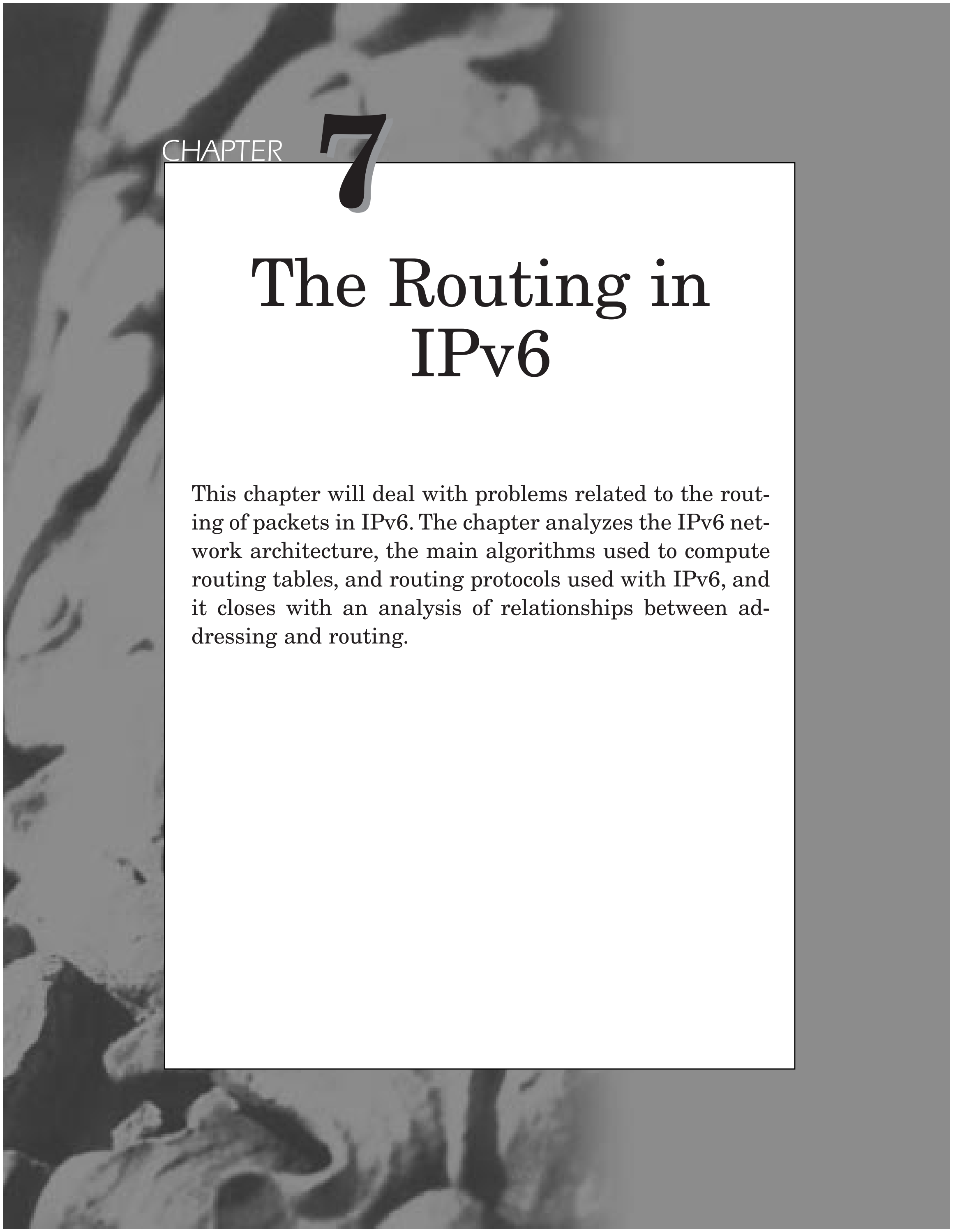


CHAPTER 7

The Routing in IPv6

This chapter will deal with problems related to the routing of packets in IPv6. The chapter analyzes the IPv6 network architecture, the main algorithms used to compute routing tables, and routing protocols used with IPv6, and it closes with an analysis of relationships between addressing and routing.

7.1 Terminology

The following terms are used in this chapter:

- *routing:* Determination of the path that an IP packet must follow to reach its destination.
- *path:* An ordered set of links that connect a source with a destination.
- *subnet:* A subset of nodes identified by addresses with a common prefix; these nodes are connected to the same physical link.
- *Autonomous System (*AS*):* A set of routing domains managed by a unique administrative authority.
- *routing domain:* A hierarchical partitioning of the network that contains a set of hosts and routers; routers share the same routing information, compute tables using the same IGP, and are managed by a common administrative authority.
- *exterior router:* A router that handles connections between different ASs.
- *border router:* A synonym for *exterior router*.
- *interior router:* A router that handles connections only within an AS.
- *Interior Gateway Protocol (*IGP*):* Generic term applied to each protocol used to advertise reachability and routing information within an AS. The term *gateway,* which is obsolete, is replaced by *router*.
- *Exterior Gateway Protocol (*EGP*):* Generic term applied to each protocol used to advertise reachability and routing information between different ASs. The term *gateway,* which is obsolete, is replaced by *router.*
- *static routing:* Technique in which routing tables are statistically determined during the network configuration.
- *dynamic routing:* Technique used to compute and update routing tables dynamically, taking into account the topology and the state of the network.
- *distributed routing:* Dynamic routing technique in which routing tables are computed through processes distributed on routers.
- *distance vector:* Distributed routing algorithm that computes routing tables based on an iterative exchange of routing tables between adjacent routers.

- *link state:* Distributed routing algorithm to compute routing tables in which a router communicates to all other routers in the network the state of the links directly connected to it through an LSP.
- *Link State Packet (*LSP*):* Packet generated by a link state protocol for the computation of routing tables; it contains the list of adjacent nodes.
- *hop:* The crossing of a link.
- *cost:* Metric associated with a link or to a path.
- *load splitting:* Balancing the load on several parallel paths.
- *static route:* One entry in a routing table, written manually by the network administrator.
- *End Routing Domain (ERD):* A routing domain in which routes are computed primarily to provide intra-domain routing services.
- *Transit Routing Domain (TRD):* A routing domain in which routes are computed primarily to carry transit—that is, inter-domain—traffic.
- *Routing Domain Confederation:* A set of routing domains seen as a unique entity and identified by a unique IPv6 prefix.
- *Internet Service Provider (*ISP*):* A public or a private organization that provides Internet services. Often simply called *provider*.
- *core router*: TRD's routers.
- *multihomed:* A network belonging to two or more routing domains.
- *Intranet:* A private network based on the Internet model.

