

RIP协议

2019年5月4日 16:00

距离向量算法

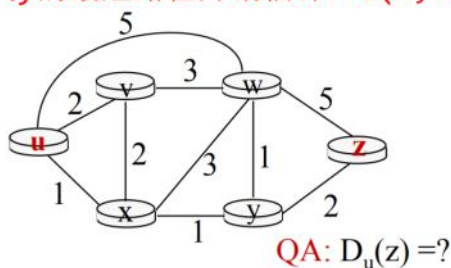
距离从向量算法(Distance vector algorithm)是一种最短路径算法,也称为 Bellman Bellman-Ford 算法或Ford-Fulkerson算法 (Bellman, 1957; Ford-Fulkerson, 1962)。其基本思想如下:

- 每个结点周期性地发送它自己的距离向量估计(distance vector estimate)给邻居。
- 当一个节点x收到来自邻居的新的距离向量估计 $D_v(y)$ ($y \in N$)时, 它将根据下面的B-F等式更新它自己的距离向量:

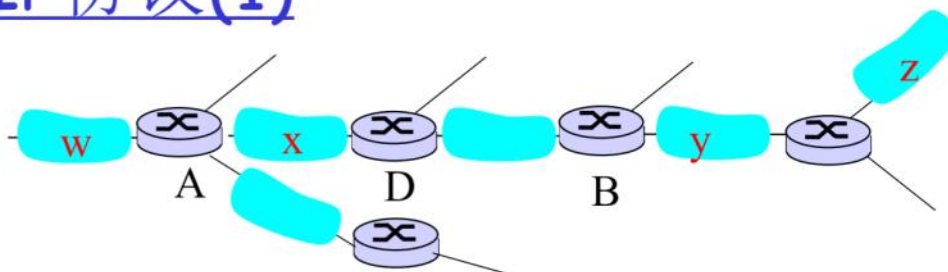
$$D_x(y) = \min_v \{c(x,v) + D_v(y)\} \quad \text{for each node } y \in N$$

其中, v 为 x 的邻居, $D_x(y)$ 为 x 到 y 的最短路径开销估计, $c(x,v)$ 为边 $\langle x,v \rangle$ 的开销

- 在自然条件下, $D_x(y)$ 会收敛到实际的最短路径开销。



RIP协议(1)



- 路由信息协议(Routing Information Protocol, RIP)是一种采用距离向量算法的路由协议。
- 到目的网络的距离以跳为单位。最大距离为15。距离16表示无穷大, 即目的网络不可达。
协议可用的网络直径最大为16
- 初始时每个RIP路由器只有到直连网的路由, 它们的距离为1。
- 每30秒RIP路由器把它的整个路由表发送给邻居。具体实现时每个邻居会错开发送, 30秒的时间也会随机变化一点。

最开始的时候每个路由器的路由表是空的, 先会将直连的路由器ip加入, 距离一般就是1

RIP协议(2)

- 当收到邻居发来的路由表(update packet), 路由器将更新它的路由表 <目的网络, 开销, 下一跳>:

(1) 收到路由的距离全部加1 (即一跳的距离)。

(2) 利用上述路由修改路由表: 如果下一跳就是送路由表的路由器, 则不管怎么样都要更新路由表

- 把路由表中不存在的路由加入路由表。
- 如果比路由表中的路由的距离更小, 则更新该路由的距离为新距离, 把下一跳改为邻居。
- 如果路由存在, 就要重置失效定时器。

- RIP路由表的每一项都有TTL(Time-To-Live), 用失效定时器(invalid timer)计时, 超时则让该路由失效。

- 一个路由器不需要知道它下一跳之后的路应该怎么走, 他只需要选取目的网络和下一跳相同的情况下, 开销最小的那一个来更新自己当前的路由表。
- 邻居的路由到他这里要加一是要算上他和邻居的距离, 然后再和自己的路由合并。
- 路由的下一条本来就是刚刚发路由表的邻居, 则它不管开销大小一定会被修改, 因为有可能有的网络发生了位置变化或者断开, 所以这样一定要相信邻居发来的路由表的最新信息。如果这个路由器有一跳
- 如果下一跳不是发路由表的邻居, 就比对相同目的网络的项, 通过开销大小更新路由, 如果要选择邻居的路由, 记得把下一跳改成邻居, 如果目的网络、下一跳都相同, 则更新TTL
- 到了TTL则就会把开销改成无穷大

RIP协议的问题

- 慢收敛问题(Slow Convergence)



- 计数到无穷问题(Count to Infinity)



