第10章　动态选路协议

10.1　引言

在前面各章中，我们讨论了静态选路。在配置接口时，以默认方式生成路由表项（对于直接连接的接口），并通过route命令增加表项（通常从系统自引导程序文件），或是通过ICMP重定向生成表项（通常是在默认方式出错的情况下）。

在网络很小，且与其他网络只有单个连接点且没有多余路由时（若主路由失败，可以使用备用路由），采用这种方法是可行的。如果上述三种情况不能全部满足，通常使用动态选路。

本章讨论动态选路协议，它用于路由器间的通信。我们主要讨论　RIP，即选路信息协议(Routing Infromation Protocol)，大多数TCP/IP实现都提供这个应用广泛的协议。然后讨论两种新的选路协议，　OSPF和BGP。本章的最后研究一种名叫无分类域间选路的新的选路技术，现在Internet上正在开始采用该协议以保持B类网络的数量。

10.2　动态选路

当相邻路由器之间进行通信，以告知对方每个路由器当前所连接的网络，这时就出现了动态选路。路由器之间必须采用选路协议进行通信，这样的选路协议有很多种。路由器上有一个进程称为路由守护程序（ routing daemon），它运行选路协议，并与其相邻的一些路由器进行通信。正如图9-1所示，路由守护程序根据它从相邻路由器接收到的信息，更新内核中的路由表。

动态选路并不改变我们在 9.2节中所描述的内核在IP层的选路方式。这种选路方式称为选路机制（routing mechanism）。内核搜索路由表，查找主机路由、网络路由以及默认路由的方式并没有改变。仅仅是放置到路由表中的信息改变了 —— 当路由随时间变化时，路由是由路由守护程序动态地增加或删除，而不是来自于自引导程序文件中的 route命令。

正如前面所描述的那样，路由守护程序将选路策略（ routing policy）加入到系统中，选择路由并加入到内核的路由表中。如果守护程序发现前往同一信宿存在多条路由，那么它（以某种方法）将选择最佳路由并加入内核路由表中。如果路由守护程序发现一条链路已经断开（可能是路由器崩溃或电话线路不好），它可以删除受影响的路由或增加另一条路由以绕过该问题。

在像Internet这样的系统中，目前采用了许多不同的选路协议。 Internet是以一组自治系统(AS，Autonomous System)的方式组织的，每个自治系统通常由单个实体管理。常常将一个公司或大学校园定义为一个自治系统。 NSFNET的Internet骨干网形成一个自治系统，这是因为骨干网中的所有路由器都在单个的管理控制之下。

每个自治系统可以选择该自治系统中各个路由器之间的选路协议。这种协议我们称之为内部网关协议IGP（Interior Gateway Protocol）或域内选路协议（intradomain routing protocol）。

最常用的IGP是选路信息协议 RIP。一种新的IGP是开放最短路径优先 OSPF（Open Shortest Path First）协议。它意在取代RIP。另一种1986年在原来NSFNET骨干网上使用的较早的IGP协议——HELLO，现在已经不用了。

新的RFC [Almquist 1993]规定，实现任何动态选路协议的路由器必须同时支持OSPF和RIP，还可以支持其他IGP协议。

外部网关协议EGP（Exterier Gateway Protocol）或域内选路协议的分隔选路协议用于不同自治系统之间的路由器。在历史上，（令人容易混淆）改进的EGP有着一个与它名称相同的协议：EGP。新EGP是当前在NSFNET骨干网和一些连接到骨干网的区域性网络上使用的是边界网关协议BGP（Border Gateway Protocol）。BGP意在取代EGP。

10.3 Unix选路守护程序

Unix系统上常常运行名为`routed`路由守护程序。几乎在所有的TCP/IP实现中都提供该程序。该程序只使用RIP进行通信，我们将在下一节中讨论该协议。这是一种用于小型到中型网络中的协议。

另一个程序是`gated`。IGP和EGP都支持它。[Fedor 1998]描述了早期开发的`gated`。图10-1对`routed`和两种不同版本的`gated`所支持的不同选路协议进行了比较。大多数运行路由守护程序的系统都可以运行`routed`，除非它们需要支持`gated`所支持的其他协议。

守护程序	内部网点协议			外部网点协议	
	HELLO	RIP	OSPF	EGP	BGP
routed		V1			
gated, 版本2	•	V1		•	V1
gated, 版本3	•	V1, V2	V2	•	V2, V3

