

بخش پنجم

EIGRP

EIGRP یا Enhanced Interior Gateway Routing Protocol در اوایل دهه 90 توسط سیسکو ارائه شد. با اینکه EIGRP نسخه بهبود یافته IGRP است اما از نظر کارایی و عملکرد تفاوت‌هایی زیر بنایی با کلیه Distance Vector ها دارد و از این لحاظ بیشتر دارای شباهت و خصوصیات Link State گونه است، بطوریکه به آن Hybrid Routing Protocol گفته میشود و سیسکو آنرا Advance Distance Vector Protocol می نامد.

EIGRP برای حل مشکلات رشد شبکه های IGRP و کلا Distance Vector بوجود آمد و منجر به کاهش زمان Convergence در شبکه شد.

این پروتکل بر اساس DUAL یا Diffusing Update Algorithm کار میکند و برای ارتباط با همسایگان خود از Multicast استفاده میکند. (آدرس 224.0.0.10)

همسایه به محض دریافت این Packet به فرستنده بصورت Unicast, ACK (رسید) ارسال میکند.

برای جلوگیری از Loop در مسیر، روتر مسیر Backup (نام دیگر آن Feasible Successor) را نیز ذخیره میکند. همچنین EIGRP برای Summarization برخلاف OSPF نیازی به Area ندارد و هرجائی از شبکه این امکان وجود دارد.

EIGRP بعنوان یک Routing Protocol قابلیت Route پروتکل‌های IP, IPX و AppleTalk را داراست و برای هر یک، Routing Table مجزا میسازد.

از آنجا که قابلیت Route کردن پروتکل های مختلف را داراست، به ازای هر پروتکل سه جدول وضع میکند: Routing Table و Topology Table, Neighbor Table.

مشخصات کلی این پروتکل:

IP Protocol 88

Multicast Address: 224.0.0.10

Classless supported (VLSM)

Auto-Summary (IANA Classful) and Manual.

Administrative Distance = 90 (Internal), 5 (Summary), 170 (External)

Authentication supported. (Only MD5)

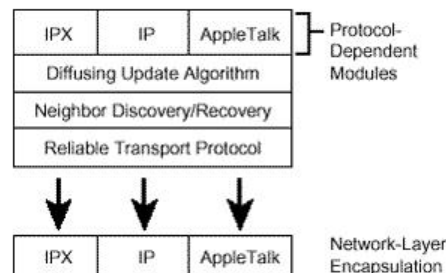
Partial updates supported.

اجزای EIGRP

چهار مزیت اصلی EIGRP عبارتند از:

• Protocol-Dependent Modules

پروتکل‌های IP، AppleTalk و Route را می‌کند و برای هر یک Routing Table مجزا می‌سازد. پروتکل Encapsulate کننده EIGRP برای هر یک از این پروتکل‌ها از جنس خودش است بطور مثال بسته‌های EIGRP برای Update های IPX داخل IPX حمل میشوند.



EIGRP بصورت خودکار IPX RIP، AppleTalk RTMP و IP IGRP را Redistribute میکند.

• RTP

Reliable Transport Protocol وظیفه انتقال پیام‌های EIGRP را بعهده دارد. انتقال پیامها بوسیله RTP همراه با گارانتی است. در واقع هر جا RTP همراه با ACK استفاده شود (مطمئن) است. این بسته‌ها از IP به شماره Type 88 و آدرس Multicast رزرو شده 224.0.0.10 استفاده می‌کنند. پیام‌های Hello نیازی به ACK نداشته و Unreliable رد و بدل میشوند. برای انتقال ACK ها از Unicast استفاده شده و انتقال update، query و reply نیاز به ارسال Reliable دارد.

اگر بسته‌ای به آدرس Multicast ارسال شود اما ACK از یکی از روترها دریافت نشود، بسته بصورت Unicast برای او ارسال می‌گردد و اگر تا 16 بار Retransmit شد و ACK دریافت نشد، آن همسایه بعنوان dead و غیرفعال شناخته میشود. فاصله بین ارسال این Unicast ها را RTO یا Retransmission Timeout می‌گوئیم. برای محاسبه این گونه زمانبندی‌ها در EIGRP، از فرمول SRTT یا Smooth Round Trip Time استفاده میشود.