7.2 Network Model

In Section 6.3.2, we saw that in IP packet routing a first level of hierarchy is represented by *subnets*. In fact, nodes, before transmitting packets, make a test to determine whether the destination is on-link or off-link. In the first case, the nodes send the packet directly to the final destination; in the second case, they use a router that, by consulting routing tables, determines which is the best path toward a given destination. If we take into account that IP addresses are associated with interfaces, not with nodes, the resulting model of the network is as illustrated in Figure 7-1.

Subnets are grouped into *Autonomous Systems (AS)*—that is, into sets of subnets controlled and administered by a unique authority[1]. Routers routing messages within the same ASs are called *interior routers,* and those routing messages between different ASs are called *exterior routers.*

An example of interconnection between two ASs (indicated by letters A and B) is shown in Figure 7-2.

Interior routers exchange routing information through an *Interior Gateway Protocol* (IGP), whereas exterior routers use an *Exterior Gateway Protocol* (EGP). The same IGP is normally used on all routers within an AS.

Figure 7-1
Model of an IP network

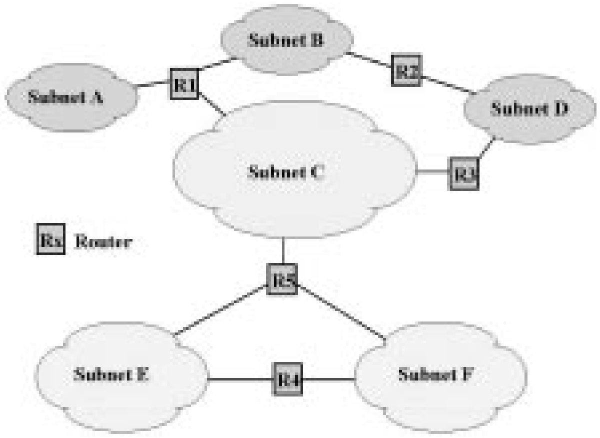
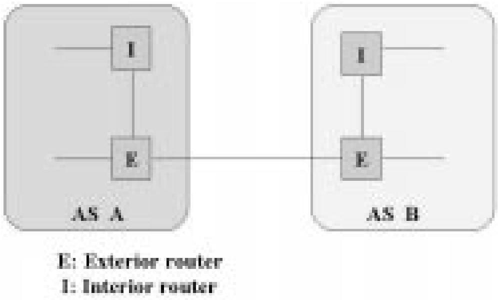


Figure 7-2
Example of interconnection of two ASs



7.3 Routing Algorithms

Routers, no matter whether they are interior or exterior, base their operation on routing tables (see Section 2.6). Routing tables can be written manually by the network administrator (*static routing*) or automatically computed through an appropriate algorithm (*dynamic routing*)[2]. These algorithms operate through an exchange of information between routers, relative to the topology and to the state of the network.

Today, the most-used dynamic routing algorithms are the *distributed routing* algorithms, which don't have a central point where tables are computed, but each router computes its tables by interacting with other routers. Among these types of algorithms, the two main families are *distance vector* algorithms and *link state* algorithms.

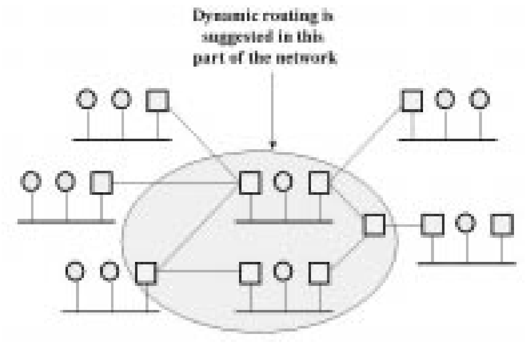
Both static routing and dynamic routing exist in different regions of the network for various reasons, as shown in Figure 7-3. In fact, even if having dynamic routing algorithms is necessary in order to take advantage of meshed networks, static routing can be more simple and may not present drawbacks in the most peripheral regions of the network with tree topology, regions in which only a path interconnects them to the rest of the network.

Note that, because IP subnets are associated with physical networks, each entry of the routing tables, independently from the type of routing used, specifies the reachability of a subnet or of a set of subnets (when the subnets belonging to the set can be aggregated).

7.3.1 Static Routing

Static routing requires the network administrator to write the routing tables manually. The administrator has total control of traffic flow on the

Figure 7-3
Dynamic and static routing



network, but manual intervention is required to reroute this flow in case of an error. This approach is frequently used in IP in the regions of the network that are not meshed; in these regions, no alternative routing paths are available, and tables can be simplified by using an entry that indicates a default path for all unknown destinations. A static entry within a routing table is called *static route.*

In large networks, the manual management of routing tables can be very complex.

An entry in the routing table can be manually created by a command of the type

```
route add 4800:600:0:C00:5/80 4800:600:0:C00:7:800:2B3C:
      4D5E
```

that specifies that all addresses beginning with the prefix on 80 bits **4800:600:0:C00:5** can be reached through the router connected on the same link (and therefore a neighbor), whose interface address is

```
4800:600:0:C00:7:800:2B3C :4D5E
```

The default entry can be manually created by a command of the type

```
route add default 4800:600:0:C00:9:800:2B3C:1234
```

that specifies that all addresses without a matching entry in the routing table can be reached through the router whose interface address is

```
4800:600:0:C00:9:800:2B3C:1234
```

Note that specifying default entries on hosts for the default router is not necessary. (This operation is necessary in IPv4, however.) In IPv6, routers present on links are automatically learned through the Neighbor Discovery process (see Section 6.3.1).

7.3.2 Metrics

To implement dynamic routing algorithms, introducing metrics is essential. Using metrics, we can measure a path's characteristics. This process is necessary for choosing, for example, the best among several alternative paths.