图10-1 routed 和gated 所支持的选路协议

我们在下一节中描述 RIP 版本1，10.5节描述它与RIP版本2的不同点，10.6节描述OSPF，10.7节描述BGP。

10.4 RIP：选路信息协议

本节对RIP进行了描述，这是因为它是最广为使用（也是最受攻击）的选路协议。对于 RIP的正式描述文件是RFC 1058 [Hedrick 1988a]，但是该RFC是在该协议实现数年后才出现的。

10.4.1 报文格式

RIP报文包含中在UDP数据报中，如图10-2所示（在第11章中对UDP进行更为详细的描述）。

图10-3给出了使用IP地址时的RIP报文格式。

命令字段为1表示请求，2表示应答。还有两个舍弃不用的命令（3和4），两个非正式的命令：轮询（5）和轮询表项（6）。请求表示要求其他系统发送其全部或部分路由

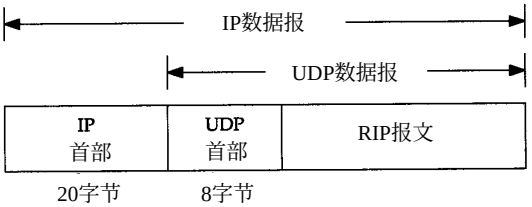

图10-2 封装在UDP数据报中的RIP报文

表。应答则包含发送者全部或部分路由表。

版本字段通常为1，而第2版RIP（10.5节）将此字段设置为2。

紧跟在后面的20字节指定地址系列（address family）（对于IP地址来说，其值是2）、IP地址以及相应的度量。在本节的后面可以看出， RIP的度量是以跳计数的。

采用这种20字节格式的RIP报文可以通告多达25条路由。上限25是用来保证RIP报文的总长度为20×25 + 4 = 504，小于512字节。由于每个报文最多携带 25个路由，因此为了发送整个路由表，经常需要多个报文。

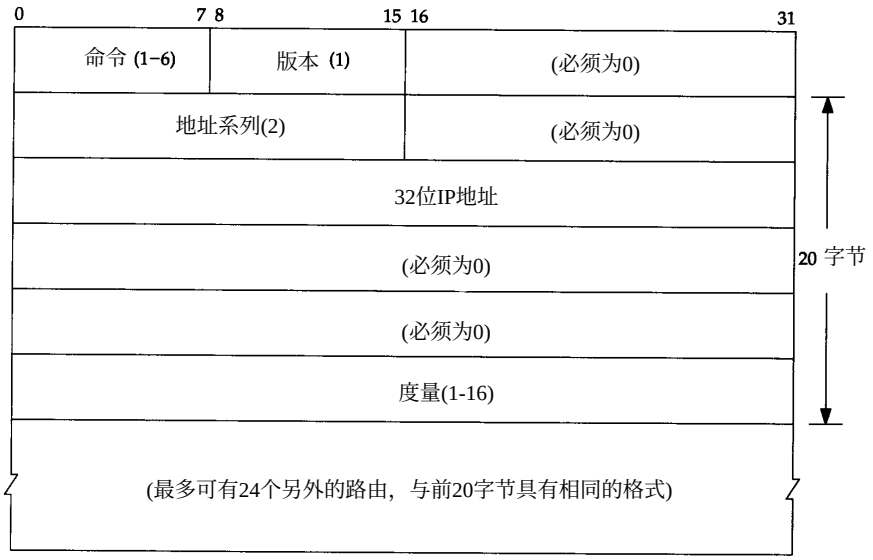

图 10-3

10.4.2 正常运行

让我们来看一下采用RIP协议的routed程序正常运行的结果。RIP常用的UDP端口号是520。

- 初始化：在启动一个路由守护程序时，它先判断启动了哪些接口，并在每个接口上发送一个请求报文，要求其他路由器发送完整路由表。在点对点链路中，该请求是发送给其他终点的。如果网络支持广播的话，这种请求是以广播形式发送的。目的 UDP端口号是520（这是其他路由器的路由守护程序端口号）。

