11. Dynamic Routing Protocols

In this section, we will talk about why we might want routers to find out about destination networks and routes from each other rather than from static routing tables. And we will find out how this is accomplished. Obviously, the routers must communicate using a protocol specially designed (payload formatted) for this specific purpose.

There are two major classes of routing protocols:

- Those used within groups of networks that make up an autonomous system.
- Those used between different autonomous systems. An autonomous system will be defined later.

Readings:

Chapter 14, 16, 15.1-15.9, and 15.8 of [Comer00]

Optional:

- The other course references also have good chapters on routing protocols.
- Chapters 5 and 6 of the new book on managing IP networks with Cisco routers [Ballew97] are also very good.
11.1 The Need for Dynamic Routing

In many cases of small simple networks, static routing is adequate. In static routing, the network administrator manually manages routing tables with 'route add' commands. In addition, (s)he must do this in every router in the system, and also in many hosts (in those that have multiple rather than just a default route). For simple, small nets, this is a manageable task, and results in no added network traffic or computational overhead.

But static routing quickly becomes inappropriate in larger networks, and in networks with redundant connectivity. There are two reasons:

- It becomes a huge burden to maintain routing tables in the face of almost daily changes to the number of hosts and to the connectivity. Many networks have 1000 hosts (e.g. SFU), and I have heard of one particularly huge network that has over 1000 routers! What is worse, if the routing data in all the routers is not perfectly consistent, bad things like routing loops can occur.

- Second, in large complex networks, the connectivity is changing due to power outages, re-boots, machine failures, and software problems. It is no use having a network with connectivity that provides redundant routes, if those routes cannot be used within a few minutes of a failure (vs. the administrator having to be found, and then having to edit routing tables for dozens or thousands of machines!).

The routers in a large corporate network are very important to the overall function of the network, as they are the nodes of the topology graph.

Note: A routing protocol does not change the routing algorithm which examines the routing table and decides which way a packet should be sent. Instead, it updates the data/contents of the routing table.
11.2 Interior Routing Protocols

An interior routing protocol (IRP) is one which is used within a network or a set of physical networks which are generally administered by one organization. Note that these physical networks do not have to be subnets of one network. Nor do they have to be the same brand of network. It is not uncommon for a large institution like SFU, or a large corporation, to have several full fledged networks as part of their computer infrastructure. In essence, their intranet is an internet!

Generally, such a macro network is called an autonomous system (AS), because those who administer the backbone generally have little control over the topology and administration of the corporate macro network (except that IP addresses must be unique and hierarchical, or at least must ‘appear’ so to the rest of the world).

An AS also pretty well defines the limits of a particular routing protocol. Different ASs may run different interior routing protocols, but generally, only one particular routing protocol is run within an AS. In addition, that protocol is chosen by the organization that owns the macro network.

A routing protocol process decides:

- What kind of, and how much information to broadcast to other routers.
- Select the best route (when there are several choices) depending on some routing ‘policy’, or use load spreading.
- Determine when the best route has changed and modify the routing table appropriately.

Hosts often do not run a routing protocol. Either they only have a default route and redundant alternate routes are not an issue. Or there is maybe just one other route to one other company network (not an alternate route to same network as default). In either case, these can simply be added manually/statically. Contrast this with a router in the middle of a huge multi-segment network like SFU.

A host can run a routing protocol daemon, but if so, generally runs it in what is called quiet (-q) mode. Such a daemon listens for routing protocol information and if it hears any, modifies its routing table appropriately. But it does not broadcast any routing information because, generally, a host does not have any interesting topology information to broadcast (or if it does, a nearby router probably also has that information and will broadcast it). So, in summary:

- a host or subnet with one connection out to the rest of the macro or Internet can and probably should just use static routing tables.
- a machine (host or router) with a single gateway to the macro or Internet, but with connections to several internal routers and subnets, could use static or dynamic routing.
- a machine with multiple gateways to the macro or Internet will typically be configured to use a routing protocol.