SRTT میانگین زمان صرف شده از ارسال بسته تا دریافت ACK، بر حسب میلی ثانیه است

• Neighbor Discovery & Recovery

از آنجا که EIGRP از Update های نوبتی و دوره ای (Periodic Update) استفاده نمی‌کند از مکانیزم Hello ای بین همسایگان خود سود می‌برد که هر 5 ثانیه و به صورت Multicast انجام میشود. حال اگر ارتباط به شکل WAN و با پهنای باند کمتر از یک T1 (2 مگابیت) باشد بصورت Unicast و هر 60 ثانیه رخ میدهد. نباید فراموش کرد که در هر

حال Hello نیازی به ACK ندارد. Holddown در صورت عدم دریافت Hello تا سه برابر زمان Hello محاسبه شده و بعد از آن همسایه Dead شناخته میشود.

اطلاعات هر Neighbor یا همسایه داخل Neighbor Table قرار میگیرد.

در EIGRP تنها پیام های Hello بصورت Connection-less ارسال شده و بقیه پیام ها Connection-Oriented هستند.

• DUAL

Database شبکه توسط DUAL در EIGRP کشف و ایجاد میشود. فلسفه طراحی DUAL بر اساس Diffusing Computation است که اولین بار توسط Dijkstra و Scholten ارائه شد و الگوریتم DUAL توسط دکتر Dr. J. J. Garcia-Luna-Aceves پیشنهاد گردید.

برای فراگیری بهتر EIGRP بهتر است مفاهیم زیر را قبل از بررسی DUAL مرور کنیم:
Feasible Distance: یا FD، کمترین Metric تا مقصد است.

Advertised Distance: فاصله (Distance) گزارش شده از همسایه تا مقصد. (در واقع Metric همسایه تا مقصد)

Feasible Condition: یا FC حالتی است که Advertised Distance کمتر از FD برای همان مقصد باشد.

Feasible Successor: همسایه ایست که AD آن کوچکتر از FD است و نهایتاً FC رخ میدهد پس در Topological Database تمام روتر هایی که FS هستند ثبت شده و کوتاهترین مسیر به مقصد Successor میشود.
Successor: روتر بعدی به سمت مقصد یا همان Next-hop Router.

از آنجائیکه روتر براساس مقادیر FD و FS تصمیم گیری میکند، Loop Free بودن پروتکل تضمین شده و از آنجا که AD باید کوچکتر از FD باشد میتوان این استنباط را کرد که مسیر تبلیغ شده به مقصد خاص، قبلاً از خود روتر گرفته نشده است.

هدف DUAL ایجاد یک توپولوژی Loop Free در شبکه EIGRP است.

Route های Active

در EIGRP یک مسیر سالم، Passive نامیده میشود چرا که تلاشی برای پیدا کردن آن قبلاً صورت گرفته و اکنون در Routing Table قرار دارد اما اگر مسیر Active باشد به این معناست که روتر در حال جستجو برای پیدا کردن مسیری دیگر به آن شبکه down شده، میباشد. در واقع وقتی FS برای یک مقصد وجود نداشته باشد محاسبات Diffusing برای پیدا کردن مسیر به آن مقصد صورت گرفته و Route بحالت Active در میاید. در این حالت روتر بدلیل اینکه از Link های متصل به روتر های دیگر اطلاعات کافی ندارد (چون Link-State نیست) پس به سوال پرسیدن درباره شبکه خاص روی می آورد.

روتر برای مسیری که Active شده به تمام همسایگانش Query میفرستد و FD خود به مقصد را بینهایت یا Infinity اعلام میکند. اگر همسایگان - یک یا چند FS- برای مقصد مورد سوال داشته باشند به همراه Distance خود به درخواست جواب میدهند. در غیر اینصورت خود بحالت Active در آمده و به پرسش از همسایگان خود می پردازند . به ازای پاسخ های دریافت شده Diffusing Computation انجام میشود. از آنجاییکه جواب دهنده با Distance خود (که مسلماً بینهایت نیست) جواب Query را میدهد، پس وی خود به خود FS آن مسیر شده و با داشتن کوتاهترین Distance. عنوان Successor را میگیرد.

اگر هیچ جوابی از همسایگان در زمان مورد نظر دریافت نشود Route بحالت Stuck in Active یا به اختصار SIA درمیآید. همسایگانی که در زمان Active Timer (بمدت 3 دقیقه) جواب ندهند dead شناخته میشوند. تا زمانی که همه جوابها دریافت نشود Successor انتخاب نمیشود.