  这种请求报文的命令字段为 1，但地址系列字段设置为 0，而度量字段设置为 16。这是一种要求另一端完整路由表的特殊请求报文。
- 接收到请求。如果这个请求是刚才提到的特殊请求，那么路由器就将完整的路由表发送给请求者。否则，就处理请求中的每一个表项：如果有连接到指明地址的路由，则将度量设置成我们的值，否则将度量置为 16（度量为16是一种称为“ 无穷大 ”的特殊值，它意味着没有到达目的的路由）。然后发回响应。
- 接收到响应。使响应生效，可能会更新路由表。可能会增加新表项，对已有的表项进行修改，或是将已有表项删除。
- 定期选路更新。每过30秒，所有或部分路由器会将其完整路由表发送给相邻路由器。发送路由表可以是广播形式的（如在以太网上），或是发送给点对点链路的其他终点的。

• 触发更新。每当一条路由的度量发生变化时，就对它进行更新。不需要发送完整路由表，而只需要发送那些发生变化的表项。

每条路由都有与之相关的定时器。如果运行 RIP的系统发现一条路由在 3分钟内未更新，就将该路由的度量设置成无穷大（ 16），并标注为删除。这意味着已经在 6个30秒更新时间里没收到通告该路由的路由器的更新了。再过 60秒，将从本地路由表中删除该路由，以保证该路由的失效已被传播开。

10.4.3 度量

RIP所使用的度量是以跳 (hop)计算的。所有直接连接接口的跳数为 1。考虑图10-4所示的路由器和网络。画出的 4条虚线是广播 RIP报文。

路由器R1通过发送广播到N1通告它与N2之间的跳数是 1（发送给N1的广播中通告它与N1之间的路由是无用的）。同时也通过发送广播给N2通告它与N1之间的跳数为1。同样， R2通告它与N2的度量为1，与N3的度量为1。

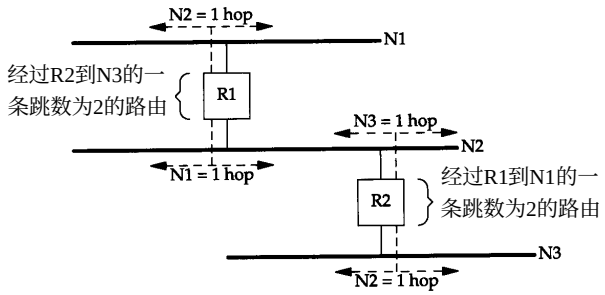

图10-4 路由器和网络示例

如果相邻路由器通告它与其他网络路由的跳数为1，那么我们与那个网络的度量就是 2，这是因为为了发送报文到该网络，我们必须经过那个路由器。在我们的例子中， R2到N1的度量是2，与R1到N3的度量一样。

由于每个路由器都发送其路由表给邻站，因此，可以判断在同一个自治系统 AS内到每个网络的路由。如果在该AS内从一个路由器到一个网络有多条路由，那么路由器将选择跳数最小的路由，而忽略其他路由。

跳数的最大值是15，这意味着RIP只能用在主机间最大跳数值为 15的AS内。度量为16表示到无路由到达该IP地址。

10.4.4 问题

这种方法看起来很简单，但它有一些缺陷。首先， RIP没有子网地址的概念。例如，如果标准的B类地址中16 bit的主机号不为0，那么RIP无法区分非零部分是一个子网号，或者是一个主机地址。有一些实现中通过接收到的RIP信息，来使用接口的网络掩码，而这有可能出错。

其次，在路由器或链路发生故障后，需要很长的一段时间才能稳定下来。这段时间通常需要几分钟。在这段建立时间里，可能会发生路由环路。在实现 RIP时，必须采用很多微妙的措施来防止路由环路的出现，并使其尽快建立。 RFC 1058 [Hedrick 1988a]中指出了很多实现RIP的细节。

采用跳数作为路由度量忽略了其他一些应该考虑的因素。同时，度量最大值为 15则限制了可以使用RIP的网络的大小。

10.4.5 举例

我们将使用`ripquery`程序来查询一些路由器中的路由表，该程序可以从 `gated`中得到。

ripquery程序通过发送一个非正式请求（图10-3中命令字段为5的“poll”）给路由器，要求得到其完整的路由表。如果在5秒内未收到响应，则发送标准的RIP请求（command字段为1）（前面提到过的，将地址系列字段置为0，度量字段置为16的请求，要求其他路由器发送其完整路由表）。

图10-5给出了将从sun主机上查询其路由表的两个路由器。如果在主机 sun上执行ripquery程序，以得到其下一站路由器netb的选路信息，那么可以得到下面的结果：

```
sun % ripquery -n netb
504 bytes from netb (140.252.1.183):第一份报文包含504字节
                                    这里删除了许多行
      140.252.1.0, metric 1         图10-5中上面的以太网
      140.252.13.0, metric 1        图10-5中下面的以太网
244 bytes from netb (140.252.1.183):第二份报文包含剩下的244字节下面删除了许多行
```