The only two metric parameters universally accepted are the following:

- The number of *hop*s—that is, the number of routers along a path
- The *cost*—that is, the sum of the costs of all links that compose the path

Both of these parameters state a negative metric because the cost of a line is assigned in a way inversely proportional to the speed of the line itself, and the hop count indicates the number of routers to be traversed and therefore a potential increase of the delay.

Taking into consideration the load of the network, metrics are more difficult to deploy because they easily lead to routing instability. The most modern techniques allow us to implement *load splitting* between parallel paths. This may also imply the activation of switched circuits like those provided by *Integrated Service Digital Network* (ISDN), either to manage an overloaded link or in the case of an error (backup function of a point-to-point WAN link).

7.3.3 Distance Vector

The distance vector algorithm is the first distributed routing algorithm to be implemented. Each router, besides the routing table, maintains a data structure, called a *distance vector,* for each line. The distance vector contains an entry for each destination, and each entry contains the destination address and the associated metrics. The distance vector contains information extracted from the routing table of the router connected on the other end of the line. Routing tables are computed, merging all the distance vectors associated with the router active lines. Each router periodically sends its routing table to other adjacent routers (neighbor routers) in the form of distance vectors.

When a router receives a distance vector from an adjacent router, it adds the received line metrics to those of the distance vector; it stores the results in its local data structure; it checks whether any change occurred in comparison with the distance vector previously stored, and if so, it recomputes routing tables by merging all distance vectors of active lines. The same recomputation operation occurs when a line goes from the ON state to the OFF state, or vice versa.

The merging is based on a criterion of lowest metrics: For each destination, the chosen path is the one with the lowest metrics among all possible paths.

If the routing table turns out to be changed in comparison with the previous one, the relevant distance vector is sent to adjacent routers. Some implementations of distance vector protocols periodically send distance vectors, too; for example, the RIP (see Section 7.4.1) sends the distance vector every 30 seconds.

The benefit of this class of algorithms is the extreme ease of implementation. Its drawbacks are as follows:

- The high complexity, exponential in the worst case and normally in the range between O(n^2) and O(n^3), where n is the number of entries. This makes the use of this algorithm not suitable for routing tables with more than 1000 entries.
- The slow convergence toward steady routing. The algorithm converges at a speed proportional to that of the slowest link and of the slowest router on the network.
- The difficulty to understand and to foresee its behavior on large networks because no node has the map of the network.

This algorithm is used to compute routing tables in RIP (see Section 7.4.1) and IGRP (see Section 7.4.4).

7.3.4 Path Vector

Path vector algorithms are similar to distance vector algorithms, but instead of metrics, they advertise the list of ASs to be traversed to reach each destination. Using the AS list is a simple way to discover possible loops on the network and to implement routing policies that prefer certain routings, in function of ASs to be traversed.

Path vector algorithms are used in EGP protocols (see Section 7.4.3).

7.3.5 Link State

Link state algorithms have been recently adopted. They are based on the idea that each router, interacting with other ones, builds a complete map of the network on which it computes optimal routings by using Dijkstra's algorithm[3] or *Shortest Path First* (SPF).

Routers interact by exchanging *Link State Packets* (LSPs). Through LSPs, each router communicates to other routers which subnets it is directly connected to. Each router contains a database called an *LSP database* in which it stores the most recent LSP generated by each other router. The LSP database is a representation of the graph of the network given as a matrix of adjacent neighbors (see **2** and **3**). Note that the LSP database is, by definition, exactly identical on all routers of the network.

Moreover, the previous approach presents a duality: Distance vector routers send information concerning all subnets only to neighbor routers; link state routers send information concerning only subnets to which they are directly connected to all routers on the network.

The LSP database, representing the map of the network with associated metrics, provides necessary and sufficient information for a router to compute its routing table.

Again, note the difference with the distance vector: In that case, routers directly cooperate to compute routing tables; whereas here routers cooperate to maintain the updated map of the network, and then each router autonomously computes its own routing table.

The computation of the link state algorithm is equal to O(L • log(N)), where L is the number of links and N is the number of nodes. Because metrics are small integers, sophisticated data structures, which make the complexity algorithm tend to O(N), can be implemented.

The link state algorithm can administer very large networks (10,000 entries in the routing table). It quickly converges; it rarely generates loops; and in any case, it can easily detect and interrupt them. Also, it can be easily understood and predicted because each node contains the whole map of the network.

Link state algorithms have been used in the OSI IS-IS (Intermediate System to Intermediate System) ISO 10589[4] standard, in the OSPF protocol (see Section 7.4.2), and in the Dual IS-IS (see Section 7.4.4) protocol.

7.3.6 Redistribution

Though the definition of AS clearly indicates that, within an AS, all interior routers must use the same IGP, in practice this rule is frequently violated. Many ASs use different IGPs at the same time because the software available on routers allows them to do so. Therefore, there is the need to allow an IGP #1 to redistribute reachability information learned from an IGP #2, and vice versa. This operation implies an accurate correspondence of metrics used by the two IGPs. This can be quite easily implemented in parts of the network with a star topology (for example, redistributing the reachability information learned from static routes is fairly common), but it presents considerable problems in the presence of meshes partly managed by IGP #1 and partly by IGP #2. This configuration is highly discouraged because it can easily create loops not easy to detect.

7.3.7 Multi-Protocol Routing

Real networks rarely are mono-protocol—that is, networks using only one layer 3 (Network) protocol. Usually, LANs simultaneously transport many protocols by marking frames with different Protocol Types (see Section 2.9) —for example, 0800 hexadecimal for IPv4 and 86DD hexadecimal for IPv6. Network administrators also sometimes need to transport many protocols at the same time on the geographic part of the network; for this purpose, *multi-protocol routers* are used. These routers must compute routing tables for many protocols, and this process can be performed through the use of two different approaches: *integrated* or *ships in the night*.

In the integrated approach, only one protocol is used to compute all routing tables. This result is achieved by enabling the protocol to transport the reachability information of several protocols at the same time.

In the "ships in the night" approach, each routing table is computed by a specific protocol, and the different protocols travel in parallel, ignoring each other like ships that pass in the night.

