Network Working Group
INTERNET-DRAFT

Philip Almquist
Stanford University/BARRNet
July 1991

Ruminations on Route Leaking

Status of This Memo

status per RFC1111, to be filled in by the RFC Editor (This is an informational document; it does not propose a standard, although the author hopes that ideas contained herein will influence future standards).

This is an as yet incomplete draft, dated July 23, 1991 2:32 am. A final version is (hopefully) forthcoming.

Distribution of this memo is unlimited.

Summary

This memo discusses how routers can “leak” routing information learned from one routing domain into another routing domain.

Although this memo is not an official product of the Router Requirements Working Group, it is being published as an INTERNET DRAFT because it is required reading for those who wish to actively participate in the Router Requirements sessions during the IETF meeting in Atlanta.

Acknowledgments

Vince Fuller, Toni Li, and Bob Albrightson provided useful comments on an early partial draft of this memo.

Contents

1. Introduction 3
   1.1. Terminology 3
   1.2. Overview of the Issues 3
   1.3. Overview of this Paper 4

2. Models of Route Leaking 4
   2.1. Model A: Leaking the Routes in a Routing Domain 4
   2.2. Model B: Leaking the Forwarding Table 4
   2.3. The Combination Model 5
   2.4. Leaking Using Third Party Advertisements 5

3. Classification of Routes 5

4. Issues in Route Leaking 6
   4.1. Route Summarization 6
   4.2. Controls on Route Leaking 7
   4.3. TOS 8
   4.4. Route Classes 8
   4.5. Synthesis of Route Attributes 8
      4.5.1. Metrics 8
      4.5.2. Other Route Attributes 9

References 11

Security Considerations 12

Author's Address 12

1. Introduction

Many times, a router has multiple sources of information about paths through the network. For example, a router may be simultaneously learning routes from multiple routing protocols and may also have static routes configured by the network manager (or, more precisely, the router may belong to multiple routing domains). A companion paper, *Ruminations on the Next Hop* [1], discusses how a router picks routes in such a situation. This paper considers the related issues of how a router passes on (through its routing protocols) routes that it learns. I recommend that you read the other paper first.

1.1 Terminology

This paper uses the following technical terms:

Forwarding table A forwarding table is the set of routes which a router uses for forwarding packets. This is usually a subset of the routes available to the router, because the forwarding table contains only the best route to each destination.

Routing domain A routing domain is a collection of routers which coordinate their routing knowledge using a single (instance of) a routing protocol. A degenerate case is a single router which has some statically configured routes. The Internet community often uses the term Autonomous System instead, but I have avoided that term in this paper because it is often also used to refer to other similar concepts.

1.2 Overview of the Issues

1. Controls needed
2. Route summarization (e.g. collapsing subnet routes into net routes).
3. Transformations to account for differences in route lookup algorithms:
   - TOS
   - route classes (intra-area, inter-area, external)
   - subnet support (variable length, fixed length, none)
   - metric transformation/setting
4. Issues of routing stability (e.g. split horizon) in route leaking.

1.3 Overview of this Paper

2. Models of Route Leaking

There are several ways of looking at exactly what route leaking is. This section contrasts two extreme views, followed by a third which seems correct to me.

2.1 Model A: Leaking the Routes in a Routing Domain

Under this view, route leaking consists of the bulk transfer of routes known in a source routing domain into a destination routing domain. Although policy mechanisms may restrict which routes are leaked, routes are leaked without regard as to whether they are in the router's forwarding table. This might occur, for example, if the router preferred a route from a third routing domain to the leaked route, or if the router was prevented by policy mechanisms from believing (using) the route being leaked.

To me, at least, this view has a certain formal elegance. It also would seem a good basis for certain sorts of policy-based routing, since it implicitly recognizes that all routes to a destination may not be equivalent.

However, this view has some less desirable consequences. One is what I call "diversion": a router may leak one route to a particular destination but use an altogether different route to reach the destination. This may merely result in counterintuitive routing, or it may have more serious consequences, such as violating the policy of the network manager. Related but even more serious consequences are possible if the router is leaking a route but has no route to the destination in its forwarding table: the router may black hole the packets or, if it has a default route, may route the packets in a way that causes looping.

2.2 Model B: Leaking the Forwarding Table

Under this view, route leaking consists of dumping the contents of the forwarding table into a specified destination routing domain. Again, there may be controls over the destinations for which routes may be leaked, but under this view the router doesn't consider the source routing domain when deciding whether to leak the route.

