Network Working Group
Request for Comments: 1245

J. Moy, Editor
Proteon, Inc.

July 1991

OSPF protocol analysis

Status of this Memo

This memo provides information for the Internet community. It does not specify any Internet standard. Distribution of this memo is unlimited.

Abstract

This is the first of two reports on the OSPF protocol. These reports are required by the IAB/IESG in order for an Internet routing protocol to advance to Draft Standard Status. OSPF is a TCP/IP routing protocol, designed to be used internal to an Autonomous System (in other words, OSPF is an Interior Gateway Protocol).

Version 1 of the OSPF protocol was published in RFC 1131. Since then OSPF version 2 has been developed. Version 2 has been documented in RFC 1247. The changes between version 1 and version 2 of the OSPF protocol are explained in Appendix F of RFC 1247. It is OSPF Version 2 that is the subject of this report.

This report attempts to summarize the key features of OSPF V2. It also attempts to analyze how the protocol will perform and scale in the Internet.

Please send comments to ospf@trantor.umd.edu.

Table of Contents

1.0 Introduction

This document addresses, for OSPF V2, the requirements set forth by the IAB/IESG for an Internet routing protocol to advance to Draft Standard state. This requirements are briefly summarized below. The remaining sections of this report document how OSPF V2 satisfies these requirements:

- What are the key features and algorithms of the protocol?
- How much link bandwidth, router memory and router CPU cycles does the protocol consume under normal conditions?
- For these metrics, how does the usage scale as the routing environment grows? This should include topologies at least an order of magnitude larger than the current environment.
- What are the limits of the protocol for these metrics? (i.e., when will the routing protocol break?)
- For what environments is the protocol well suited, and for what is it not suitable?

1.1 Acknowledgments

The OSPF protocol has been developed by the OSPF Working Group of the Internet Engineering Task Force.

2.0 Key features of the OSPF protocol

This section summarizes the key features of the OSPF protocol. OSPF is an **Internal gateway protocol**; it is designed to be used internal to a single Autonomous System. OSPF uses **link-state or SPF-based** technology (as compared to the distance-vector or Bellman-Ford technology found in routing protocols such as RIP). Individual **link state advertisements (LSAs)** describe pieces of the OSPF routing domain (Autonomous System). These LSAs are flooded throughout the routing domain, forming the **link state database**. Each router has an identical link state database; synchronization of link state databases is maintained via a **reliable flooding algorithm**. From this link state database, each router builds a routing table by calculating a shortest-path tree, with the root of the tree being the calculating router itself. This calculation is commonly referred to as the **Dijkstra procedure**.

Link state advertisements are small. Each advertisement describes a small pieces of the OSPF routing domain, namely either: the neighborhood of a single router, the neighborhood of a single transit network, a single inter-area route (see below) or a single external route.

The other key features of the OSPF protocol are:

- **Adjacency bringup**. **Certain pairs of OSPF routers become "adjacent"**. As an adjacency is formed, the two routers synchronize their link state databases by **exchanging database summaries** in the form of OSPF Database Exchange packets. Adjacent routers then maintain synchronization of their link state databases through the **reliable flooding algorithm**. Routers connected by serial lines always become adjacent. On multi-access networks (e.g., ethernets or X.25 PDNs), all routers attached to the network become adjacent to both the Designated Router and the Backup Designated router.
- **Designated router.** A Designated Router is elected on all multi-access networks (e.g., ethernets or X.25 PDNs). The network's Designated Router **originates the network LSA** describing the network's local environment. It also plays a **special role in the flooding algorithm**, since all routers on the network are synchronizing their link state databases by sending and receiving LSAs to/from the Designated Router during the flooding process.
- **Backup Designated Router**. A Backup Designated Router is elected on multi-access networks to speed/ease the transition of Designated Routers when the current Designated Router disappears. In that event, the Backup DR takes over, and does not need to go through the adjacency bringup process on the LAN (since it already had done this in its Backup capacity). Also, even before the disappearance of the Designated Router is noticed, the Backup DR will enable the reliable flooding algorithm to proceed in the DR's absence.