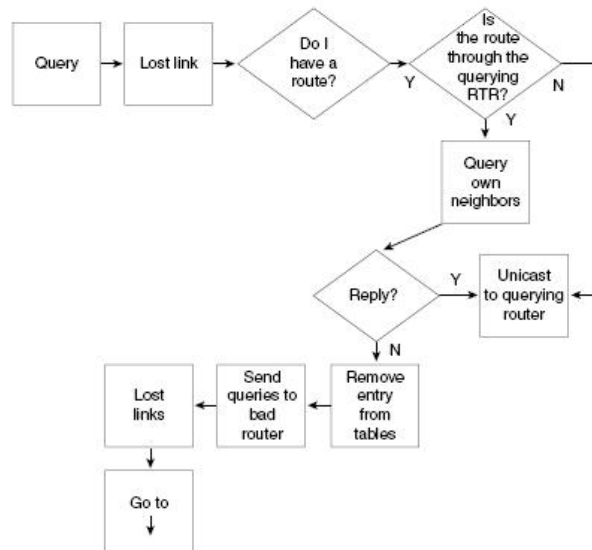
روتر تا زمانی که برای مسیری Backup Route یا Feasible Successor داشته باشد بحالت Active نمی رود.

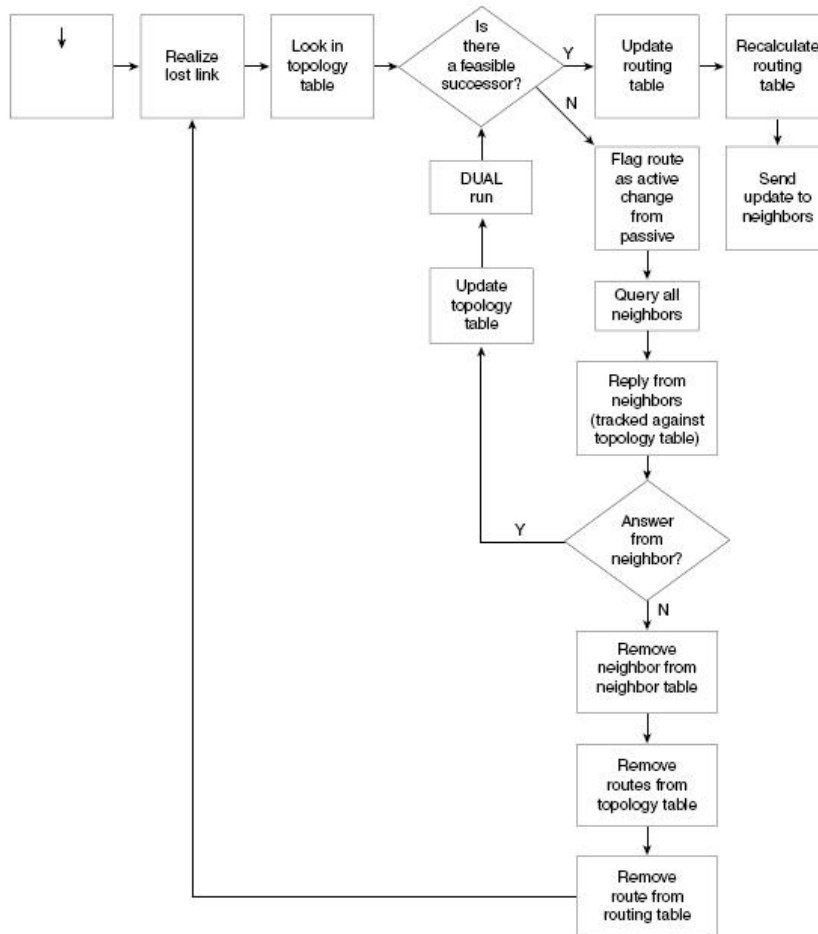
یک مسیر Infinite با Metric زیر مشخص میگردد:

Delay = 0xFFFFFFFF یا 4294967295

در Chart زیر نحوه نگهداری Topology Table رسم شده است:

EIGRP—Maintaining the Topology Table





انواع Packet در EIGRP

- **Hello**: برای پیدا کردن همسایه به آدرس 224.0.0.10 ارسال میشوند.
- **ACK**: یک بسته Hello بدون Data است که بصورت Unicast به فرستنده بعنوان رسید ارسال میشود.
- **Update**: حاوی اطلاعات برای همسایگان است که برای همسایه جدید بصورت Unicast اما در بقیه موارد بصورت Multicast برای همه ارسال میشود.
- **Query**: برای پیدا کردن Feasible Successor و به فرم Multicast ارسال میشود.
- **Reply**: در جواب یک Query و بصورت Unicast به فرستنده درخواست، ارسال شده و حاوی مشخصات Feasible Successor مسیر درخواستی است.