This view avoids the problems of looping and black holes that can result from Model A. Likewise, diversion cannot occur because only routes which the router will actually use are leaked.

However, this view also has less desirable consequences. The most obvious is that it does not support leaking based on the source of the route. In situations where policy restricts use of certain paths to certain communities, routes to the same destination may need to be treated differently. A related problem is that it may be unsafe to leak a particular route when it comes from a particular routing domain,

since leaking it will cause a routing loop through multiple routing domains. With this view, it is easy for a network manager to set up routing which appears to work but which creates a persistent routing loop when the primary path to a particular destination fails.

2.3 The Combination Model

Under this view, route leaking is similar to Model A, except that only the subset of the routes from the source routing domain which are also in the router's forwarding table are leaked. This gives this approach the advantages of Model A, but avoids its pitfalls by advertising only those routes which the router actually uses.

2.4 Leaking Using Third Party Advertisements

Most routing protocols only allow a router to advertise routes through the router itself. Some routing protocols (such as EGP [6] and OSPF [7]) provide mechanisms to allow a router to specify that some other router should be used as the path to a particular destination. These mechanisms can be collectively referred to as "third party advertisements".

The pitfalls of Model A do not affect routes leaked as third party advertisements. Thus, it is appropriate to leak routes which are not in the forwarding table in cases where they can be leaked as third party advertisements.[1]

3. Classification of Routes

To use a number of the mechanisms described later in this paper, it is useful to be able to refer to a collection of routes by specifying some property that the routes share. Fortunately for implementors, there are only a very few useful ways of classifying routes, and each is useful in most of the mechanisms I will be describing:

Address range — It is useful to specify all routes to destinations which fall (numerically) into an address range that is a member of a specified set of address ranges.

Milo's Number[2] — It is useful to specify all OSPF routes which have in their Milo's Number field any of a set of specified values.

1. Some people believe that there are cases where third party advertisements (even of routes that the router believes) can cause routing loops. I need to think about this some more. Also, it is not clear how important third party advertisements would be in practice; maybe this section should go away or discourage third party advertisements???
2. This is the four octet tag field which OSPF carries with external routes. It's use is outside of the scope of the OSPF specification; it is intended for use by border routers to communicate among themselves. Most commonly, it is used to carry the autonomous system number of the source of the route. I need to check the OSPF speck to find the real name of this field.

AS path It is useful to specify all BGP routes whose AS path "matches" any of a set of specified values. It is not yet clear exactly what kinds of matches are most useful. A simple and reasonably useful option would be to allow matching of all routes which do (or do not) have a specified AS number at some point in the AS path. A more general but somewhat more difficult alternative would be to allow matching all routes for which the AS path matches a specified regular expression.

Route class For routing protocols which maintain the distinction, it is useful to subdivide the routes into internal routes and external routes. It is also useful (though less so) to further subdivide internal routes into intra-area routes and inter-area routes, and to subdivide external routes into those that use internal metrics and those that use external metrics.

4. Issues in Route Leaking

4.1 Route Summarization

Traditionally, routing protocols have carried routes describing paths to networks. However, as the technology has evolved routing protocols have sprouted a variety of mechanisms to carry routes describing paths to individual end systems or to ranges of addresses (called subnets by the Internet community).

This is a concern in this paper for two reasons. One is that it is often desirable, in order to keep routing information to a manageable size, to replace a collection of routes with a single route which summarizes the information. For example, it may be desirable to invent and leak a route to a network instead of leaking all of the routes to that network's component subnets.

The other reason this is a concern of this paper is that different routing protocols have different subnetting capabilities. Some still support only routes to networks. Others support routes to subnets, but assume that all subnets of a network use the same address mask (which is conveyed to all routers through a mechanism outside of the routing protocol). The most flexible routing protocols either carry address ranges or carry for each route an arbitrary bit mask indicating which bits of the destination of the route are significant. Most routers which support route leaking will have to address the issue of how to leak routes between protocols with different mechanisms and different degrees of capability in this area.

I originally considered addressing this problem by creating a matrix, where one axis described the capability of the protocol that was the source of the route, and the other axis described the capability of the protocol into which the route was being leaked. The cells inthe matrix described the action the router should take for each

combination of capabilities. This was an interesting exercise, but I rapidly rejected that approach because the complexity rapidly got out of hand. Even if I did succeed in describing all possible cases, and even if nobody made my list obsolete by inventing a routing protocol with different mechanisms, It seemed unlikely that most network managers would be able to guess what their router would really do without carefully pondering a copy of my matrix.