The integrated approach is undoubtedly very elegant, but its implementation is very complex and less flexible. The author of this book takes the liberty, after many years spent working on networks, to suggest that all readers use the "ships in the night" approach.

7.4 Routing in IPv6

The three main protocols for the computation of routing tables that will be used with IPv6 are RIPv6 (see Section 7.4.1), OSPFv6 (see Section 7.4.2), IDRPv2 (see Section 7.4.3), and probably EIGRP and Dual IS-IS (see Section 7.4.4).

None of the algorithms previously used for IPv4 can be used without modifications because they are unable to transport IPv6 addresses on 128 bits.

7.4.1 RIPv6

The *Routing Information Protocol* (RIP) is an IGP originally designed by Xerox for its XNS network. It was introduced in the TCP/IP architecture in 1982 at the University of California at Berkeley with the name *routed* (route daemon), defined in RFC 1058 in 1988[5] and updated by RFC 1388 in 1993[6]. RIP is widely adopted, mainly in implementations of personal computer networks, and many other routing protocols are based on it, such as AppleTalk, Novell, 3Com, Banyan, and so on.

RIP is a distance vector protocol in which each router sends its distance vector to adjacent routers, every 30 seconds (see Section 7.3.3). Routing tables store only the best next hop toward each destination. The main limit of RIP is that it allows a maximum of 15 hops; each destination more distant than 15 hops is considered unreachable.

Moreover, RIP ignores lines' speeds, not allowing the definition of costs or other metrics, but it bases the routing only on the minimization of the number of hops. In case of modifications of the network topology, RIP is slow to converge. For these reasons, RIP can be used only on small networks.

RIPv6[7] is the version of RIP that can be used with IPv6. This update of RIP allows it to bear the new 128-bit addresses and relevant prefix lengths without adding any new features and without eliminating the limits cited previously. The reason for this choice is based on the need to maintain RIPv6 simplicity so that it can also be implemented on very simple devices on which the implementation of OSPFv6 would be problematic.

RIPv6 has only two types of messages—Request and Response—that are transported in the UDP (User Datagram Protocol)[8]. In RIPv6, a limited number of destinations per each packet is allowed so that the resulting IPv6 packet doesn't exceed the link-MTU.

7.4.2 OSPFv6

The *Open Shortest Path First* (OSPF) is an IGP purposely developed for IP. In 1988, an IETF working group was appointed to implement a *link state* protocol (see Section 7.3.5) for IP. OSPF was defined by RFC 1247 in 1991[9] and redefined by RFC 1583 in 1994[10].

OSPF is based on the concept of hierarchy. The root of the hierarchy is the AS that can be subdivided into areas, each one containing a group of interconnected networks. The routing within an area is called *intra-area;* the routing between different areas is called *inter-area*. Each AS has a *backbone* area that can also be not contiguous; in this case, configuring *virtual links* is necessary to guarantee its cohesion. All other areas are connected to the backbone area.

OSPF routers are classified into four categories, not mutually exclusive:

- *Internal router:* A router connecting subnets all belonging to the same area. These routers use only one instance of the OSPF algorithm. Routers having interfaces only on the backbone belong to this category.

- *Area border router:* A router connecting the backbone area to one or more areas. These routers use many instances of the OSPF algorithm: one instance for each directly connected area and one instance for the backbone. Area border routers collect the reachability information from areas to which they are connected and redistribute it on the backbone. The backbone redistributes this information to other areas.
- *Backbone router:* A router with an interface on the backbone. This category includes all routers connected to more than one area (area border router). Backbone routers with all interfaces on the backbone are considered internal routers.
- *AS boundary router:* A router exchanging router information with routers belonging to other ASs. This classification is orthogonal to the previous ones; an AS boundary router can be an internal or an area border router.

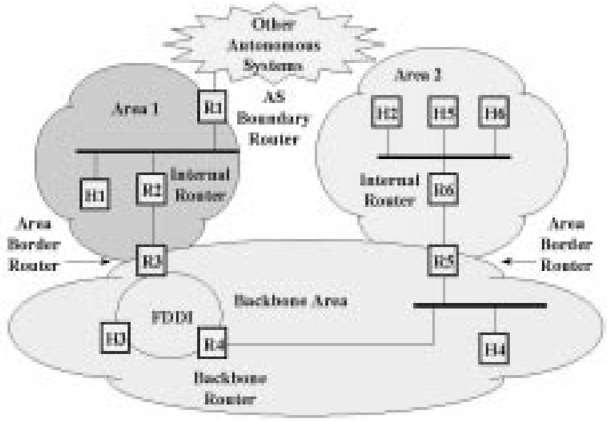
Figure 7-4 shows an example of AS subdivided into three OSPF areas and connected to another AS.

OSPFv6[11] is the version of OSPF that can be used with IPv6; it is also the IGP protocol suggested for IPv6. As the standard implemented by all router manufacturers, it is suited for large networks.

OSPFv6, which is an update of OSPF, allows transportation of the new 128-bit addresses and the associated prefix lengths. In OSPFv6, areas are identified by 128-bit addresses.

No new functions have been added because OSPF represents the "state of the art" of IGP protocols. OSPF for IPv4 and OSPF for IPv6 operate in parallel, following the "ships in the night" approach (see Section 7.3.7).

Figure 7-4
Example of use of OSPF



OSPFv6 is layered directly on IPv6, and the OSPFv6 header is identified by the value 89 in the Next Header field of the preceding header (see Table 3-2 and Section 3.1.5).

7.4.3 IDRPv2

The *Inter-Domain Routing Protocol* (IDRP)[12] is an EGP protocol to be used with IPv6. The IDRP is a path vector protocol (see Section 7.3.4), designed to be used in the OSI architecture for the CLNP ISO 8473 protocol and derived from the BGP-4 (Border Gateway Protocol version 4, RFC 1711[13]) used as EGP on the Internet. The IDRP version suitable for operating with IPv6 is version 2 (IDRPv2[14]).

IDRPv2 uses the term *routing domain* instead of the term *autonomous system*. A routing domain is identified by an IPv6 prefix (128-bit address); this identification simplifies the IANA's work (see Section 4.5) because explicitly assigning the AS's identifiers, which in IPv4 are on 16 bits, is no longer necessary.

