skickar ett Ack-svar tillbaka så kommer routern att skicka om paketet via Unicast. På interface som är konfigurerade som P2P och mot statiskt konfigurerade grannar kommer unicast alltid att användas. Update paket använder sig av Opcode 1.

Query Paket

Query paket skickas när en granne letar efter den bästa routen till en destination. Ack-svar förväntas på Query paket. De kan skickas via multicast och unicast. Per default på multiaccess-interface (ex. Ethernet) så skickas paketen som Multicast. Kommer ingen Ack så kommer routern att skicka om paketet via unicast. På P2P interface och mot statisk konfigurerade grannar kommer paketen att gå som unicast. Query paket använder sig av Opcode 3.

Reply

Reply paket innehåller svaret på frågan i Query paket. Ack-svar förväntas. De skickas alltid som unicast till frågeställaren. Opcode 4 används.

SIA-Query och SIA-Reply

SIA står för *Stuck In Active*. Dessa opcodes används när en router ej har mottagit ett EIGRP Reply paket som svar på en EIGRP Query. SIA-Query frågar om en router fortfarande letar efter svaret på en Query. Om grannen fortfarande kör DUAL och letar efter ett bra svar svarar den direkt med en SIA-Reply. Då nollställs tiden som frågeställaren väntar på svar. Paketen skicas som unicast och Ack förväntas. SIA-Query använder Opcode 10 och SIA-Reply använder Opcode 11.

Reliable Transport Protocol

Hur RTP fungerar har beskrivits ovan, men bygger på att vissa Opcodes förväntar sig Ack-svar. I svarspaketet har siffran från Sequence Number kopierats till Acknowledgement Number.

RTP har en funktion som kallas Conditional Receive. Det låter EIGRP dela upp grannar på ett multiaccess-interface (ex. Ethernet med flera EIGRP grannar) i två grupper. I den ena gruppen finns grannar som har skickat Ack på samtliga multicast EIGRP paket och i den andra de som har misslyckats att svarar på minst 1. För att inte behöva vänta på samtliga grannar för att gå vidare med nästa paket så skickas paket med en flagga till den grupp som har Ack:at samtliga tidigare. Utan Conditional Receive hade EIGRP varit tvungen att vänta på samtliga grannar innan EIGRP kan skicka nästa paket. Detta åstakommes med specifika TLV:er, Sequence TLV och Next Multicast Sequence TLV. I Sequence TLVn finns en lista med IP-adresser. Finner en router sin egen IP adress i den listan kommer den att ignorera paketet. En router som ej finner sig själv i listan kommer placera sig själv i Conditional Receive mode (CR-mode).

Routergrannskap

Per default så hittar EIGRP grannar automagiskt. Det går även att manuellt konfigurerat grannar.

Dynamiska grannskap etablera genom att EIGRP är aktiverat på ett icke passivt interface vilket triggar att Hello-paket skickas till 224.0.0.10/FF02::A. När potentiella grannar upptäcker varandra så kontrolleras först kompabilitet. Följande värden måsta matcha:

1. Autentisering
2. Likadant konfigurerade K-värden
3. AS-nummer
4. Primära adresser för grannskap
5. Samma IP-nät på ett enda subnät

Hello-intervallen, alltså hur ofta ett Hello paket skickas, och Hold-intervallen, hur länge en router väntar på ett Hello-paket innan grannskapet anses dött, behöver inte matcha i EIGRP. Detta då Hold-intervallen som konfigureras på Router A indikirerar hur länge Router A:s grannar, ex. Router B, ska vänta på Hello paket från Router A.

```
! Konfiguration klassisk EIGRP

interface TenGigabitEthernet1/1/1
 ip hello-intervall eigrp 1 1
 ip hold-time eigrp 1 3

! Konfiguration named EIGRP

router eigrp EIGRP-NAMN
 !
 address-family ipv4 unicast autonomous-system 1
  !
  af-interface TenGigabitEthernet1/1/1
   hello-interval 1
   hold-time 3
```