- **Non-broadcast multi-access network support.** OSPF treats these networks (e.g., X.25 PDNs) pretty much as if they were LANs (i.e., a DR is elected, and a network LSA is generated). Additional configuration information is needed however for routers attached to these network to initially find each other.
- **OSPF areas**. OSPF allows the Autonomous Systems to be broken up into regions call areas. This is useful for several reasons. First, it provides an extra level of **routing protection**: routing within an area is protected from all information external to the area. Second, by splitting an Autonomous System into areas the **cost of the Dijkstra procedure** (in terms of CPU cycles) is reduced.
- **Flexible import of external routing information.** In OSPF, **each external route** is imported into the Autonomous System in **a separate LSA**. This reduces the amount of flooding traffic (since external routes change often, and you want to only flood the changes). It also enables **partial routing table updates** when only a single external route changes. OSPF external LSAs also provide the following features. A **forwarding address** can be included in the external LSA, eliminating extra-hops at the edge of the Autonomous System. There are two levels of external metrics that can be specified, **type 1** and **type 2**. Also, external routes can be tagged with a 32-bit number (the **external route tag**; commonly used as an AS number of the route's origin), simplifying external route management in a transit Autonomous System.
- **Four level routing hierarchy**. OSPF has a four level routing hierarchy, or trust model: **intra-area**, **inter-area**, **external type 1** and **external type 2** routes. This enables multiple levels of routing protection, and simplifies routing management in an Autonomous System.
- **Virtual links**. By allowing the configuration of virtual links, OSPF **removes topological restrictions** on area layout in an Autonomous System.
- **Authentication of routing protocol exchanges**. Every time an OSPF router receives a routing protocol packet, it authenticates the packet before processing it further.
- **Flexible routing metric.** In OSPF, metric are assigned to outbound router interfaces. The cost of a path is then the sum of the path's component interfaces. The routing metric itself can be assigned by the system administrator to indicate any combination of network characteristics (e.g., delay, bandwidth, dollar cost, etc.).
- **Equal-cost multipath.** When multiple best cost routes to a destination exist, OSPF finds them and they can be then used to load share traffic to the destination.
- **TOS-based routing.** Separate sets of routes can be calculated for each IP type of service. For example, low delay traffic could be routed on one path, while high bandwidth traffic is routed on another. This is done by (optionally) assigning, to each outgoing router interface, one metric for each IP TOS.
- **Variable-length subnet support.** OSPF includes support for variable-length subnet masks by carrying a network mask with each advertised destination.

- **Stub area support.** To support routers having insufficient memory, areas can be configured as stubs. External LSAs (often making up the bulk of the Autonomous System) are not flooded into/throughout stub areas. Routing to external destinations in stub areas is based solely on default.

3.0 Cost of the protocol

This section attempts to analyze how the OSPF protocol will perform and scale in the Internet. In this analysis, we will concentrate on the following four areas:

- **Link bandwidth**. In OSPF, a reliable flooding mechanism is used to ensure that router link state databases are remained synchronized. Individual components of the link state databases (the LSAs) are refreshed infrequently (every 30 minutes), at least in the absence of topological changes. Still, as the size of the database increases, the amount of link bandwidth used by the flooding procedure also increases.
- **Router memory**. The size of an OSPF link state database can get quite large, especially in the presence of many external LSAs. This imposes requirements on the amount of router memory available.
- **CPU usage**. In OSPF, this is dominated by the length of time it takes to run the shortest path calculation (Dijkstra procedure). This is a function of the number of routers in the OSPF system.
- **Role of the Designated Router.** The Designated router receives and sends more packets on a multi-access networks than the other routers connected to the network. Also, there is some time involved in cutting over to a new Designated Router after the old one fails (especially when both the Backup Designated Router and the Designated Router fail at the same time). For this reason, it is possible that you may want to limit the number of routers connected to a single network.