Dynamic routing is desirable when you want automatic adjustment of routes to avoid traffic congestion, or to handle topology changes (either as a result of new or changed installations, or due to failures and power downs).
11.3 RIP

Though there are some historical routing protocols, the oldest protocol still in widespread use is called Routing Information Protocol (RIP). RIP Version 1 will be discussed first, then the enhancements that have recently been put into Version 2.

RIP is a distance vector protocol, in that it uses a crude approximation of distance as the cost that the RIP process tries to minimize. This crude approximation is the hop count. Unfortunately, minimizing the hop count is silly if the route that minimizes that count has a link which is only 1200 bps.

Other drawbacks of RIP are:

• it is limited to a maximum of 15 hops. This definitely reduces its applicability to medium or smaller macro nets.
• it only chooses a single best route, rather than being able to load share.

Nonetheless RIP is in widespread use because of its:

• simplicity.
• low network capacity use (can you remember 300 bps modems?).
• low CPU utilization.
• and the fact that the traditional IP routing algorithm only uses the first single matching route in the table anyway.

11.3.1 RIP Protocol Details

A router running the RIP daemon named ‘routed’ broadcasts every 30 seconds to port UDP port 520 the destination networks it knows about, and their hop metric, in a frame format shown in [Comer95]. Each frame has a 4 byte command and version, followed by a repeating group which forms a list of destinations.

The two main commands are request (1) and response (2). This allows solicitation of updates if you need them urgently (e.g. a route went down, and you didn’t retain the alternate so you must solicit it).

RIP is not limited to 32 bit IP addresses. It can broadcast addresses up to 14 octets long (unfortunately no enough for IPv6!). In addition, it has an address family field in each group to designate what kind of address (e.g. IP, DECnet, XNS) is being broadcast in that group.

A maximum of 25 destinations is broadcast in any one UDP packet to keep below the 512 byte limit which causes fragmentation.
11.3.2 The ‘routed’ Routing Daemon

When a RIP update is received, the ‘routed’ daemon process examines each destination group in the broadcast and:

- If the group is for a new destination not previously heard of, ‘routed’ adds the destination to the routing table with a metric one bigger than what was received (why?).
- If the group describes a destination already in the routing table, the new route is used only if when one is added to the received metric, the result is less than the metric already in the routing table for that destination.
- A routing table entry is deleted when it has not been heard about in the last 3 minutes. It is also deleted if a received group after adding 1 has a metric of 16 (which is regarded as infinity).

Notice now how the metric is used. Supplying a metric with a static ‘route add’ command doesn’t really accomplish much. But when a routing protocol is being used, the routing update process uses the metrics to decide whether to change the route to a destination, and it then updates the metric also.

‘routed’ is generally started at boot time. It reads the file /etc/gateways to find out at least some starting default gateway. A default entry in this file might look like:

```
net 0.0.0.0 gateway 128.66.12.1 metric 1 active
```

You can also add hosts, and declare the route passive (which prevents RIP from ever deleting it). There is more information on setting up RIP and other protocols on Cisco routers in [Ballew97].

11.3.3 RIP V.1’s Drawbacks

As previously mentioned, RIP:

- uses a crude approximation to cost (hop count).
- very unfortunately, it does not support broadcasting of subnet masks associated with destination networks.
- it is limited to a maximum of 15 hops. This definitely reduces its applicability to medium or smaller macro nets.
- it only chooses a single best route, rather than being able to load share (though according to [Ballew97], Cisco routers will do load balancing within their routing tables).
- it is relatively slow (10s of minutes!) to converge on start up or after a topology change, even if programmed very carefully to minimize the possibility of transient loops. All RIP implementations have an infinity of 16 to eliminate some count to infinity problems. [Comer95] and other texts describe why you should also program split horizon and poison reverse to help convergence.
- it has no authentication. In essence, someone from outside of your network can broadcast routing table changes to your network (using net or subnet directed broadcasts) which will:
  - disrupt your net causing a denial of service to your institution.
  - possibly cause proprietary data to flow toward the hacker rather than toward its intended destination.
11.4 RIP Version 2

Each destination group in Version 2 (RFC 1388 published in 1993) also carries:

• subnet mask of destination.
• routing domain AS number.
• tag to facilitate some exterior gateway protocol stuff.
• actual address of next hop gateway.
• optional authentication.