Routing domains can be grouped into a *Routing Domain Confederation*. Confederate routing domains are seen as unique entities, and they are identified by IPv6 prefixes, too. Confederate routing domains can be confederated by introducing an arbitrary number of hierarchy levels.

IDRP subdivides routing domains into two types:

- *End Routing Domain (ERD)*: A routing domain in which routes are computed primarily to provide intra-domain routing services.
- *Transit Routing Domain (TRD)*: A routing domain in which routes are computed primarily to carry transit (that is, inter-domain) traffic.

The IDRPv2 has been chosen to replace the BGP because of the following reasons:

- Although defined in the OSI architecture, it doesn't present any specific dependence on the OSI architecture itself.
- It has been conceived from the beginning for the multi-protocol routing, allowing several types of addresses.
- It includes all BGP-4 functions, and it is based on the same path vector philosophy (it advertises the routing domain or routing domain confederation sequence to be traversed to reach a given destination).

Each router computes its preferred routing toward a given destination and transmits it to IDRP-adjacent routers through a path vector. The policy to make this computation is configurable on each IDRP router.

IDRP is layered on IPv6, and the IDRP header is identified by the value 45 in the Next Header field of the preceding header (see Table 3-2 and Section 3.2.5).

7.4.4 Other Routing Protocols

Other protocols to compute routing tables have been used in IPv4, and some of them will probably be used in IPv6 as well. Among them, the most important are IGRP and Dual IS-IS.

7.4.4.1 IGRP The *Interior Gateway Routing Protocol* (IGRP)[15] is an IGP developed by Cisco Systems, Inc., in the mid '80s to overcome RIP's limits. It is a distance vector protocol, but it features a very sophisticated metric. IGRP chooses the best path by combining metric vectors containing delay, bandwidth, reliability, maximum length of the packet, and load. Moreover, IGRP allows *multi-path routing*—that is, the subdivision of traffic among parallel lines. The traffic is subdivided on the basis of metrics associated with lines.

Extended IGRP (EIGRP), which is an improved version of IGRP, allows multi-protocol routing and the management of the variable subnetting and of the *Classless Inter-Domain Routing* (CIDR)[16]. Cisco will probably introduce support for IPv6 in future versions of EIGRP.

7.4.4.2 DUAL IS-IS The *integrated IS-IS*, also called *dual IS-IS*[17], is a version of the IS-IS (ISO 10589) protocol[4] that also can compute routing information for protocols different from OSI CLNP (ISO 8473).

RFC 1195[17] standardizes operation of the dual IS-IS in a mixed OSI CLNP and IPv4 environment. The IETF will probably introduce support for IPv6 in future versions of the dual IS-IS.

7.5 Relationships between Addressing and Routing

So far, we have analyzed routing problems (in this chapter) and addressing problems (in Chapter 4) separately. Now we can further analyze existing relationships between addressing and routing. Topics reported in the following subsections are discussed in more depth in RFC 1887[18].

7.5.1 Internet Structure

The Internet is organized into routing domains that exchange information on the reachability of networks on which they are composed. These routing domains do not have equal importance, and we have already seen that IDRP makes a distinction between Transit Routing Domain (TRD) and End Routing Domain (ERD). An example of interconnection between ERDs and TRDs is illustrated in Figure 7-5.

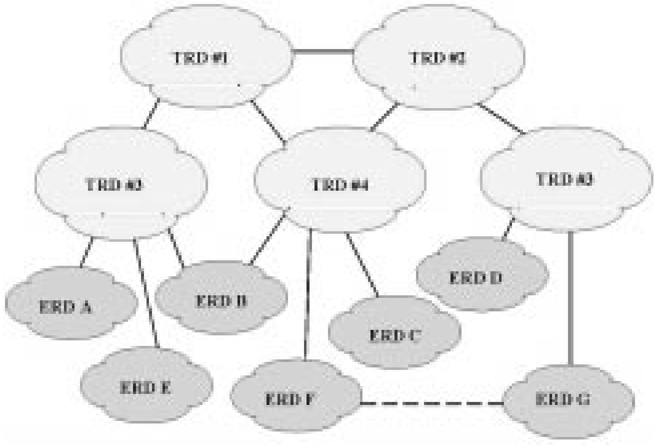
ERDs are associated with the network's end users—that is, to organizations connected to the Internet that usually have connections with only one TRD. Sometimes an ERD can have connections with many TRDs; in this case, the ERD is called *multihomed* (for example, in Figure 7-5, the ERD B). It, however, maintains its ERD nature—that is, it doesn't operate as a transit domain—and it therefore remains a leaf (see Section 7.5.5).

Another possibility is that two ERDs have a private link (see Section 7.5.7) because they have to exchange large volumes of traffic, without passing through the Internet. This is the case of ERDs F and G in Figure 7-5.

TRDs are usually associated with *Internet Service Providers* (ISPs); in the following text, we will simply call them *providers*. These providers can be subdivided into the following categories:

- *Direct Service Providers*: These providers connect end users and connect themselves to international backbones. Examples of Direct Service Providers are America Online and NSFnet regional.

Figure 7-5 Interconnection between ERDs and TRDs



- *Indirect Service Providers*: These providers administer large international backbones, the highest level in the hierarchy. They connect only Direct Service Providers and big users.

7.5.2 IPv4 Problems

In IPv4, no relationship exists between addresses and topology. In fact, addresses are directly assigned to end users and, even if an effort is made to assign addresses by nations or continents, this use poses no particular benefits for routing. The Internet, by its nature, doesn't respect nations' political borders. For example, Italian organizations can connect to Italian providers and these to European providers, but they can also connect to American providers. As a result, Italian networks are announced partly in Europe and partly in the United States. This situation is likely to become more and more complicated with the coming of a telecommunications free market.

In this situation, ERD routers don't present any particular drawbacks; in fact, it is sufficient that they maintain in their routing table one entry for each network within the ERD and one default network for all other networks. The default entry points to the TRD of the provider to which the ERD is connected.

The case of TRD routers (also called *core routers*) is more complex. In fact, they must maintain in their routing tables one entry for each network connected to the Internet (this is undoubtedly true for Indirect Service Provider routers). Therefore, the routing tables tend to explode with the dizzying growth of the Internet.