The remaining section will analyze these areas, estimating how much resources the OSPF protocol will consume, both now and in the future. To aid in this analysis, the next section will present some data that have been collected in actual OSPF field deployments.

3.1 Operational data

The OSPF protocol has been deployed in a number of places in the Internet. For a summary of this deployment, see [1]. Some statistics have been gathered from this operational experience, via local network management facilities. Some of these statistics are presented in the following table:

TABLE 1. Pertinent operational statistics

Statistic	BARRNet	NSI	OARnet
Data gathering (duration)	99 hours	277 hours	28 hours
Dijkstra frequency	50 minutes	25 minutes	13 minutes
External incremental frequency	1.2 minutes	.98 minutes	not gathered
Database turnover	29.7 minutes	30.9 minutes	28.2 minutes
LSAs per packet	3.38	3.16	2.99
Flooding retransmits	1.3%	1.4%	.7%

The first line in the above table show the length of time that statistics were gathered on the three networks. A brief description of the other statistics follows:

- **Dijkstra frequency.** In OSPF, the Dijkstra calculation involves only those routers and transit networks belonging to the AS. The Dijkstra is run only when something in the system changes (like a serial line between two routers goes down). Note that in these operational systems, the Dijkstra process runs only infrequently (the most frequent being every 13 minutes).
- **External incremental frequency**. In OSPF, when an external route changes only its entry in the routing table is recalculated. These are called external incremental updates. Note that these happen much more frequently than the Dijkstra procedure. (in other words, incremental updates are saving quite a bit of processor time).
- **Database turnover.** In OSPF, link state advertisements are refreshed at a minimum of every 30 minutes. New advertisement instances are sent out more frequently when some part of the topology changes. The table shows that, even taking topological changes into account, on average an advertisement is updated close to only every 30 minutes. This statistic will be used in the link bandwidth calculations below. Note that NSI actually shows advertisements updated every 30.7 (> 30) minutes. This probably means that at one time earlier in the measurement period, NSI had a smaller link state database that it did at the end.
- **LSAs per packet.** In OSPF, multiple LSAs can be included in either Link State Update or Link State Acknowledgment packets.The table shows that, on average, around 3 LSAs are carried in a single packet. This statistic is used when calculating the header overhead in the link bandwidth calculation below. This statistic was derived by diving the number of LSAs flooded by the number of (non-hello) multicasts sent.
- **Flooding retransmits.** This counts both retransmission of LS Update packets and Link State Acknowledgment packets, as a percentage of the original multicast flooded packets. The table shows that flooding is working well, and that retransmits can be ignored in the link bandwidth calculation below.

3.2 Link bandwidth

In this section we attempt to calculate how much link bandwidth is consumed by the OSPF flooding process. The amount of link bandwidth consumed increases linearly with the number of advertisements present in the OSPF database.We assume that the majority of advertisements in the database will be AS external LSAs (operationally this is true, see [1]).

From the statistics presented in Section 3.1, any particular advertisement is flooded (on average) every 30 minutes. In addition, three advertisements fit in a single packet. (This packet could be either a Link State Update packet or a Link State Acknowledgment packet; in this analysis we select the Link State Update packet, which is the larger). An AS external LSA is 36 bytes long. Adding one third of a packet header (IP header plus OSPF Update packet) yields 52 bytes. Transmitting this amount of data every 30 minutes gives an average rate of 23/100 bits/second.

If you want to limit your routing traffic to 5% of the link's total bandwidth, you get the following maximums for database size:

TABLE 2. Database size as a function of link speed (5% utilization)

Speed	# external advertisements
9.6 Kb	2087
56 Kb	12,174

Higher line speeds have not been included, because other factors will then limit database size (like router memory) before line speed becomes a factor. Note that in the above calculation, the size of the data link header was not taken into account. Also, note that while the OSPF database is likely to be mostly external LSAs, other LSAs have a size also. As a ballpark estimate, router links and network links are generally three times as large as an AS external link, with summary link advertisements being the same size as external link LSAs.