If the first (20 byte) destination group’s first 4 bytes is FFFF0002, the next 16 bytes are an authenticating password transmitted in clear text. If RIP broadcasts never get outside of your AS, then a hacker would not likely be able to guess the password which could disrupt your network routing.

Obviously these are some important improvements, and you should always endeavor to use Version 2 when possible.

11.5 OSPF

The newest interior routing protocol is called Open Shortest Path First (OSPF). In 1991, Version 2 was described in RFC 1247. In 1993, a new router requirement RFC came out which suggested that all machines that support a routing protocol must implement both RIP and OSPF. This doesn’t mean it must run both, only that both be available. I suspect this is to try and force the deployment of routers that will be ready to handle the new IPv6 which has 128 bit long addresses plus many other features that an OSPF router can take advantage of.

OSPF is a link state (and cost) protocol. Instead of broadcasting complete routes to destinations, it broadcasts only information about whether its direct links (to other routers?) are in the up state. In addition it also broadcasts a cost to be associated with each outgoing link (a very slow link can be assigned a very high pseudo-cost). OSPF broadcasts, in essence, a picture of its immediate neighborhood and the neighborhood path performance.

The receiver of such information from a variety of routers can build a complete topological map of the net, determine least cost routes and thus suitably create/modify a routing table.

OSPF advantages:

• It handles subnet masks.
• Most importantly, because of the link rather than route information, after a topology change OSPF converges much faster and with fewer problems. The drawback is
the construction and maintenance of the topological map is more computationally burdensome.

- If the routing algorithm can handle it, OSPF will recommend load sharing for equal cost routes.

- OSPF can build separate routes for each separate IP Type of Service (TOS) (i.e. low delay vs. high throughput vs. high reliability).

- Can handle serial links that do not necessarily have an address assigned to each end.

- Has clear text passwords like RIP V.2.

- Is a special Layer 4 protocol. It does not rely on TCP or UDP.

- It uses multicasting rather than broadcasting which apparently reduces the load on stations not participating in OSPF.

- And as mentioned previously, it is geared to handling IPv6.

11.5.1 Configuring OSPF and ‘gated’

The Unix daemon used to run OSPF is called ‘gated’. ‘gated’ is actually a kind of super routing daemon in that it can run several protocols at once (for each of the routers various links) and integrate the information received by all of them to update the routing table. In addition, I suspect it can broadcast routing information appropriate to each link’s routing protocol, based on information it has gained from all the other links.

The latest version of ‘gated’ I have read about is Version 3, which contains RIP, RIPv2, OSPF, and exterior routing protocols EGP and BGPv3 (which we will study shortly). There is probably by now an even newer version of ‘gated’ available.

Configuring ‘gated’ for such a router attached to multiple routing protocols is not trivial. But for just OSPF it is probably not much harder than configuring RIP. Thus most new installs of a routing daemon use ‘gated’, because of its ability to support the newer and more capable OSPF protocol, and its flexibility and interoperability. See [Hunt92] for about 14 pages of ‘gated’ configuration tutorial.

For an example of how to configure OSPF on Cisco routers, you can optionally see [Ballew97] pages 134-142. There is also a bit about configuring RIP on a Cisco router, including how to handle variable length subnet masks (note: Ballew recommends all such sub-subnets be attached directly to one router/gateway). Ballew also show how to lie about variable length subnet masks using only classful protocols like RIP, and about configuring backup and dial-up routes.
11.6 **EIGRP**

Enhanced Interior Gateway Routing Protocol (EIGRP) is a proprietary Cisco routing protocol. It has been proven on huge actual networks containing over 1000 Cisco routers. It can apparently be used as a routing protocol on nets running IP and non-IP internet protocols, thus eliminating the need to run multiple routing protocols on a network that runs multiple kinds of internet protocols!

Note that [Ballew97] has a good table comparing the features and costs of various interior gateway protocols.