因为路由器之间每隔 30 秒向邻居传送一次路由表, 因为路由表 30 秒发送是错开的, 所以会出现一些最快最慢的问题。当

N1接入网络后R1立刻会加入<N1 1->这一个路由，因为它到N1是一个直连网，这也是为什么是 $30 * (m-1)$ 的原因

最快情况：如果R1发送完后，R2刚好到发送时间，R3收到后也刚好发送，依次类推

最差情况：R1发送完，R2也刚好发送完，则要再等30秒才能继续下一次，以此类推

计数到无穷问题：

如果R1把N1路由修改成16后还没向R2发送消息，R2就发送了消息过来，那就会更新R1到N1的路由，且距离更新为R2到N1原来距离+1的距离且下一跳变成R2，然后R1又向R2发这条消息，R2因为下一条是R1则必须更改，同理R1的下一跳也是R2则也必须更改，然后距离就一直加——直到无穷了

有以下的方法解决计数到无穷的问题：

RIP协议的技术

R2给R1先发送，然后R3给R1再发送，最后R1又给R2发送

如果有回路还是有问题

❑ 水平分割技术(split horizon)

从一个接口学来的路由不会从该接口发回去。

❑ 毒性反转技术(poison reverse)

当一条路由变为无效之后，路由器并不立即将它从路由表中删除，而是将其距离改为用16后广播给邻居，使邻居所拥有的该路由立即失效，而不是等待TTL到期后删除，以迅速消除路由环路，这种方法称为毒性反转，距离为16的路由称为毒化路由(poisoned route)。

❑ 抑制技术(hold down)

距离被改为无穷大的路由在一段短时间(180秒)内其距离不允许被修改。

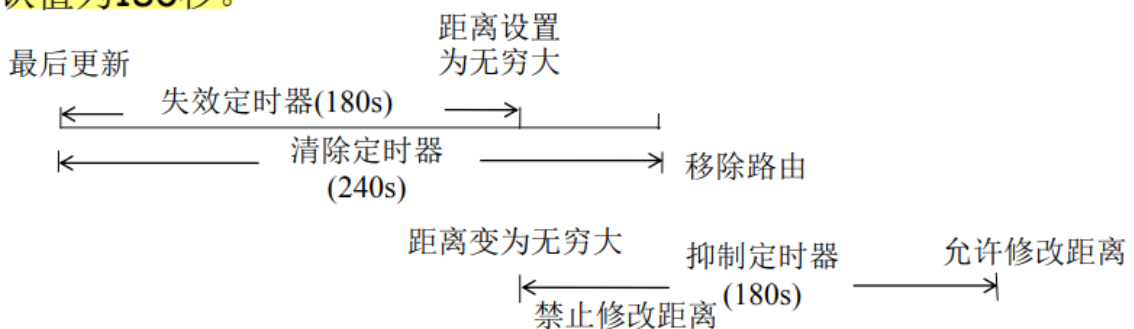
❑ 触发更新(triggered update)