正如我们所猜想的那样，netb告诉我们子网的度量为1。另外，与netb相连的位于机端的以太网（140.252.1.0）的metric也是1（-n参数表示直接打印IP地址而不需要去查看其域名）。在本例中，将netb配置成认为所有位于140.252.13子网的主机都与其直接相连 —— 即，netb并不知道哪些主机真正与140.252.13子网相连。由于与140.252.13子网只有一个连接点，因此，通告每个主机的度量实际上没有太大意义。

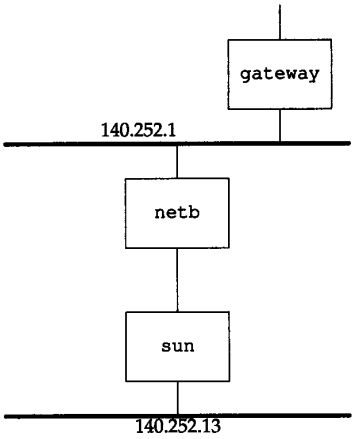

图10-5 查询其路由表内容的两个路由器netb 和gateway

图10-6给出了使用tcpdump交换的报文。采用 -i s10选项指定SLIP接口。

第1个请求发出一个RIP轮询命令（第1行）。这个请求在5秒后超时，发出一个常规的 RIP请求（第2行）。第1行和第2行最后的24表示请求报文的长度：4个字节的RIP首部（包括命令和版本），然后是单个20字节的地址和度量。

第3行是第一个应答报文。该行最后的25表示包含了25个地址和度量对，我们在前面已经计算过，其字节数为504。这是上面的ripquery程序所打印出来的结果。我们为tcpdump程序指定-s600选项，以让它从网络中读取600个字节。这样，它可以接收整个UDP数据报（而不是报文的前半部），然后打印出RIP响应的内容。该输出结果省略了。

```
   sun % tcpdump -s600 -i sl0
1  0.0                    sun.2879 > netb.route: rip-poll 24
2  5.014702 (5.0147)      sun.2879 > netb.route: rip-req 24
3  5.560427 (0.5457)      netb.route > sun.2879: rip-resp 25:
4  5.710251 (0.1498)      netb.route > sun.2879: rip-resp 12:
```

图10-6 运行ripquery 程序的tcpdump 输出结果

第4行是来自路由器的第二个响应报文，它包含后面的12个地址和度量对。可以计算出该报文的长度为12×20 + 4=244，这正是ripquery程序所打印出来的结果。

如果越过netb路由器，到gateway，那么可以预测到我们子网（140.252.13.0）的度量为2。可以运行下面的命令来进行验证：

```
sun %ripquery -n gateway
504 bytes from gateway (140.252.1.4):
                                                    这里删除了许多行
      140.252.1.0, metric 1          图10-5上面的以太网
      140.252.13.0, metric 2                    图10-5下面的以太网
```

这里，位于图10-5上面的以太网（140.252.1.0）的度量依然是1，这是因为该以太网直接与gateway和netb相连。而我们的子网140.252.13.0正如预想的一样，其度量为2。

10.4.6 另一个例子

现在察看以太网上所有非主动请求的RIP更新，以看一看RIP定期给其邻站发送的信息。图10-7是noao.edu网络的多种排列情况。为了简化，我们不用本文其他地方所采用的路由器表示方式，而以Rn来代表路由器，其中n是子网号。以虚线表示点对点链路，并给出了这些链路对端的IP地址。