image.png

Proceseen börjar när R2 mottager ett Hello-paket från R1. Vid mottagning av Hello-paket kommer R2 att skicka ett eget Hello-paket och placera grannen i läget Pending. Även R1 placerar R2 i Pending. Sedan skickas ett tomt Update paket från R2, det innehåller alltså inga TLV:er, med Init flaggan. Init flaggan indikerar att det är ett nytt grannskap och grannen ska skicka över hela EIGRP-databasen efter etablering. R1 skickar också ett tomt Update paket med Init-flaggen. Notera att Ack-fältet innehåller Seq-värdet som kom med R2s paket. R2 flyttar R1 från Pending till Up och

skickar ett Ack. R1 flyttar nu R2 från Pending till Up. Nu kommer hela EIGRP databasen att utbytas via Update paket. Därefter skickas enbart uppdateringar vid förändringar eller queries samt Hello-paket.

Grannskap kan kontrolleras via `show eigrp address-family ipvx detail`. Byt ut x till 4 eller 6, beorende på vilket routing-protokoll som används. Släng in `vrf VRF-NAMN` efter adressfamiljen om en VRF används. Detail är inte så mycket större än utan detail.

```
! Exempeloutput


catalyst#show eigrp address-family ipv4 vrf VRF-NAMN neighbors  detail
EIGRP (EIGRP) Address-Family Neighbors for AS(1)
          VRF(VRF-NAMN)
H   Address             Interface           Hold Uptime   SRTT   RTO  Q  Seq
                                          (sec)        (ms)       Cnt Num
1   1.2.3.4         Vl1337               12 6w1d       1   100  0  1641
   Static neighbor
   Version 23.0/2.0, Retrans: 0, Retries: 0, Prefixes: 479
   Topology-ids from peer - 0
0   5.6.7.8         Vl1337               12 6w2d       1   100  0  1748
   Static neighbor
   Version 23.0/2.0, Retrans: 1, Retries: 0, Prefixes: 12
   Topology-ids from peer - 0
Max Nbrs: 0, Current Nbrs: 0
```

H-värdet (Handle) är ett värde som EIGRP processen tilldelar grannar. Address är grannens adress. Interface är vilket interface grannskapet är etablerat över. Hold visar hur många sekunder det är kvar på grannskapet innan det tas ned, förutsatt att det inte kommer ett nytt Hello paket såklart. Uptime visar hur länge grannskapet har varit aktivt. SRTT (Smooth Round Trip Time) är en uppskattning för hur lång tid det tar att skicka ett paket till grannen och få tillbaka en Ack. RTO (Retransmit Time Out) visar hur länge routern väntar på en Ack för ett omskickat unicast paket när tidigare paket ej fick en Ack tillbaka. Q Cnt visar hur många paket som har förberets för att skicka och paket som har skickats men som ingen ACK har kommit tillbaka från grannen. Bör vara 0 i ett bra nätverk (vilket det även är i exemplet ovan). Sekvensnumret är sekvensnumret i det senast skickade EIGRP paketet.

Diffusing Update Algorithm (DUAL)

DUAL är en konvergensalgoritm som varje EIGRP router använder sig. av. Algoritmen räknar ut den bästa vägen till rotdestinationen och förhindrar loopar.

EIGRPs databas kallas för *topology table*. Varje post i databasen innehåller:

1. Nätverk (adress och nätmask)
2. Feasible Distance för prefix
3. Adress för EIGRP granne som annonserar nätverk samt utgående interface mot grannen
4. Metrics för nätverk annonserat av grannen (advertised distance) samt metrics för att nå nätverket genom lokala interfacet (received distance)
5. Status för nätverket
6. Extra information om nätverket (flaggor, nätverkstyp, origin m.m.)

Databasen populeras av lokalt injicerade nätverk, det vill säga direkt anslutna EIGRP nätverk och rötter som redistribueras lokalt, samt innehållet från samtliga EIGRP Update, Query, Reply, Sia-Query och SIA-Reply paket. För varje nätverk kommer EIGRP att köra DUAL-uträkning för att hitta den väg med minst kostnad och som är utan loopar till destinationen. Vinner nätverket i EIGRP sedan mot rötter genom andra källor så kommer den att installeras i routing-tabellen.