انواع Table در EIGRP

در EIGRP به ازای هر Protocol سه Table (جدول) نگهداری میشود:

• Neighbor Table

حاوی مشخصات همسایگان است مشخصاتی نظیر آدرس، Interface، Holdtime، Uptime، Sequence Number (بخاطر Connection-Oriented بودن)، SRTT، RTO و تعداد بسته در صف Queue.

• Topology Table

بعد از شناسایی Neighbor روتر قادر به ساخت Topology Table است و از طریق DUAL این جدول تشکیل میشود. از آنجائیکه Routing Table از این جدول بدست می آید، Topology Table باید حاوی اطلاعات لازم برای Routing Table باشد.

• Routing Table

بر اساس Routing Table - Topology Table شکل میگیرد و بهترین مسیر برای هر مقصد انتخاب میشود.

EIGRP در Metric

EIGRP نیز مثل پدرش IGRP، از Metric مرکب یا Composite بهره میبرد. تفاوت در این است که Metric در EIGRP ضربدر 256 شده تا از Granularity (درجه تفکیک بالاتری) برخوردار شود. این Metric براساس Bandwidth, Delay, Load, Reliability و MTU است. (Metric دخیل نیست).

در محاسبه بصورت Default و پیش فرض تنها از Bandwidth و Delay استفاده میشود. در صورتیکه بخواهیم از همه مولفه ها در محاسبه Metric استفاده شود از دستور زیر استفاده میکنیم:

```
Router(config-rtr) # metric weights 0 1 1 1 1 1
```

صفر اول نشاندهنده ToS صفر است و بترتیب مقادیر k_1, k_2, k_3, k_4, k_5 تنظیم میشوند. k_1 برای پهنای باند، k_2 برای Load، k_3 برای محاسبه Delay، k_4 برای Reliability و k_5 نشاندهنده MTU میباشد. که در حالت Default، $k_1=k_3=1$ و $k_2=k_4=k_5=0$ و محاسبه Metric به فرمول زیر است:

$$BW = 10^7 / \text{min-BW in Kbps}$$
$$\text{Metric} = 256 * (BW + \text{Delay})$$

در صورتیکه بخواهیم از همه مولفه های Metric استفاده کنیم از فرمول زیر Metric حاصل میشود:

$$\left[\left(K_1 \cdot \text{Bandwidth} + \frac{K_2 \cdot \text{Bandwidth}}{256 - \text{Load}} + K_3 \cdot \text{Delay} \right) \cdot \frac{K_5}{K_4 + \text{Reliability}} \right] \cdot 256$$

حداکثر تا شش مسیر هم عرض (با Metric مساوی برای یک مقصد) در Routing Table میتوان داشت که بصورت پیش فرض تا چهار مسیر نگهداری میشود.

مسیرهای داخلی یا Internal بعنوان مسیرهای درون Autonomous System شناسایی میشوند و دارای 90 Administrative Distance (بصورت Default) میباشند.

مسیرهای خارجی که بداخل EIGRP، Redistribute شده اند بنام External Route با Distance=170 و Summary شده با Distance=5 در Routing Table قرار میگیرند.

اصول طراحی EIGRP

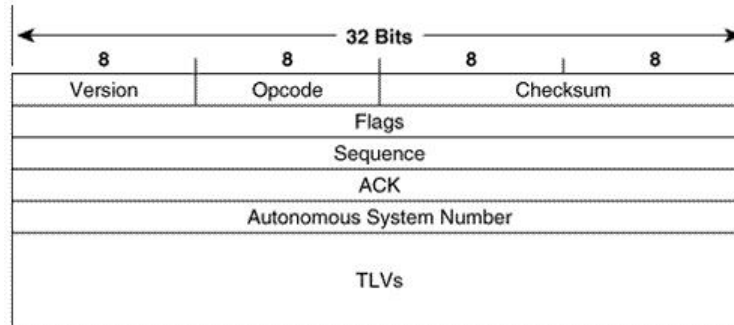
EIGRP برای کار در شبکه های بزرگ طراحی شده و هر شبکه بزرگی برای کارایی بهتر نیاز به استفاده بهینه از Bandwidth و صرفه جوئی در مصرف منابع (Resources) که منجر به عملکرد بهتر شبکه میشود دارد. معیار برای پیدا کردن یک شبکه ضعیف EIGRP، تعدد Route های SIA، وجود تاخیر در Convergence و استفاده زیاد از CPU و حافظه است.