OSPF consumes considerably less link bandwidth than RIP. This has been shown experimentally in the NSI network. See Jeffrey Burgan's "NASA Sciences Internet" report in [3].

3.3 Router memory

Memory requirements in OSPF are dominated by the size of the link state database. As in the previous section, it is probably safe to assume that most of the advertisements in the database are external LSAs. While an external LSA is 36 bytes long, it is generally stored by an OSPF implementation together with some support data. So a good estimate of router memory consumed by an external LSA is probably 64 bytes. So a database having 10,000 external LSAs will consume 640K bytes of router memory. OSPF definitely requires more memory than RIP.

Using the Proteon P4200 implementation as an example, the P4200 has 2Mbytes of memory. This is shared between instruction, data and packet buffer memory. The P4200 has enough memory to store 10, 000 external LSAs, and still have enough packet buffer memory available to run a reasonable number of interfaces.

Also, note that while the OSPF database is likely to be mostly external LSAs, other LSAs have a size also. As a ballpark estimate, router links and network links consume generally three times as much memory as an AS external link, with summary link advertisements being the same size as external link LSAs.

3.4 Router CPU

Assume that, as the size of the OSPF routing domain grows, the number of interfaces per router stays bounded. Then the Dijkstra calculation is of order (n * log (n)), where n is the number of routers in the routing domain. (This is the complexity of the Dijkstra algorithm in a sparse network). Of course, it is implementation specific as to how expensive the Dijkstra really is.

We have no experimental numbers for the cost of the Dijkstra calculation in a real OSPF implementation. However, Steve Deering presented results for the Dijkstra calculation in the "MOSPF meeting report" in [3]. Steve's calculation was done on a DEC 5000 (10 mips processor), using the Stanford internet as a model. His graphs are based on numbers of networks, not number of routers. However, if we extrapolate that the ratio of routers to networks remains the same, the time to run Dijkstra for 200 routers in Steve's implementation was around 15 milliseconds.

This seems a reasonable cost, particularly when you notice that the Dijkstra calculation is run very infrequently in operational deployments. In the three networks presented in Section 3.1, Dijkstra was run on average only every 13 to 50 minutes. Since the Dijkstra is run so infrequently, it seems likely that OSPF overall consumes less CPU than RIP (because of RIP's frequent updates, requiring routing table lookups).

As another example, the routing algorithm in MILNET is SPF-based. MILNET's current size is 230 nodes, and the routing calculation still consumes less than 5% of the MILNET switches' processor bandwidth [4]. Because the routing algorithm in the MILNET adapts to network load, it runs the Dijkstra process quite frequently (on the order of seconds as compared to OSPF's minutes). However, it should be noted that the routing algorithm in MILNET incrementally updates the SPF-tree, while OSPF rebuilds it from scratch at each Dijkstra calculation

OSPF's Area capability provides a way to reduce Dijkstra overhead, if it becomes a burden. The routing domain can be split into areas. The extent of the Dijkstra calculation (and its complexity) is limited to a single area at a time.

3.5 Role of Designated Router

This section explores the number of routers that can be attached to a single network. As the number of routers attached to a network grows, so does the amount of OSPF routing traffic seen on the network. Some of this is Hello traffic, which is generally multicast by each router every 10 seconds. This burden is borne by all routers attached to the network. However, because of its special role in the flooding process, the Designated router ends up sending more Link State Updates than the other routers on the network. Also, the Designated Router receives Link State Acknowledgments from all attached routers, while the other routers just receive them from the DR. (Although it is important to note that the rate of Link State Acknowledgments will generally be limited to one per second from each router, because acknowledgments are generally delayed.)