Varje nätverk har en status. De kan vara Passive (bra) eller Active (dåligt). Nätverk som är Passive är DUAL klara med och bästa vägen hittats. Nätverk som är Active letar EIGRP efter bästa vägen till genom Query-paket. När ett nätverk är Active får inte routern modifiera nuvarande post i routingtabellen. Status Active försvinner först när alla grannar har svarat på Query-paketet med en Reply.

```
! Det går att slänga på all-links på slutet av show-kommandot för att se rötter som inte har klarat Feasability Condition.

catalyst#show eigrp address-family ipv4 vrf VRF-NAMN topology
EIGRP-IPv4 VR(EIGRP-1) Topology Table for AS(1)/ID(5.5.5.5)
          Topology(base) TID(0) VRF(-NAMN)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 1.1.1.0/30, 1 successors, FD is 656670720, tag is 8928
        via 11.11.11.11 (656670720/656015360), Vlan1337

...

catalyst#show eigrp address-family ipv4 vrf -NAMN topology 1.1.1.0/30
EIGRP-IPv4 VR(EIGRP-1) Topology Entry for AS(1)/ID(5.5.5.5)
          Topology(base) TID(0) VRF(-NAMN)
EIGRP-IPv4(1): Topology base(0) entry for 1.1.1.0/30
  State is Passive, Query origin flag is 1, 1 Successor(s), FD is 656670720, RIB is 5130240
  Descriptor Blocks:
  193.44.81.54 (Vlan1337), from 11.11.11.11, Send flag is 0x0
```

```
    Composite metric is (656670720/656015360), route is External
    Vector metric:
      Minimum bandwidth is 1000 Kbit
      Total delay is 20000000 picoseconds
      Reliability is 255/255
      Load is 1/255
      Minimum MTU is 1500
      Hop count is 1
      Originating router is 172.31.255.75
    External data:
      AS number of route is 65534
      External protocol is BGP, external metric is 10
      Administrator tag is 8928 (0x000022E0)
```

I exemplet ovan så finns bara 1 successor till 1.1.1.0/30 och Feasible Distance är 656670720. Det som syns nedanför är till vänster Computed Distance och till höger Reported Distance/Advertised Distance, alltså uträknad metric inklusive vägen till grannen och den annonserade metricen från grannen. Finns flera successors, eller feasible successors, så syns de här.

Värt att veta om Feasible Distance är att det är en historisk siffra som ändras när en ny successor har en bättre computed distance. Skulle nuvarande successor ha en Computed Distance på 2000, Feasible Distance är 2000, men ex. delay ändras så att Computed Distance blir 1000 blir nu Feasible Distance 1000. Men om dess Computed Distance går upp till 2000 igen och ingen annan successor är bättre så kommer Feasible Distance att ligga kvar på 1000.

DUAL körs när en ny granne kommer upp på samtliga rötter som annonseras via Update/Query/Reply/Sia-Query/SIA-Reply. När en granne går ned så sätts alla rötter mottagna från den till metric infinity - alltså onårbara. Finns en Feasible Successor så kommer den direkt att befodras till en Successor. Det sker lokalt i routern, ingen granne är inblandad, och kallas därmed *local computation*.

Finns ingen feasible successor och bästa hittade väg i EIGRP topologin är via en granne som ej är en feasible successor kan den routen inte användas direkt då en routingloop kan skapas. Nuvarande route i RIB låses, FD sätts till nuvarande CD genom den oförändrade Successorn och nätverket markeas som Active. Query paket skickas till grannar för att hitta bästa vägen. Finns en Feasible Successor eller Successor som inte går via frågeställaren kommer grannarna att svara tillbaka med Reply till frågeställaren och en ny routinväg är etablerad. DUAL har inte körts på routrar som redan har en successor/FS. Finns däremot ingen FS/Successor som inte går via frågeställaren kommer nätverket att markeras som Active, DUAL-uträkning körs och Queries kommer att skickas till grannar. När alla routrar har svarat med Replies kan till slut nätverket markeras som Passive, antingen via originalposten som fanns i EIGRP topologin eller via en ny väg. Nätverket i RIB låses upp och ny route installeras.