در بسیاری از موارد در شبکه های بزرگ از چند EIGRP Autonomous System استفاده میشود تا Update های آنها از یکدیگر تفکیک گردد.

Summarization در EIGRP روی Interface تنظیم میشود.

شکل EIGRP Packet

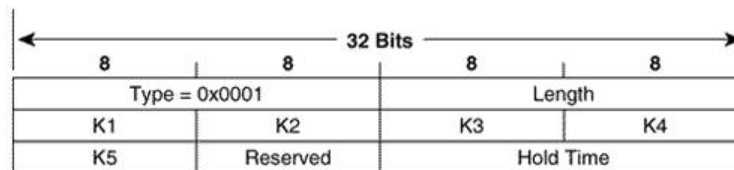
در زیر Header بسته های EIGRP ترسیم شده است:



معنای Opcode های مختلف بصورت جدول زیر است:

Opcode	Type
1	Update
3	Query
4	Reply
5	Hello
6	IPX SAP

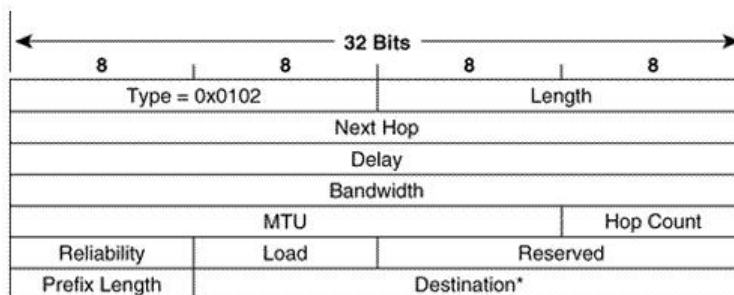
یک TLV در EIGRP بصورت زیر است:



انواع TLV در EIGRP در جدول زیر ذکر شده است:

Number	TLV Type	
0x0001	EIGRP Parameters	General TLV Types
0x0003	Sequence	
0x0004	Software Version	
0x0005	Next Multicast Sequence	
0x0102	IP Internal Routes	IP-Specific TLV Types
0x0103	IP External Routes	
0x0202	AppleTalk Internal Routes	AppleTalk-Specific TLV Types
0x0203	AppleTalk External Routes	
0x0204	AppleTalk Cable Configuration	
0x0302	IPX Internal Routes	IPX-Specific TLV Types
0x0303	IPX External Routes	

یک IP TLV در EIGRP Header به این صورت است:



*This field is variable. If it is less than or more than three octets, the TLV will be padded with zeros to the next four-octet boundary. For example, if the destination address is 10.1, the Destination field will be two octets and will be followed with a pad of 0x00. If the address is 192.168.16.64, the Destination field will be four octets and will be followed with a pad of 0x000000.

تنظیمات EIGRP

پروتکل های Routing بکمک دستور Network یک شبکه را درون پروتکل شروع به تبلیغ میکنند، با وارد کردن یک شبکه توسط دستور Network چهار اتفاق برای آن Interface که متعلق به شبکه مورد نظر است رخ میدهد:

- Update ها روی آن Interface گرفته میشوند.
- Update ها از Interface به بقیه ارسال میگردند.
- شبکه تنظیم شده توسط دستور Network، از همه Interface ها تبلیغ میشود.
- روی Interface Hello ارسال میگردد.

```
Router(config-router)# network network-number [wild-card-mask]
```

قبل از IOS 12.04(T)، Mask قابل تنظیم نبود و آدرس به فرم IANA Classful تبدیل میشد. در نتیجه اگر دو Interface با آدرس های مختلف ولی در یک Class قرار داشتند، هر دو آنها در EIGRP فعال میشدند. استفاده از Wildcard-mask برای تفکیک Interface در نظر گرفته شده است.