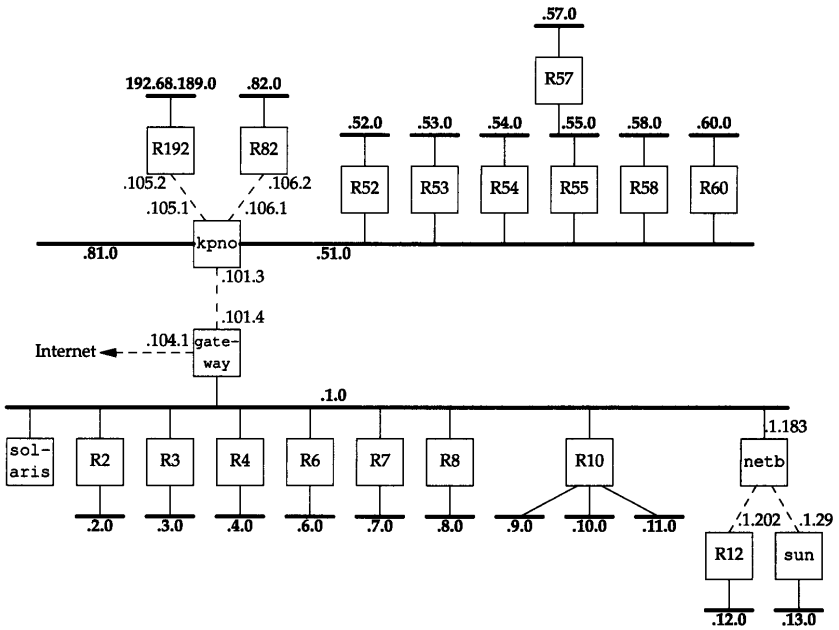

图10-7 noao.edu 140.252的多个网络

在主机solaris上运行Solaris 2.x的snoop程序，它与tcpdump相类似。我们可以在不需要超用户权限的条件下运行该程序，但它只捕获广播报文、多播报文以及发送给主机的报文。图10-8给出了在60秒内所捕获的报文。在这里，我们将大部分正式的主机名以Rn来表示。-P标志以非混杂模式捕获报文，-tr打印出相应的时戳，而udp port 520捕获信源或信宿端口号为520的UDP数据报。

来自R6、R4、R2、R7、R8和R3的前6个报文，每个报文只通告一个网络。查看这些报文，可以发现R2通告前往140.252.6.0的跳数为1的一条路由，R4通告前往140.252.4.0的跳数为1的一条路由，等等。

但是，gateway路由器却通告了15条路由。我们可以通过运行snoop程序时加上-v参数来查看RIP报文的全部内容（这个标志输出全部报文的全部内容：以太网首部、IP首部、UDP首部以及RIP报文。我们只保留了RIP信息而删除了其他信息）。图10-9给出了输出结果。

```
solaris % snoop -P -tr udp port 520
  0.00000 R6.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
  4.49708 R4.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
  6.30506 R2.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 11.68317 R7.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 16.19790 R8.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 16.87131 R3.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 17.02187 gateway.tuc.noao.edu -> 140.252.1.255 RIP R (15 destinations)
 20.68009 R10.tuc.noao.edu -> BROADCAST    RIP R (4 destinations)

 29.87848 R6.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 34.50209 R4.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 36.32385 R2.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 41.34565 R7.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 46.19257 R8.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 46.52199 R3.tuc.noao.edu -> 140.252.1.255 RIP R (1 destinations)
 47.01870 gateway.tuc.noao.edu -> 140.252.1.255 RIP R (15 destinations)
 50.66453 R10.tuc.noao.edu -> BROADCAST    RIP R (4 destinations)
```

图10-8 solaris 在60秒内所捕获到的RIP广播报文

把这些子网140.252.1上通告报文经过的路由与图10-7中的拓扑结构进行比较。

使人迷惑不解的一个问题是为什么图 10-8输出结果中， R10通告其有4个网络而在图10-7中显示的只有3个。如果查看带snoop的RIP报文，就会得到以下通告路由：

```
RIP: Address         Metric
RIP:  140.251.0.0     16 (not reachable)
RIP:  140.252.9.0     1
RIP:  140.252.10.0    1
RIP:  140.252.11.0    1
```

前往B类网络140.251的路由是假的，不应该通告它（它属于其他机构而不是noao.edu）。

```
solaris % snoop -P -v -tr udp port 520 host gateway
                                        删去许多行
RIP:  Opcode = 2 (route response)
RIP:  Version = 1

RIP:  Address         Metric

RIP:  140.252.101.0   1
RIP:  140.252.104.0   1

RIP:  140.252.51.0    2
RIP:  140.252.81.0    2
RIP:  140.252.105.0   2
RIP:  140.252.106.0   2

RIP:  140.252.52.0    3
RIP:  140.252.53.0    3
RIP:  140.252.54.0    3
RIP:  140.252.55.0    3
RIP:  140.252.58.0    3
RIP:  140.252.60.0    3
RIP:  140.252.82.0    3
RIP:  192.68.189.0    3

RIP:  140.252.57.0    4
```