To limit the growth of routing tables, the *Classless Inter-Domain Routing* (CIDR)[16] was introduced with BGP-4. The CIDR allows grouping of announcements of many networks whose addresses are contiguous in only one entry (see Section 1.2.1). Nevertheless, the CIDR cannot bring important benefits due to the assignment philosophy of IPv4 addresses. In fact, it is not sure that contiguous addresses are assigned to users connected to the same TRD and that the TRD can therefore group them.

7.5.3 The IPv6 Solution

To solve the problems cited in the preceding subsection, IPv6 migrates from a scheme based on the assignment of addresses to end users (like that of IPv4) to a provider-based scheme (see Section 4.6.2). In this new scheme, each Direct Service Provider is assigned a set of addresses that it divides into smaller sets to be assigned to its users. Because the IPv6 address is much longer than the IPv4 address, it can easily contain this new hierarchy level. Sets of addresses assigned to the users can be grouped by definition by the provider because they are the result of a partition.

For ERDs' routers, the situation remains unchanged. They continue to have one entry for each network within the ERD, one default entry toward the TRD, and they announce their set of addresses to the TRD with only one entry.

For Indirect Service Providers' TRD routers, the situation is completely different. In fact, now each Direct Service Provider announces all its networks with only one entry; therefore, the size of routing tables is proportional to the number of providers, not to the number of networks.

For the Direct Service Provider's TRD routers, the situation can change significantly if many connections are made with other providers (either Direct or Indirect). In fact, all networks associated with a provider are announced with a single entry in routing tables in this case.

Other possible aggregation schemes have been proposed. For example, providers can be aggregated on a continental basis, or Indirect Service Providers can be assigned address sets to be subdivided by assigning the addresses to Direct Service Providers, and the Direct Service Providers, in their turn, can assign the addresses to end users. The usefulness of these schemes is questionable.

What is not questionable, however, is that the providers' assignment of addresses to end users brings about a significant containment of routing tables (that we can estimate in two orders of magnitude). IPv6 will therefore follow this approach.

7.5.4 Drawbacks for Users

The main drawback for users happens when they decide to change providers—that is, to buy Internet services from another ISP. In fact, users have to renumber their networks. As we already explained in Section 6.7.2, this operation is simplified by IPv6 Neighbor Discovery mechanisms, but it still can cause some inefficiency.

Nevertheless, a user can operate with addresses from provider A while still being connected to provider B. In this case, provider B must explicitly announce addresses assigned to the user by provider A. All Internet routers should have one additional entry to indicate that the user, though having addresses from provider A, can still be reached through provider

B. This situation can occur for a limited period of time during a transition to allow the user to renumber networks without service interruptions; however, this situation cannot continue indefinitely because it will rapidly recreate the unacceptable growth of routing tables, as in the previously analyzed IPv4 case.

7.5.5 Multihomed Routing Domains

The previously discussed theories apply to ERDs that are connected to only one TRD. However, what happens when we want an ERD to be multihomed—that is, to be connected to many TRDs—without becoming a TRD, but remaining a leaf routing domain?

Examples of multihomed ERDs are routing domains in a big organization covering the whole nation that decides to connect to the Internet in many points through different providers, or even that of an international organization that decides to connect its network to the Internet in the nations where its main subsidiaries are located.

There are several reasons to have an ERD multihomed. The two main reasons are the larger availability of bandwidth, and the possibility of having alternative paths in case of errors and, therefore, a more reliable network.

In IPv6, an entire domain can be multihomed, but also a single subnet or a single host can be. A multihomed host can, in turn, be multihomed because it has many IPv6 addresses assigned to different interfaces (this case is common in reliable hosts) or because it has many addresses associated with the same interface (for example, a LAN with many prefixes associated with different providers). This topic is still the subject of debate in the Internet community, and at the time this chapter was written, only an Internet Draft[18] on this topic is available.

RFC 1887[19] provides four possible solutions for connecting an ERD to many TRDs. C. Huitema[20], who highlights the existing implications between multihoming and upper layer protocols, proposes a fifth solution.

7.5.5.1 SOLUTION #1 A multihomed organization obtains a prefix independently of the providers to which it is connected. This solution causes an additional entry in all core routers, and it is acceptable only for a few very large organizations. This solution does not scale to all organizations that will connect to the Internet in the future and that want to be multihomed because many hundreds of thousands of organizations could want this capability.

7.5.5.2 SOLUTION #2 The organization is assigned as many different prefixes as there are providers it will be connected to. In each part of the network, the organization will use a prefix chosen on the basis of the distance of that part of the network to a particular provider. For example, let's suppose that an organization has a network covering Italy, France, and Spain, and that it wants to be connected to the Internet in these three nations. For the Italian part of the network, it will use addresses derived from the set it has been assigned by an Italian provider; for the French part, addresses from a French provider; and for the Spanish part, addresses from a Spanish provider.

For this solution, core routers don't need to maintain any additional information for the organization because it will be reached as three separate organizations that are part of three different providers. Routers within the organization can be efficiently configured by using private links (see Section 7.5.7), without upgrading the ERD to a TRD.

The main disadvantage of this solution is the lack of backup mechanisms in case one of the three connections with the providers fails. The part of the network configured with addresses of that provider simply becomes unreachable because those addresses are not announced by the other two providers. Announcing them would be possible, but doing so would be much more expensive than in the preceding case because core routers should maintain three entries for the organization, one for each prefix used on the network. Moreover, if a provider is changed, all addresses associated with that provider should be changed, too.

Also, note that, with the previous approach, packets enter the organization via the point that is closest to the source node (which tends to maximize the load on the internal network); with this second solution, packets enter the organization via the point that is closest to the destination node (which tends to maximize the load on the Internet).

7.5.5.3 SOLUTION #3 Now suppose that a second organization uses provider A's prefix as the prefix for its networks because provider A is meant to be used as the default to the Internet. Other TRDs to which this organization is connected will advertise A's prefix only in restricted and controlled areas. For example, let's suppose that this organization also belongs to the Italian Public Administration network, administered by provider B. Provider B will advertise, within the public administration network, that this organization can be reached by a set of addresses from provider A. This capability entails that routers of the TRD of B have an explicit entry in routing tables for the organization, but it doesn't introduce any additional entry on core routers.