I therefore opted for a simpler approach, inspired in part by the route summarization mechanisms in OSPF [7]. This approach also has the advantage that it handles in a theoreticaaly elegant way the traditionally difficult question of when a router should generate a default route (rather than leaking all of the individul routes).[1]

In this approach, a network manager may configure a set of a address ranges. Associated with each address range is a destination address (and an address mask). If a router is considering leaking a particular route, it checks to see if it is a route to a destination which falls within one of the address ranges. If so, it instead creates and leaks a route to the destination address associated with the address range which was matched.

If, on the other hand, the route does not fall within any of the address ranges, the router does one of two things. If the protocol into which the route is being leaked has sufficient capabilities that it can accurately represent the destination of the route, then the route is leaked. Otherwise, the route is not leaked. Thus, for example, a route to a subnet would be leaked into OSPF, since it can represent subnets, but would not be leaked into EGP, since it cannot carry subnet routes. A subnet route from OSPF would be leaked into RIP if and only if the mask associated with the OSPF route was the same as the implicit address mask used by the RIP routing domain

4.2 Controls on Route Leaking

Route leaking is vital part of creating non-local connectivity in the Internet. However, it is a two-edged sword, and can create numerous problems if routes are simply leaked with abandon. Thus, any route which supports route leaking also needs to support mechanisms to allow the network manager to control which routes go where.

At the very least, a router must not leak any routes from one routing domain to another unless it has been specifically configured by the network manager to do so. This prevents a partially-configured router from causing major havoc.

1. It has been pointed out that although this may be technically elegant, configuring the route leaking mechanism to generate a default route may be less intuitive to the uninitiated than the existing “originate default” commands. However, I suspect that could be fixed though appropriate user interface design.

In practice, finer-grained controls are often needed. The system of classification of routes described in Section 3. on page 5 provides a useful shorthand to allow the network manager to describe which routes ought to be leaked from a particular source routing domain to a particular destination routing domain.

4.3 TOS

TBD - basically:

- When leaking from a protocol which does not support TOS to one that does, leaked routesare assigned the default TOS.
- When leaking from a protocol which does support TOS to one that does not, routes which have a TOS other than the default TOS are not leaked.
- When leaking from a protocol which supports all TOS values into Dual IS-IS, routes which have a TOS not supported by Dual IS-IS are discarded.

4.4 Route Classes

TBD - basically:

- When leaking into a protocol which supports route classes (intra-area, inter-area, etc.), the protocol specifies rules for choosing the route class. Where an option is provided, there needs to be a configuration option (and once again, the route classification system from section 3 is useful here)
- Leaking from a protocol that supports route classes into one that doesn't presents no special challenges as long as the router follows the rule that routes not in the forwarding table are never leaked (this rule is particularly important here since routing protocols which use route classes don't choose the longest match if there is an acceptable route with a more preferable route class than the longest match route).

4.5 Synthesis of Route Attributes

4.5.1 Metrics

Metrics pose a particular problem in route leaking, since metrics are generally meaningful only within a routing domain, and different routing protocols often have very different types of metrics.

There are two basic approaches to setting metrics in leaked routes. The first is the *metric translation* approach: the metric from the source routing domain is passed through a mathematical function (often represented in practice as a translation table) to obtain the metric in the destination routing domain. This approach has some definite advantages if the translation function is well-chosen: the metric represents on some ways the quality of the route, and "counting to infinity" can be

depended upon to suppress routing loops that pass through multiple routing domains. The disadvantage of this approach is that the variations in the range and semantics of metrics between the different routing protocols has left the choice of appropriate translation functions as an unsolved problem (the major exception I am aware of is a standard function [9] for translating between the metrics used by RIP [3] and the metrics used by the now obsolete HELLO protocol [8]).

Therefore, the recommended approach is the *static metric* approach: the metric used in the destination routing domain is established by a configuration option. This approach is somewhat dangerous, in that an incautious network manager can set things up such that stable routing loops can form, but many network managers have successfully used this approach.

In many cases, it is important that the network manager be able to establish different metrics for different routes that are leaked. For example, the network manager may wish that the destination routing domain use the some of the leaked routes as primary paths to their destinations while other of the leaked routes are backup paths to their destinations. That may be impossible to configure if all of the routes have to be leaked with the same metric.