Stuck-In-Active (SIA)

SIA är ett läge där en router har skickat Query men inte mottagit en Reply, nätverket är alltså markerat som Active i EIGRP topologin. Varje EIGRP router måste vänta på Replies från sina grannar innan de kan skicka tillbaka sitt egna Reply till den ursprunliga frågeställaren. Skulle en router längs vägen bete sig konstigt och inte svara så får frågeställaren inte något Reply alls. När en Query först skickas så börjar Active timer att ticka för nätverket routern frågar om. När tiden för timern är slut (default 3 minuter, kan konfigureras med `timers active-time x` i EIGRP-kontext) då ingen Reply har mottagits så anses nätverket vara SIA och grannskap mot grannar som ej svarar tas ned. Det kan alltså leda till totalt trafikavbrott.

För att undvika detta finns SIA-Query (opcode 10) och SIA-Reply (opcode 11). Efter halva Active timer-tiden så skickas en SIA-Query som frågar "Jobbar du fortfarande på min Query?". En korrekt fungerande router svarar med en SIA-Reply med innehållet "Ja, jag inväntar Replies" eller "Nej, uträkningen är färdig, här är min metric till destinationen". Active timer nollställs vid mottagande av SIA-Reply. Max tre stycken SIA-Queries kan skickas. Är uträkningen inte klar efter tre stycken SIA-Queries kommer grannskapet att tas ned. Skulle ingen SIA-Reply kommer på SIA-Query kommer grannskapet att tas ned när Active timer är slut.

EIGRP Named Mode

Named mode är hur EIGRP ska konfigureras idag. Äldre varianten kallas för classic mode. Det går att uppgradera en miljö i classic mode till named mode med kommandot `eigrp upgrade-cli`. Det går att köra EIGRP processer samtidigt i både named och classic mode.

I classic mode är konfigurationsplatserna spretiga. Viss konfig är under EIGRP processen och viss konfig i interfaceläge. I named mode är all konfig samlad på samma ställe.

Exempelkonfiguration EIGRP named mode:

```
router eigrp NAMED-MODE
 !
 address-family ipv4 unicast autonomous-system 1337
  !
  af-interface default
   passive-interface
  exit-af-interface
  !
  af-interface Tunnel1
   authentication mode md5
   authentication key-chain KEY-CHAIN
  hello-interval 1
   hold-time 3
```

```
 no passive-interface
exit-af-interface
!
topology base
 distribute-list prefix PREFIX-LIST out Tunnel1
 maximum-paths 6
 variance 4
 redistribute static
exit-af-topology
network 10.13.37.1 0.0.0.0
eigrp router-id 1.3.3.7
exit-address-family
!
address-family ipv6 unicast autonomous-system 1337
 !
 af-interface default
  shutdown
  passive-interface
 exit-af-interface
 !
 af-interface Loopback0
  no shutdown
 exit-af-interface
 !
 af-interface TenGigabitEthernet1/0/1
  hello-interval 1
  hold-time 3
  no passive-interface
 exit-af-interface
 !
 af-interface Vlan1337
  no shutdown
 exit-af-interface
 !
 topology base
  distribute-list prefix-list PREFIX_LIST out TenGigabitEthernet1/0/1
  maximum-paths 6
  variance 4
  redistribute static metric 1 1 1 1 1
 exit-af-topology
```

```
 eigrp router-id 1.3.3.7
exit-address-family
```

För IPv4 används fortfarande network kommandot för att lägga till interface i EIGRP processen. För IPv6 är det "no shutdown" som gäller för att lägga till interface i EIGRP processen. Per default är *alla* IPv6 interface med i EIGRP processen. En `shutdown` under `af-interface default` behövs för att ändra det.