图10-9 来自gateway 的RIP响应

图10-8中，对于 R10发送的RIP报文， snoop输出“ BROADCAST ”符号，它表示目的IP地址是有限的广播地址255.255.255.255（12.2节），而不是其他路由器用来指向子网的广播地

址（140.252.1.255）。

10.5 RIP版本2

RFC 1388 [Malkin 1993a]中对RIP定义进行了扩充，通常称其结果为 RIP-2。这些扩充并不改变协议本身，而是利用图10-3中的一些标注为“必须为0”的字段来传递一些额外的信息。如果RIP忽略这些必须为0的字段，那么，RIP和RIP-2可以互操作。

图10-10重新给出了由RIP-2定义的图。对于RIP-2来说，其版本字段为2。

图10-10 RIP-2报文格式

选路域(routing domain)是一个选路守护程序的标识符，它指出了这个数据报的所有者。在一个Unix实现中，它可以是选路守护程序的进程号。该域允许管理者在单个路由器上运行多个RIP实例，每个实例在一个选路域内运行。

选路标记(routing tag)是为了支持外部网关协议而存在的。它携带着一个 EGP和BGP的自治系统号。

每个表项的子网掩码应用于相应的 IP地址上。下一站IP地址指明发往目的IP地址的报文该发往哪里。该字段为0意味着发往目的地址的报文应该发给发送 RIP报文的系统。

RIP-2提供了一种简单的鉴别机制。可以指定 RIP报文的前20字节表项地址系列为 0xffff，路由标记为2。表项中的其余16字节包含一个明文口令。

最后，RIP-2除了广播（第12章）外，还支持多播。这可以减少不收听 RIP-2报文的主机的负载。

10.6 OSPF：开放最短路径优先

OSPF是除RIP外的另一个内部网关协议。它克服了 RIP的所有限制。 RFC 1247 [Moy 1991]中对第2版OSPF进行了描述。

与采用距离向量的 RIP协议不同的是， OSPF是一个链路状态协议。距离向量的意思是，RIP发送的报文包含一个距离向量（跳数）。每个路由器都根据它所接收到邻站的这些距离向

量来更新自己的路由表。

在一个链路状态协议中，路由器并不与其邻站交换距离信息。它采用的是每个路由器主动地测试与其邻站相连链路的状态，将这些信息发送给它的其他邻站，而邻站将这些信息在自治系统中传播出去。每个路由器接收这些链路状态信息，并建立起完整的路由表。

从实际角度来看，二者的不同点是链路状态协议总是比距离向量协议收敛更快。收敛的意思是在路由发生变化后，例如在路由器关闭或链路出故障后，可以稳定下来。 [Perlman 1992]的9.3节对这两种类型的选路协议的其他方面进行了比较。

OSPF与RIP（以及其他选路协议）的不同点在于， OSPF直接使用IP。也就是说，它并不使用UDP或TCP。对于IP首部的protocol字段，OSPF有其自己的值（图3-1）。

另外，作为一种链路状态协议而不是距离向量协议， OSPF还有着一些优于RIP的特点。

1) OSPF可以对每个IP服务类型（图3-2）计算各自的路由集。这意味着对于任何目的，可以有多个路由表表项，每个表项对应着一个IP服务类型。

2) 给每个接口指派一个无维数的费用。可以通过吞吐率、往返时间、可靠性或其他性能来进行指派。可以给每个IP服务类型指派一个单独的费用。

3) 当对同一个目的地址存在着多个相同费用的路由时，OSPF在这些路由上平均分配流量。我们称之为流量平衡。

4) OSPF支持子网：子网掩码与每个通告路由相连。这样就允许将一个任何类型的 IP地址分割成多个不同大小的子网（我们在 3.7节中给出了这样的一个例子，称之为变长度子网）。到一个主机的路由是通过全1子网掩码进行通告的。默认路由是以IP地址为0.0.0.0、网络掩码为全0进行通告的。

5) 路由器之间的点对点链路不需要每端都有一个 IP地址，我们称之为无编号网络。这样可以节省IP地址 —— 现在非常紧缺的一种资源。

6) 采用了一种简单鉴别机制。可以采用类似于 RIP-2机制（10.5节）的方法指定一个明文口令。

7) OSPF采用多播（第12章），而不是广播形式，以减少不参与OSPF的系统负载。

随着大部分厂商支持OSPF，在很多网络中OSPF将逐步取代RIP。

10.7 BGP：边界网关协议

BGP是一种不同自治系统的路由器之间进行通信的外部网关协议。 BGP是ARPANET所使用的老EGP的取代品。RFC1267 [Lougheed and Rekhter 1991] 对第3版的BGP进行了描述。