7.5.5.4 SOLUTION #4 The fourth solution can be used when two or more providers have many customers in common. This solution is hypothetical and will become fairly common when the use of IPv6 on public networks is more widespread. In this case, the two providers request a third set of addresses (in addition to the two they already have) to be assigned to customers they have in common and interconnect their TRDs. There is no penalty at the core router level because all users in common between the two providers are advertised with only one entry in the routing tables.

7.5.5.5 SOLUTION #5 For the fifth solution, each station is assigned as many addresses as there are providers. This situation is illustrated in Figure 7-6, where station X has two addresses: A::X derived from provider A and B::X derived from provider B.

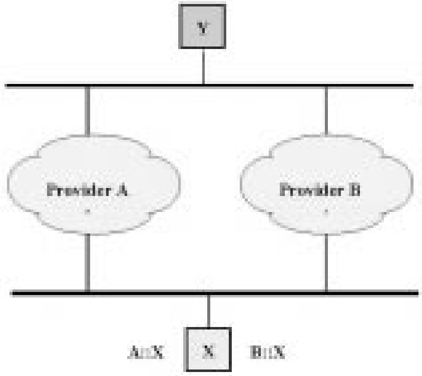
This solution is not perfect. Suppose that X establishes a Telnet session with Y by using its address A::X. If, during the session, provider A becomes overloaded or it cannot reach X through A, the session cannot be rerouted using provider B. This operation will entail the use of address B::X in the IPv6 packet instead of the A::X address, but this use is not possible. In fact, the Telnet application lays on the *Transmission Control Protocol* (TCP), which also uses the IPv6 address as the connection identifier; according to RFC 793[21], this address cannot be modified during the connection itself.

A less pragmatic solution is to close the Telnet session and to open another one, this time using the address B::X.

A second solution, currently under discussion, is to modify the TCP protocol allowing IPv6 addresses to change during the connection.

A third possibility is that Y inserts a Routing Header (see Section 3.2.5) to force the routing to pass through B::X. In this way, the destination ad-

Figure 7-6
Example of multihoming



dress in the IPv6 packet remains A::X, but the packet is delivered to B::X, which routes it within itself to A::X—that is, to itself. The only drawback to this solution is represented by the routing header overhead (24 octets in the case of a single intermediate address).

7.5.6 Tunnel

In the solutions described in the preceding subsections are frequent references to the possibility that a multihomed host decides which address to use among many source addresses. Frequently, this is not possible because hosts don't have enough information to decide correctly or because network administrators don't want this situation to occur.

Network administrators typically want to base their decisions about which provider to use on the borders of the network—that is, on border routers. A possibility is represented by the creation of tunnels, which means transporting IP packets inside other IP packets.

This possibility, at the time this chapter was written, is described by an Internet Draft[22], and it corresponds to creating "virtual links" between two IPv6 nodes that see the tunnel as a communication channel at data link level—that is, as a link. The two nodes have two specific tasks: A node encapsulates the original packet and transmits it on the tunnel; and the other one receives the packet from the tunnel, eliminates the encapsulation, and transmits it to its destination.

Tunnels are unidirectional mechanisms; a bidirectional tunnel can be implemented by using two unidirectional tunnels.

Tunnels have at least three important applications:

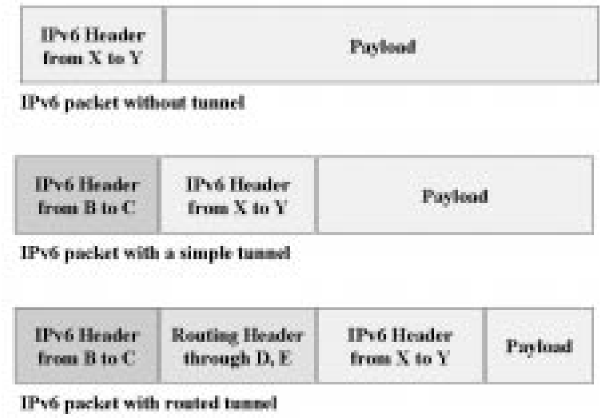
- Bypassing providers' routing policies
- Interconnecting Intranets through the Internet network (see Section 7.7)
- Implementing 6-Bone—that is, a first core of the Internet using IPv6

Tunnels can be simple or routed (see Figure 7-7).

In the case of simple tunnels, an IP packet is transported inside an IP packet with an overhead equal to the size of the IP header (in the case of IPv6, 40 octets). In the example shown in Figure 7-7, the simple tunnel allows the packet originating in the routing domain B to reach Y by traversing routing domain C.

In the case of routed tunnels, a Routing Header is inserted to specify other routing domains that must be traversed on the path toward the des-

Figure 7-7
Examples of tunnels



tination. In the example shown in Figure 7-7, the routed tunnel allows the packet originating in routing domain B to reach Y by traversing the routing domains D, E, and C.

7.5.7 Private Links

Suppose that two organizations X and Y have two ERDs and decide to improve their interconnection performance by acquiring a point-to-point link between the two ERDs. This approach doesn't raise any particular routing concerns on the Internet; it is a local agreement that is ignored by core routers. To create this link, adding one entry relevant to Y in routing tables of the ERD of X is sufficient, and vice versa. If Y connects other ERDs of other organizations with which it has an intense exchange of information to its ERD, accessing these organizations from X through a private link is also possible, by adding the necessary entries in routing tables.

7.6 Multicast Routing

The term *multicast routing* refers to routing of packets whose destination address is a multicast address—that is, the address of a group of stations. In Section 4.8, we saw that some of these multicast addresses are associated with predefined groups and have meaning only with regard to the node or to the link; whereas other multicast groups can have members in various parts of the Internet network, and therefore packets addressed to these multicast groups must be routed by routers.

The problem of multicast routing in IPv6 is similar to that in IPv4, with the following main differences:

- In IPv4, members of groups are administered with a specific protocol called *Internet Group Membership Protocol* (IGMP)[23], which in IPv6 became an integrated part of ICMPv6 (see Section 5.6.3) while maintaining the same functions.
- In IPv4, multicast packets are routed by two alternative protocols: the *Distance Vector Multicast Routing Protocol* (DVMRP) standardized in RFC 1075[24], or the *Multicast OSPF* (MOSPF) consisting of extensions to the protocol OSPF standardized in RFC 1584[25] to deal with multicast packets. In IPv6, the MOSPF extension became an integrated part of OSPFv6[11].