Router ID

EIGRP använder sig av router ID för att identifiera en router. Värdet är 4 bitar stort och ser därmed ut som en IPv4 adress. Samtliga nätverk som annonseras i EIGRP har router ID med i paketet. En router som mottager ett nätverk med sitt eget router ID kommer att kasta det paketet. Det är en loop-förhindrande mekanism.

Ett router ID kan (och bör) konfigureras statiskt med `eigrp router-id x.x.x.x` kommandot i EIGRP processen. Konfigureras det ej manuellt tas automagiskt den högsta IPv4-adressen på ett Loopback interface som är uppe till router ID. Finns inget loopback interface tas den högsta IPv4-adressen från övriga IP adresser. Ett autokonfat router ID ändras ej om man ex. tar bort IPv4 adressen på interfacet. Det ändras när EIGRP processen startar om eller om ett manuellt router ID konfigureras. Nuvarande router ID kan hittas med `show eigrp protocols` och med `show ip protocols`.

Unequal-Cost Load Balancing (Variance)

En unik funktion som EIGRP tillför är unequal-cost load balancing, alltså möjligheten att lastbalansera trafik över länkar som inte är lika snabba. Det som möjliggör detta är Feasible Successors. Detta då trafik som går via feasible successors ska vara en loopfri väg. Det konfigureras med `variance x` i `topology base`. Om x = 1 utförs ingen unequal-cost load balancing. Trafikmängden på de olika interfacen är ej jämt fördelad, de snabbare interfacen får hantera mer trafik. Antalet vägar som trafik kan gå konfigureras med `maximum-paths x` kommandot. Det gäller även för trafikvägar med samma cost.

Add-Path

Vid användning av ex. DMVPN kan en hub känna till två lika bra vägar till en spoke. Dock kan hub:en per default inte annonsera dessa två vägar till andra spokes. De andra spokes:n har per default enbart en väg dit. För att tillåta en hub annonsera två vägar finns add-path. Det aktiveras på af-interface nivå (funktion finns enbart i named EIGRP) med kommandot `add-paths x` där x är en siffra mellan 1 och 4. maximum-paths måste vara satt till minst samma nivå och split-horizon måste vara avstängt på interface mot hubbarna. Add-Path går att kombinera med `Unequal-Cost Load Balancing`.

Stub

Stubar är ett sätt att snyggt öka skalbarheten och stabiliteten på ett nät. En stub router annonserar per default enbart sina egna nätverk och inte de den har lärt sig från grannar. Det går att annonsera andra nätverk genom att använda sig av en *leak-map*. Det går att definiera exakt vilka av sina egna nätverk en EIGRP stub router ska annonsera i `eigrp stub` kommandot med tilläggen *summary, connected, static, redistributed* och *received-only*. Att en router är stub annonseras via en specifik TLV i Hello-paketen vilket gör att en granne aldrig skickar en query till stub routrar.

Om en stub router skulle mottaga en query så kommer den ändå att processa den, det beror dock på vad för nätverk queryn frågar efter. Queries som routern själv har skickat kommer hanteras som vanligt. En query för lokala nätverk (summary/connected/static/redistributed) samt de som annonseras via en leak-map kommer stubroutern att hantera som vanligt. För ett nätverk som inte faller under nyss nämnda kategorier men som routern känner till kommer routern att svara med prefixet men med en oändlig distans, dvs ej nåbar. För nätverk som routern inte känner till kommer den att svara med oändlig distans. En stubrouter kan motta queries om en granne kör gammal kod som inte känner igen Stub TLVn eller om den sitter på ett delat segment med flera stub-routers där en stub router skickar Querien. För multiaccesslänkar där vissar är stub och andra inte kommer EIGRP att skicka queries via unicast till de som inte är stub eller så används Conditional Receive i RTP för att skicka multicast som enbart processas av icke stub routrar.

Stub är jättebra att använda mot siter som kanske enbart har en upplänk och inget bakom sig. Det ser till att suboptimal routing ej sker, stub routrar med sämre länkar kommer aldrig bli transit-routrar och antalet Query paket minskas i domänen vilket kan leda till att man undviker SIA-problem.