RFC 1268 [Rekhter and Gross 1991] 描述了如何在Internet中使用BGP。下面对于BGP的大部分描述都来自于这两个RFC文档。同时， 1993年开发第4版的BGP（见RFC 1467 [Topolcic 1993]），以支持我们将在10.8节描述的CIDR。

BGP系统与其他BGP系统之间交换网络可到达信息。这些信息包括数据到达这些网络所必须经过的自治系统AS中的所有路径。这些信息足以构造一幅自治系统连接图。然后，可以根据连接图删除选路环，制订选路策略。

首先，我们将一个自治系统中的 IP数据报分成本地流量和通过流量。在自治系统中，本地流量是起始或终止于该自治系统的流量。也就是说，其信源 IP地址或信宿IP地址所指定的主机位于该自治系统中。其他的流量则称为通过流量。在 Internet中使用BGP的一个目的就是

减少通过流量。

可以将自治系统分为以下几种类型：

1) 残桩自治系统(stub AS)，它与其他自治系统只有单个连接。stub AS只有本地流量。

2) 多接口自治系统(multihomed AS)，它与其他自治系统有多个连接，但拒绝传送通过流量。

3) 转送自治系统(transit AS)，它与其他自治系统有多个连接，在一些策略准则之下，它可以传送本地流量和通过流量。

这样，可以将Internet的总拓扑结构看成是由一些残桩自治系统、多接口自治系统以及转送自治系统的任意互连。残桩自治系统和多接口自治系统不需要使用 BGP——它们通过运行EGP在自治系统之间交换可到达信息。

BGP允许使用基于策略的选路。由自治系统管理员制订策略，并通过配置文件将策略指定给BGP。制订策略并不是协议的一部分，但指定策略允许 BGP实现在存在多个可选路径时选择路径，并控制信息的重发送。选路策略与政治、安全或经济因素有关。

BGP与RIP和OSPF的不同之处在于BGP使用TCP作为其传输层协议。两个运行BGP的系统之间建立一条TCP连接，然后交换整个BGP路由表。从这个时候开始，在路由表发生变化时，再发送更新信号。

BGP是一个距离向量协议，但是与（通告到目的地址跳数的） RIP不同的是，BGP列举了到每个目的地址的路由（自治系统到达目的地址的序列号）。这样就排除了一些距离向量协议的问题。采用16 bit数字表示自治系统标识。

BGP通过定期发送keepalive报文给其邻站来检测TCP连接对端的链路或主机失败。两个报文之间的时间间隔建议值为 30秒。应用层的keepalive报文与TCP的keepalive选项（第23章）是独立的。

10.8 CIDR：无类型域间选路

在第3章中，我们指出了B类地址的缺乏，因此现在的多个网络站点只能采用多个 C类网络号，而不采用单个B类网络号。尽管分配这些C类地址解决了一个问题（B类地址的缺乏），但它却带来了另一个问题：每个 C类网络都需要一个路由表表项。无类型域间选路（ CIDR）是一个防止Internet路由表膨胀的方法，它也称为超网（ supernetting）。在RFC 1518 [Rekhter and Li 1993] 和RFC 1519 [Fuller et al. 1993]中对它进行了描述，而[Ford, Rekhter, and Braun 1993]是它的综述。CIDR有一个Internet Architecture Board ’s blessing [Huitema 1993]。RFC 1467 [Topolcic 1993] 对Internet中CIDR的开发状况进行了小结。

CIDR的基本观点是采用一种分配多个 IP地址的方式，使其能够将路由表中的许多表项总和(summarization)成更少的数目。例如，如果给单个站点分配 16个C类地址，以一种可以用总和的方式来分配这 16个地址，这样，所有这 16个地址可以参照Internet上的单个路由表表项。同时，如果有8个不同的站点是通过同一个 Internet服务提供商的同一个连接点接入 Internet的，且这8个站点分配的8个不同IP地址可以进行总和，那么，对于这 8个站点，在Internet上，只需要单个路由表表项。

要使用这种总和，必须满足以下三种特性：

1) 为进行选路要对多个IP地址进行总和时，这些IP地址必须具有相同的高位地址比特。

2) 路由表和选路算法必须扩展成根据32 bit IP地址和32 bit掩码做出选路决策。

3) 必须扩展选路协议使其除了32 bit地址外，还要有32 bit掩码。OSPF（10.6节）和RIP-2（10.5节）都能够携带第4版BGP所提出的32 bit掩码。