So, if the amount of protocol traffic on the LAN becomes a limiting factor, the limit is likely to be detected in the Designated Router first. However, such a limit is not expected to be reached in practice. The amount of routing protocol traffic generated by OSPF has been shown to be small (see Section 3.2). Also, if need be OSPF's hello timers can be configured to reduce the amount of protocol traffic on the network. Note that more than 50 routers have been simulated attached to a single LAN (see [1]). Also, in interoperability testing 13 routers have been attached to a single ethernet with no problems encountered.

Another factor in the number of routers attached to a single network is the cutover time when the Designated Router fails. OSPF has a Backup Designated Router so that the cutover does not have to wait for the new DR to synchronize (the adjacency bring-up process mentioned earlier) with all the other routers on the LAN; as a Backup DR it had already synchronized. However, in those rare cases when both DR and Backup DR crash at the same time, the new DR will have to synchronize (via the adjacency bring-up process) with all other routers before becoming functional. Field experience show that this synchronization process takes place in a timely fashion (see the OARnet report in [1]). However, this may be an issue in systems that have many routers attached to a single network.

In the unlikely event that the number of routers attached to a LAN becomes a problem, either due to the amount of routing protocol traffic or the cutover time, the LAN can be split into separate pieces (similar to splitting up the AS into separate areas).

3.6 Summary

In summary, it seems like the most likely limitation to the size of an OSPF system is available router memory. We have given as 10,000 as the number of external LSAs that can be supported by the memory available in one configuration of a particular implementation (the Proteon P4200). Other implementations may vary; nowadays routers are being built with more and more memory.

Note that 10,000 routes is considerably larger than the largest field implementation (BARRNet; which at 1816 external LSAs is still very large).

Note that there may be ways to reduce database size in a routing domain. First, the domain can make use of default routing, reducing the number of external routes that need to be imported. Secondly, an EGP can be used that will transport its own information through the AS instead of relying on the IGP (OSPF in this case) to do transfer the information for it (the EGP). Thirdly, routers having insufficient memory may be able to be assigned to stub areas (whose databases are drastically smaller). Lastly, if the Internet went away from a flat address space the amount of external information imported into an OSPF domain could be reduced drastically.

While not as likely, there could be other issues that would limit the size of an OSPF routing domain. If there are slow lines (like 9600 baud), the size of the database will be limited (see Section 3.2). Dijkstra may get to be expensive when there are hundreds of routers in the OSPF domain; although at this point the domain can be split into areas. Finally, when there are many routers attached to a single network, there may be undue burden imposed upon the Designated Router; although at that point a LAN can be split into separate LANs.

4.0 Suitable environments

Suitable environments for the OSPF protocol range from large to small. OSPF is particular suited for transit Autonomous Systems for the following reasons. OSPF can accommodate a large number of external routes. In OSPF the import of external information is very flexible, having provisions for a forwarding address, two levels of external metrics, and the ability to tag external routes with their AS number for easy management. Also OSPF's ability to do partial updates when external information changes is very useful on these networks.

OSPF is also suited for smaller, either stand alone or stub Autonomous Systems, because of its wide array of features: fast convergence, equal-cost-multipath, TOS routing, areas, etc.

5.0 Unsuitable environments

OSPF has a very limited ability to express policy. Basically, its only policy mechanisms are in the establishment of a four level routing hierarchy: intra-area, inter-area, type 1 and type 2 external routes. A system wanting more sophisticated policies would have to be split up into separate ASes, running a policy-based EGP between them.

6.0 Reference Documents

The following documents have been referenced by this report:

[1] Moy, J., “Experience with the OSPF protocol”, RFC 1246, July 1991.

[2] Moy, J., “OSPF Version 2”, RFC 1247, July 1991.

[3] Corporation for National Research Initiatives, “Proceedings of the Eighteenth Internet Engineering Task Force”, University of British Columbia, July 30-August 3, 1990.

Security Considerations

Security issues are not discussed in this memo.

Author's Address

John Moy
Proteon Inc.
2 Technology Drive
Westborough, MA 01581

Phone: (508) 898-2800
Email: jmoy@proteon.com