Per default så annonseras connected och summary när man konfigurerar `eigrp stub`. Verifiering kan ske med `show ip protocols` och grannar som är stubbar kan hittas med show `eigrp address-family ipvx neighbors detail`.

Stubnät	**Förklaring**
receive-only	Stubroutern annonserar inga prefix. Statisk routing eller NAT behövs.
leak-map x	Matchar en route-map (som matchar ACL eller prefix list) för att annonsera specifika nät som EIGRP känner till.
connected	Annonsera direktanslutna nät
static	Annonsera statiska rötter. Redistribueringskommando behövs också.
redistributed	Tillåt att rötter som redistribueras in i EIGRP annonseras.

Att en router är stub påverkar inte vad som annonseras uppströms till routern. Vill man ex. begränsa routingtabellen krävs route-filtrering och/eller route-summering.

Route Summarization

En stark fördel i EIGRP kontra ex. OSPF är att ruttsummering kan ske varsomhelst på interface-nivå. Korrekt implementerad ruttsummering, exempelvis att annonsera större block mot stub:ar, gör att antalet queries i ett nätverk minskar.

Konfigureringen sker i interface-läge (af-interface i named EIGRP) med kommandot `summary-address 10.0.0.0 255.0.0.0`. En *leak-map* går att slänga på på slutet för att skicka med vissa specifika rötter trots summeringen.

När en summering installeras så skapas också en discard route (nullroute) till Null0 för nätverket specifiserat i summeringen. Den discard routen får AD 5 per default. Det kan hända att den får en bättre AD än en manuellt konfigurerad summeringsrutt som då göms av denna. Antingen får man ändra AD:t på den manuella konfigureringsrutten eller så kan man ändra AD:t på EIGRPs summeringsrutt med kommandot `summary-metric 10.0.0.0 255.0.0.0 distance X` i `topology base`. Ändra ej AD:t till 255. IOS installerar då ej rutten och annonserar den ej.

EIGRP använder sig den lägsta metricen från de rutter som täcks av summeringen vid annonsering av summeringen. Varje gång den sämsta metricen ändras måste summeringsrutten annonseras på nytt med en ny metric. Det går att sätta en statisk metric med kommandot `summary-metric x` i `topology base` för att slippa det beteendet då i större miljöer innebär det många uträkningar och omskick.

Den äldre automatiska summeringen är avstängd per default. I äldre versioner (innan IOS 15.0(1)M) behöver den stängas av med `no auto-summary` i EIGRP konfläge.

Passive interface

Ett passivt interface är ett interface som finns med i EIGRP processen men som inte skickar Hello-paket. Per default skickar samtliga EIGRP interface hello paket. Passive interface ska användas på samtliga interface som inte är tänkt att gå mot en EIGRP granne.

Best practice är att i `af-interface default` konfigurera `passive-interface` och sedan gå in under de af-interface som går mot andra EIGRP routrar och konfigurera `no passive-interface`. Det gör att man inte råkar skicka Hellos ut på interface mot ex. klienter.

Graceful shutdown

Graceful shutdown är en funktion som tillåter EIGRP routrar att annonsera till sina grannar att EIGRP avaktiveras. Det kan vara per interface, adressfamilj eller hela processen. Paketet är ett Hello-paket med alla K-värden satta till 255. Funktionen är påslagen per default och går ej att ändra på.

Graceful restart

Stöd för NSR/GR finns. Aktiveras genom att slå `nsf` i adressfamiljen.

Verifiering går med show ip protocols | sec EIGRP NSF.

```
c9300#show ip protocols | sec EIGRP NSF
  EIGRP NSF enabled
    NSF signal timer is 20s
    NSF converge timer is 120s
```

Timers kan sättas enligt:

```
router eigrp EIGRP
 address-family ipv4 unicast autonomous-system 1337
  nsf
  timers nsf signal 25
  timers nsf converge 100
  timers graceful-restart purge-time 150
```

Autentisering

Klassisk EIGRP har stöd för MD5 autentiseringen. Named mode har stöd för MD5 och SHA-autentisering. Key chains används. Med SHA-autentisering kan man även konfigurera lösenord direkt på interfacen. Exempelkonfiguration finns längre upp för named mode. För classic konfigureras `ip authentication mode eigrp` och `ip authentication key-chain eigrp` på interfacen.