Although it is desirable to allow the network manager to establish metrics individually for each route leaked, it would also be undesirable to force him or her to do so, since that might require the network manager to provide a great deal of configuration information. Therefore, the recommended approach is to allow the network manager to establish metrics for categories of routes, using the categories defined in Section 3. on page 5, and also a default metric for use with routes which don't fall into any of the categories. This approach allows the network manager to provide only the bare minimum of configuration information necessary to set the metrics required by his or her policies. It is probably wise to force the network manager to explicitly choose the value for the defaut metric, since the implementor has no way to algorithmically choose a value which is likely to work correctly in any particular network.

4.5.2 Other Route Attributes

Some routing protocols carry with them various protocol-specific route attributes. Although this paper tries hard to solve the problems of route leaking in a generic (non-protocol-specific) manner, this is one place where discussion of particular protocols cannot be avoided. Unfortunately, that also means that this section is incomplete, either immediately because of protocols I haven't thought about or in time because of new protocols which will undoubtedly be invented.

The BGP protocol [4],[5] carries with its routes an "AS Path" attribute, which is a list of the autonomous system numbers of the routing domains on the path to the destination of the route. When leaking a route into BGP, it is necessary to create this attribute. Because it is in general not possible to know the autonomous system

numbers of all of the routing domains back to the source of the route, usually all that is possible is to include in the attribute a single autonomous system number which is the autonomous system number of the routing domain from which the route is being leaked. If the source routing domain is also using the BGP protocol, the AS path attribute should be copied from the source routing domain and the autonomous system number of the source routing domain should be appended.[1]

The OSPF protocol [7] also has an attribute to describe the source of an external route. The contents of this attribute, called “Milo’s Number”, are (intentionally) left to local convention in the specification. Thus, an implementation probably ought to be able to fill in the Milo’s number attribute of a leaked route with the autonomous system number of the source routing domain. However, in case another convention is in use in the destination OSPF routing domain, an implementation should allow static configuration of the value for this attribute. As with metrics, it may be important to allow different values to be specified for different groups of routes, again using the route classification scheme described in Section 3. on page 5. For the special case where both the source routing domain and the destination routing domain use OSPF, the implementation should be able to copy the value of Milo’s Number from the original route into the leaked route if the source route is also an external route.[2]

1. Theoretically, this is not an issue because (according to those who know more about BGP than I do) it makes no sense for a router to ever run more than a single instance of BGP.
2. At the Spring ‘91 IETF meeting there was some discussion of semi-standard values that could be used for Milo’s number when leaking BGP routes into OSPF. I need to find my notes and perhaps ask Yakov about the future of that proposal.

References

[1] P. Almquist
Ruminations on the Next Hop
Paper in progress.

[2] R. Callon
Use of OSI IS-IS for Routing in TCP/IP and Dual Environments
Request for Comments (RFC) 1195, DDN Network Information Center, SRI International, Menlo Park, California, USA, December 1990.

[3] C.L. Hedrick
Routing Information Protocol
Request for Comments (RFC) 1058, DDN Network Information Center, SRI International, Menlo Park, California, USA, June 1988.

[4] J.C. Honig, D. Katz, M. Mathis, Y. Rekhter, and J.Y. Yu
Application of the Border Gateway Protocol in the Internet
Request for Comments (RFC) 1164, DDN Network Information Center, SRI International, Menlo Park, California, USA, June 1990.

[5] K. Lougheed and Y. Rekhter
Border Gateway Protocol (BGP)
Request for Comments (RFC) 1163, DDN Network Information Center, SRI International, Menlo Park, California, USA, June 1990.

[6] D. Mills
Exterior Gateway Protocol Formal Specification
Request For Comments (RFC) 904, DDN Network Information Center, SRI International, Menlo Park, California, USA, April 1984.

[7] J. Moy
OSPF Specification, Version 2
To be published as an RFC; currently an INTERNET DRAFT dated June 1990.

[8] D.L. Mills
DCN Local-network Protocols
Request for Comments (RFC) 891, DDN Network Information Center, SRI International, Menlo Park, California, USA, December 1983.

[9] ??? (Gated reference)

Security Considerations

Security considerations are not addressed in this memo.

Author's Address

Philip Almquist
214 Cole Street, Suite 2
San Francisco, CA 94117-1916

Phone: 415-752-2427

EMail: almquist@Jessica.Stanford.EDU