11.7 **Exterior Gateway Protocols**

Exterior gateway protocols exchange reachability information between Autonomous Systems. Unlike OSPF, and like RIP, they are basically distance vector protocols. But, apparently, full paths are included so that collectors of exterior routing information can construct a complete inter-AS graph without any loops.

You might run an exterior gateway protocol between two very large universities which are separate ASs. In contrast, you would run an IGP between two campuses of the same university (i.e. within administratively common macro network).

If you didn't need to apply to the INTERNIC for an AS number (this is sort of like applying for an IP network address or DNS domain name), then you probably don't need to run an exterior gateway protocol. Your ISP probably will do this on your behalf. ISPs manage a number of networks and thus form a kind of macro network. They in turn connect to huge communications infrastructure companies like Bell, MCI, and Sprint.

One of the nice things about more recent exterior routing protocols is that they are policy-based. Most importantly, this allows communication infrastructure companies to define policies independent of the Internet backbone policy.

We need a few definitions to understand the concept of AS policy:

- A stub AS is connected to only one other AS (possibly, but not necessarily the Internet backbone).
• A multi-homed AS is connected to multiple ASs but refuses transit traffic.

• A transit AS will provide transit services and possibly charging/accounting.

One goal of an external gateway protocol is to allow policy-based routing at the periphery of an AS. Policies rules can be incorporated that, say, forbid transit traffic. These policies are often of an economic, security, or political form. Interior protocols, you realize, do not understand the concept of AS transit policy.

If you have to run an exterior routing protocol, the choice of which exterior gateway protocol (EGP or BGP) is not usually yours. For two ASs to exchange routing information, they must use the same exterior gateway protocol. Often the AS you connect to is run by a huge conglomerate like MCI or Sprint or a Bell telephone company. You must use whatever exterior protocol they are using!

Note it is possible to use an interior routing protocol as an exterior routing protocol. In fact, I believe that BC Telephone has asked SFU to do this (for reasons they don’t seem very willing to discuss). They are suggesting we run OSPF.

11.8 **EGP**

An early exterior gateway protocol was, somewhat unimaginatively, called Exterior Gateway Protocol (EGP).

EGP only announces the nets within an AS. It does not (even if multi-homed) announce other ASs that it can reach. It received information on all other ASs that can be reached; it just doesn’t re-broadcast this information.

Apparently, if you were only running a stub AS or a multi-homed AS (as defined in the previous section), then EGP was adequate. Most new installs needing an exterior protocol now choose to install BGP though.
11.9 **BGP**

The Border Gateway Protocol (BGP) is a more recent exterior gateway protocol. BGP is described in RFC 1771. Version 4 of BGP supports CIDR.

BGP uses TCP. Any two routers that have opened a TCP connection for the purpose of exchanging exterior routing information are known as peers or neighbors.

BGP peers initially exchange information describing all the ASs they know about, and the routes to each one. Each uses this information to independently construct a loop free map of ASs. Thereafter, BGP peers send incremental updates only (plus keep-alive and minor administrative/error information). BGP typically exchange keep-alive information about every 30 seconds.

The algorithm BGP uses to decide or update the best path to a destination, given information from two other routers, mainly uses distance vector information. But the decision process can also apparently take into account a number of other attributes called:

- AS_path
- Origin
- Next Hop
- Weight
- Local Preference
- Multi-Exit Discriminator
- Community

There is a large and good tutorial paper available online from the Cisco web site on BGP. See:


The above mentioned Cisco online tutorial contains information on each of these additional attributes used by the inter-AS routing update process.

This tutorial also interestingly points out that it is possible to run BGP in the interior of an AS. I think this is done so exterior routing information can cross a transit AS to an external gateway on the other side of an AS.
11.10 IDRP

IDRP (which I believe stands for Internet Domain Routing Protocol) is a new exterior protocol that is design for IPv6. I don't have much information on IDRP, but I am sure a search for it on the internet will turn up lots (e.g. RFCs, Cisco papers, 6Bone white papers?).

So for IPv6, you will likely want OSPF for your interior routing protocol, and IDRP for your exterior routing.

11.11 References