例如，RFC 1466 [Gerich 1993] 建议欧洲新的C类地址的范围是194.0.0.0～195.255.255.255。以16进制表示，这些地址的范围是0xc2000000～0xc3ffffff。它代表了65536个不同的C类网络号，但它们地址的高7 bit是相同的。在欧洲以外的国家里，可以采用IP地址为0xc2000000和32 bit 0xfe000000 (254.0.0.0) 为掩码的单个路由表表项来对所有这些65536个C类网络号选路到单个点上。C类地址的后面各比特位（即在194或195后面各比特）也可以进行层次分配，例如以国家或服务提供商分配，以允许对在欧洲路由器之间使用除了这32 bit掩码的高7 bit外的其他比特进行概括。

CIDR同时还使用一种技术，使最佳匹配总是最长的匹配：即在32 bit掩码中，它具有最大值。我们继续采用上一段中所用的例子，欧洲的一个服务提供商可能会采用一个与其他欧洲服务提供商不同的接入点。如果给该提供商分配的地址组是从194.0.16.0到194.0.31.255 (16个C类网络号)，那么可能只有这些网络的路由表项的IP地址是194.0.16.0，掩码为255.255.240.0 (0xfffff000)。发往194.0.22.1地址的数据报将同时与这个路由表表项和其他欧洲C类地址的表项进行匹配。但是由于掩码255.255.240比254.0.0.0更“长”，因此将采用具有更长掩码的路由表表项。

“无类型”的意思是现在的选路决策是基于整个32 bit IP地址的掩码操作，而不管其IP地址是A类、B类或是C类，都没有什么区别。

CIDR最初是针对新的C类地址提出的。这种变化将使Internet路由表增长的速度缓慢下来，但对于现存的选路则没有任何帮助。这是一个短期解决方案。作为一个长期解决方案，如果将CIDR应用于所有IP地址，并根据各洲边界和服务提供商对已经存在的IP地址进行重新分配（且所有现有主机重新进行编址！），那么[Ford, Rekhter, and Braun 1993] 宣称，目前包含10 000网络表项的路由表将会减少成只有200个表项。

10.9 小结

有两种基本的选路协议，即用于同一自治系统各路由器之间的内部网关协议（IGP）和用于不同自治系统内路由器通信的外部网关协议（EGP）。

最常用的IGP是路由信息协议（RIP），而OSPF是一个正在得到广泛使用的新IGP。一种新近流行的EGP是边界网关协议（BGP）。在本章中，我们讨论了RIP及其交换的报文类型。第2版RIP是其最近的一个改进版，它支持子网，还有一些其他改进技术。同时也对OSPF、BGP和无类型域间选路（CIDR）进行了描述。CIDR是一种新技术，可以减小Internet路由表的大小。

你可能还会遇到一些其他的OSI选路协议。域间选路协议（IDRP）最开始时，是一个为了使用OSI地址而不是IP地址，而进行修改的BGP版本。Intermediate System to Intermediate System 协议（IS-IS）是OSI的标准IGP。可以用它来选路CLNP（无连接网络协议），这是一种与IP类似的OSI协议。IS-IS和OSPF相似。

动态选路仍然是一个网间互连的研究热点。对使用的选路协议和运行的路由守护程序进行选择，是一项复杂的工作。[Perlman 1992]提供了许多细节。

习题

10.1 在图10-9中哪些路由是从路由器kpno进入gateway的？

10.2 假设一个路由器要使用RIP通告30个路由，这需要一个包含25条路由和另一个包含5条路由的数据报。如果每过一个小时，第一个包含25条路由的数据报丢失一次，那么其结果如何？

10.3 OSPF报文格式中有一个检验和字段，而RIP报文则没有此项，这是为什么？

10.4 像OSPF这样的负载平衡，对于传输层的影响是什么？

10.5 查阅RFC1058 关于实现RIP的其他资料。在图10-8中，140.252.1网络的每个路由器只通告它所提供的路由，而它并不能通过其他路由器的广播中知道任何其他路由。这种技术的名称是什么？

10.6 在3.4节中，我们说过除了图10-7中所示的8个路由器外，140.252.1子网上还有超过100个主机。那么这100个主机是如何处理每30秒到达它们的8个广播信息呢（图10-8）？