برای تنظیم Summarization بصورت Manual ابتدا باید Automatic Summarization را غیر فعال کنیم:

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

سپس در Interface میتوان دستور زیر را وارد کرد:

```
Router(config-if)# ip summary-address eigrp as-number address mask
```

برای حل مشکل SIA و یا ساده کردن update ها در یک محیط Hub and spoke میتوانیم از مزیت stub استفاده کنیم که عملکردی نظیر پروتکل ODR سیسکو دارد. برای تنظیم یک روتر stub از دستور زیر استفاده میکنیم تا خود را به همسایه ها Stub معرفی کنیم (روتر Stub بر عکس یک روتر Transit در بین مسیرها قرار نداشته و در لبه های شبکه قرار دارد):

```
Router(config-router)# eigrp stub
```

اگر بخواهیم هیچ Update ی برای همسایه مذکور ارسال نشود و تنها یک default route ارسال شود از پارامتر receive-only در مقابل eigrp stub استفاده میکنیم. حتی میتوان تنها route های static یا connected و یا summary خود را به همسایه اعلام کنیم که میتوان بدین گونه عمل کرد:

```
Router(config-router)# eigrp stub [receiveonly |connected| static|  
summary]
```

اگر چند مسیر برای یک مقصد وجود داشته باشد ، بهترین (از نظر Distance) انتخاب میشود و اگر بخواهیم چند مسیر غیر مساوی را بصورت مساوی در نظر بگیریم باید یک ضریب انتخاب کنیم که بر اساس کوتاهترین distance (ضرب در عدد مورد نظر) حد لازم برای مساوی بودن آنها محاسبه میگردد. تنها کفایت عدد ضریب یا variance به روتر اعلام شود.

```
Router(config-router) # variance number
```

مثلا اگر این عدد 2 باشد (Default = 1) حتی دو برابر کوتاهترین Metric نیز با آن برابر قلمداد شده و از این طریق Load Balancing بین مسیرهای صورت میگردد.

برای تغییر زمان Hello میتوان از دستور زیر استفاده کرد (برای لینکهای WAN با پهنای باند زیر 1.5 Mbps که بصورت Default، 60 ثانیه است دستور زیر میتواند بسیار مفید واقع شود).

```
Router(config-if) # ip hello-interval eigrp as-number seconds
```

باید توجه داشت که زمان Holdtime سه برابر Hello Interval است (در حالت Default) اما اگر زمان Hello را در تنظیمات تغییر دهیم در زمان Holdtime تغییری حاصل نمیشود مگر آنکه جداگانه تنظیم گردد:

```
Router(config-if) # ip hold-time eigrp as-number seconds
```

میزان Load روی یک Interface براساس ترافیک نسبت به Bandwidth لینک محاسبه میشود و لینکهای Serial بصورت Default با پهنای باند 1.5Mbps ست شده اند و در EIGRP آنها اعلام میکنند. در ضمن بصورت پیش فرض EIGRP برای ارسال Update ها و Message های خود حداکثر تا 50 درصد پهنای باند را میتواند اشغال کند.

ذکر دقیق Bandwidth به ازای Interface بسیار مفید است:

- برای محاسبه Bandwidth و Load در Metric
- برای اختصاص صحیح حداکثر 50 درصد پهنای باند مورد نیاز EIGRP

در ضمن به ازای Sub-Interface ها بهتر است Bandwidth داخل هر کدام بصورت تفکیک شده ذکر شود . (مثلا در Frame-Relay)

```
Router(config-if) # bandwidth speed  
Router(config-if) # ip bandwidth-percent eigrp as-number percent
```

برای چک کردن تنظیمات از دستور های زیر استفاده میکنیم:

```
Router# show ip eigrp neighbors
Router# show ip eigrp topology
Router# show ip eigrp topology all-links
Router# show ip eigrp traffic
```

برای پیدا کردن مشکل و **debugging** از دستور های زیر استفاده میکنیم:

```
Router# debug eigrp packet
Router# debug eigrp neighbors
Router# debug ip eigrp route
Router# debug ip eigrp summary
```

Command
accept-lifetime start-time{infinite end-time duration seconds}
auto-summary
bandwidth kilobits
debug eigrp packets
debug ip eigrp neighbor process-id address
delay tens-of-microseconds
ip authentication key-chain eigrp process-id key-chain
ip authentication mode eigrp process-id md5
ip bandwidth-percent eigrp process-id percent
ip hello-interval eigrp process-id seconds
ip hold-time eigrp process-id seconds
ip summary-address eigrp process-id address mask
key number
key chain name-of-chain
key-string text
metric weights tos k1 k2 k3 k4 k5
network network-number
passive-interface type number
router eigrp process-id
send-lifetime start-time {infinite end-time duration seconds}
show ip eigrp neighbors [type number]
show ip eigrp topology [process-id [[ip address]mask]]
timers active-time {minutes disabled}
traffic-share {balanced min}
variance multiplier