Default route

EIGRP har inte något kommando för att skapa en default route. Istället så kan man redistribuera in en default route eller använda sig av manuell summering. Då det är en redistribuerad route blir AD:t 170.

Det går även att annonsera ut en default-route via ett interface genom summering, ex. `summary-address 0.0.0.0 0.0.0.0`. Dock så kommer då en route för 0.0.0.0/0 installeras till Null0 med AD 5. Ens

redan installerade defaultroute kan alltså bli dold av denna.

EIGRP Over the Top

EIGRP OTP är ett sätt att köra EIGRP mellan routrar som ej är direktanslutna, ex. över ett L3 VPN nät via en SP. EIGRP OTP använder sig av LISP för dataforwarding, UDP enkapsulerad trafik. Det går även att köra route-reflectorer i EIGRP OTP, de fyller samma funktion som i BGP i en hub-and-spoke topologi. Grannar konfigureras statiskt med neighbor-kommandot.

```
! Exempelkonf


router eigrp EIGRP
 !
  address-family ipv4 unicast autonomous-system 1337
   !
   af-interface GigabitEthernet0/0
    no next-hop-self
    no split-horizon
   exit-af-interface
   !
   topology base
   exit-af-topology
   ! Nedan konfas på spokes
   neighbor 1.1.1.1 GigabitEthernet0/1 remote 100 lisp-encap
   ! Nedan kan konfas på route-reflektor för att dynamiskt finna grannar
   remote-neighbors source GigabitEthernet0/0 unicast-listen lisp-encap
```

En route till grannens IP-adress behövs inte. Då interfacet som används specifiseras i EIGRP konfigurationen så används det för att skickas iväg paketet med rätt IP header. Dock vid Ethernet interface kommer routern skicka en ARP fråga ut på det interfacet, så det är nog bäst att specifisera i routingtabellen. Om man inte vill leva med bös som Proxy ARP.

Det går även att lägga tillen ACL för remote neighbors på route reflektorn.

Logging

Loggingnivå konfigureras under routingprocessen. Det som kan konfigureras är eigrp event-log-size (maxstorlek på EIGRP loggen), event-logging (logga routing-events), log-neighbor-changes (logga när grannskap går upp/ned) och log-neighbor-warnings. Loggen kan kommas åt geom att slå `show eigrp address-family ipvx events`.

Route Filtering

Det går att manipulera vilka rötter som skickas ut eller installeras via vilket interface som helst. Det går även att konfigurera för en hel adressfamilj under *topology base*. `distribute-list` kommandot används. Det går att filtrera på ACLer, prefix-listor och route-maps.

Syntax:

- ACLs: distribute-list acl-number | acl-name { in | out } [interface]
- Prefix lists: distribute-list prefix prefix-list-name { in | out } [interface]
- Route maps: distribute-list route-map route-map-name { in | out } [interface]

Offset list

En offset list ändras för att modifiera metric på specifika nätverk inkomna via visst interface eller för hela processen istället för ex. att ändra hela interfacets delay. En ACL används för att identifiera trafiken. Syntax är ex. `offset-list 10 out 100 serial 0/0`. 100 läggs då till på rötter identifierade i standard ACL 10 för rötter som annonseras ut på interface serial0/0.

Rensa EIGRP databasen

`clear ip route *` tömmer routingtabellen men inte EIGRP topologin. För att tömma hela EIGRP topologin och ta ned samtliga grannskap kan man använda sig av `clear eigrp address-family ipvx neighbors`.

Revision #1
Created 26 April 2025 12:01:10 by Jehrlander
Updated 26 April 2025 12:01:14 by Jehrlander