一旦出现路由变化将立即把变化的路由发送给邻居。原有的30秒发一次完整的路由表依然不变。但还是存在一定概率出错



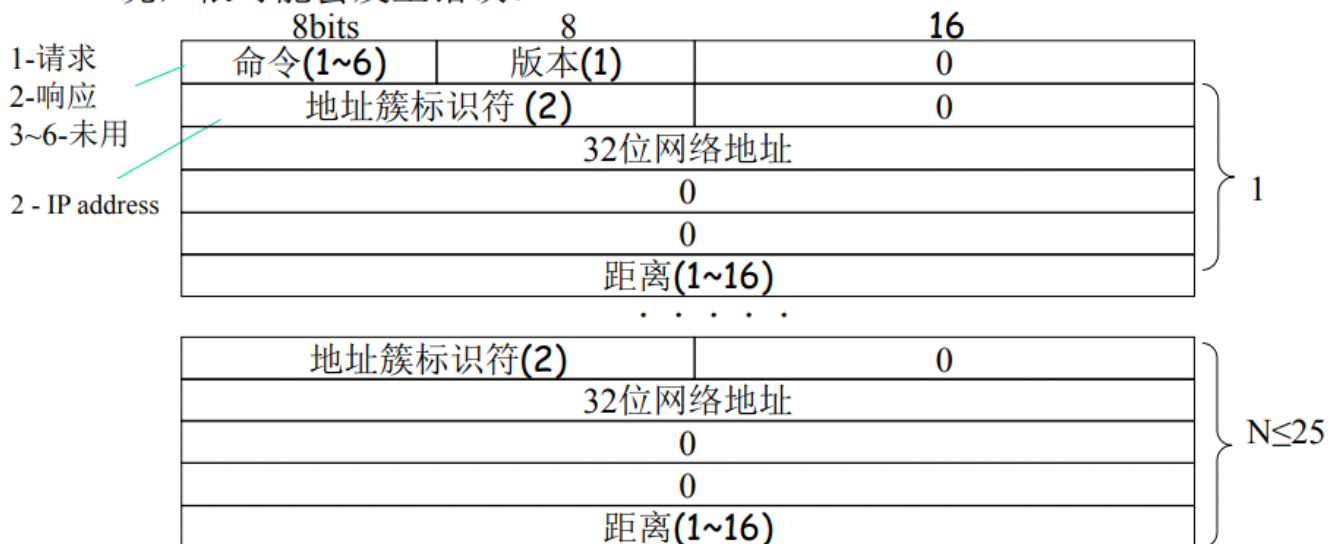
RIP协议的定时器

- ❑ 更新定时器(Update Timer)控制一个路由器如何定期把路由表发送给邻居。默认时间为30 秒。
- ❑ 一条路由的失效定时器(Invalid Timer)到期时被标记为无效路由(距离改为16)。路由被更新时其失效定时器会被重置。默认值为180秒。
- ❑ 一条路由的清除定时器(Flush Timer)到期时该路由将从路由表中删除。路由被更新时其清除定时器会被重置。默认值为240秒。
- ❑ 抑制定时器(Hold-down Timer)在路由的距离变为无穷大(包括收到毒化路由)时启动。在其到期之前不允许修改该路由的距离。默认值为180秒。



RIPv1的数据包格式

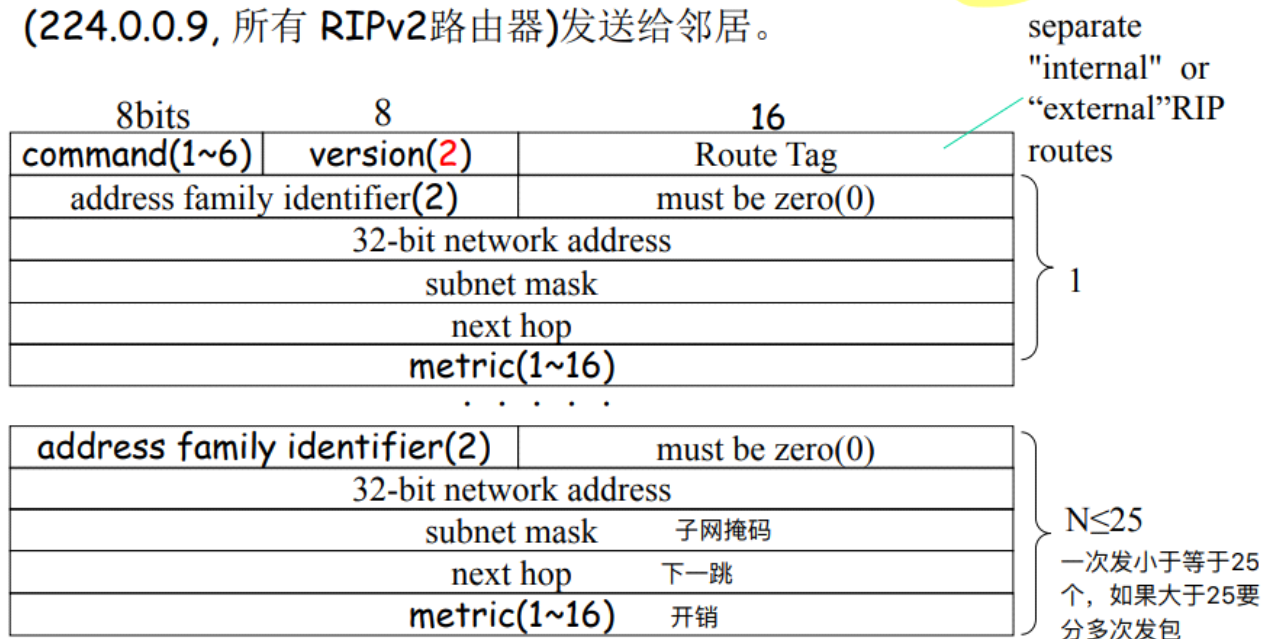
- ❑ RIPv1数据包用UDP数据报封装(端口号为520)，并且采用广播方式发送给邻居。RIPv1只能发布有类网，因此，对于存在非邻接子网的情况，很可能会发生错误。 无子网掩码



- ✓ 如果请求分组填写了网络地址, 则用距离进行响应。如果请求的网络地址为0.0.0.0, 则用整个路由表进行响应。
- ✓ 每30秒和触发更新都是发送响应分组。
- ✓ 如果项目超过25项, 则可以发送多个响应分组。

RIPv2的数据包格式(1)

- ❑ RIPv2支持无类网。RIPv2数据包可以采用广播方式或多播方式(224.0.0.9, 所有 RIPv2路由器)发送给邻居。



RIP协议的特点

- ❑ RIP协议简单、容易实现。
- ❑ 网络的直径不能超过16跳。
- ❑ 不允许把一个大网络分成多个区
- ❑ 开销缺乏灵活性
- ❑ 存在慢收敛问题和计数到无穷问题
- ❑ 每30秒发送完整路由表会消耗大量的带宽。
- ❑ 实际运行的RIP协议具有如下特性：
 - ❖ 可以保存多达6个等距离的路由在路由表中, 默认为4个。
 - ❖ 直连网的开销为0。
 - ❖ 发送的路由表会把所有距离加1。