In summary, to route multicast packets, we must create a distribution tree (*multicast tree*) to reach all members of the group. The tree is clearly dynamic because new members can join the group, and existing members can leave it at any moment. The addition of members typically induces growth of the tree; whereas members leaving the group potentially "prunes" the tree.

Therefore, the multicast routing problem turns out to be an integrated part of IPv6 and, in particular, of ICMPv6 and OSPFv6 protocols.

7.7 Intranet

Many organizations, while deciding to implement networks based on the IP protocol, don't want to be interconnected to the Internet or want to have extremely controlled access to the Internet. These organizations implement Intranets, which are private networks based on the Internet model (see RFC 1918[26], even if relevant to IPv4). The configuration of Intranet networks is hugely simplified in IPv6, from the addressing point of view, because assigning site local addresses to the private part of a network is sufficient (see Section 4.6.5). The public part has, on the other hand, provider-based global addresses.

Figure 7-8 shows an example of Internet/Intranet configuration. To communicate between the public and the private part, a consolidated technical solution is used; it provides the installation of application gateways (for example, for the electronic mail) and proxy servers (for example, for WWW, FTP, and Telnet) on public hosts.

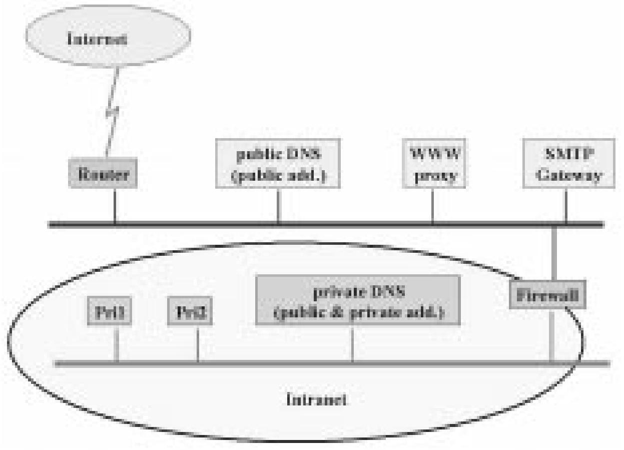


Figure 7-8 Connection scheme between an Intranet and the Internet

Between public and private networks, either a router, with appropriate access filters, or a real firewall is inserted to avoid propagating information about the private network on the Internet. Moreover, if a company implements many Intranets—for example, one for each subsidiary—it can interconnect these Intranets by implementing "tunnels" on the Internet between the firewalls of the different subsidiaries. The term *tunnel* (see Section 7.5.6) indicates an encapsulation of an IP packet in another IP packet: The IP packet of the Intranet is encapsulated in an IP packet of the Internet.

A public DNS server, connected to worldwide DNS systems, must be available; it is used to define the addresses of public hosts. A second private DNS server contains both public hosts' addresses and private hosts' addresses, and uses the public DNS as the sender toward the Internet. All hosts (either public or private) use the private DNS.

Another practical method to increase the security is to adopt a separate cabling for the public part (Internet) and the private part (Intranet) of the network. The term *separate cabling* here means a physical organization of the cabling in which, even if a hacker succeeds in loading a program for the capture of the network packets on a host that can be reached on the Internet, this program cannot see the Intranet packets because they travel on other cables.

REFERENCES

[1] G. Bennett, *Designing TCP/IP Internetworks*, Van Nostrand Reinhold, 1995.

[2] S. Gai, P.L. Montessoro, P. Nicoletti, *Reti Locali: dal Cablaggio all'Internetworking*, SSGRR (Scuola Superiore G. Reiss Romoli), 1995.

[3] J. V. Aho, J. E. Hopcroft, J. D. Ullman, *Data Structures and Algorithms*, Addison-Wesley, 1983.

[4] ISO 10589, *Intermediate system to Intermediate system Intra-Domain routing information exchange protocol for use in conjunction with the Protocol for providing the connectionless-mode network service*.

[5] C.L. Hedrick, *RFC 1058: Routing Information Protocol*, June 1988.

[6] G. Malkin, *RFC 1388: RIP Version 2 Carrying Additional Information*, January 1993.

[7] G. Malkin, R. Minnear, *RIPng for IPv6*, August 1996.

[8] J. Postel, *RFC 768: User Datagram Protocol*, August 1980.

[9] J. Moy, *RFC 1247: OSPF Version 2*, July 1991.

[10] J. Moy, *RFC 1583: OSPF Version 2*, March 1994.

[11] R. Coltun, D. Ferguson, J. Moy, *OSPF for IPv6*, June 1996.

[12] ISO 10747, *Protocol for Exchange of Inter-Domain Routing Information among Intermediate Systems to Support Forwarding of ISO 8473 PDUs*.

[13] Y. Rekhter, T. Li, *RFC 1771: A Border Gateway Protocol 4 (BGP-4),* March 1995.

[14] Yakov Rekhter, Paul Traina, *Inter-Domain Routing Protocol, Version 2*, June 1996.

[15] Cisco Systems, *Router Products Configuration and Reference*, Cisco Systems DOC-R9.1, Menlo Park, CA, September 1992.

[16] V. Fuller, T. Li, J. Yu, K. Varadhan, *RFC 1519: Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*, September 1993.

[17] R.W. Callon, *RFC 1195: Use of OSI IS-IS for routing in TCP/IP and dual environments*, December 1990.

[18] M. Shand, M. Thomas, *Multi-homed Host Support in IPv6*, June 1996.

[19] Y. Rekhter, T. Li, *RFC 1887; An Architecture for IPv6 Unicast Address Allocation*, December 1995.

[20] C. Huitema, *IPv6: the new Internet Protocol*, Prentice-Hall, 1996.

[21] J. Postel, *RFC 793: Transmission Control Protocol*, September 1981.

[22] A. Conta, S. Deering, *Generic Packet Tunneling in IPv6 Specification*, June 1996.

[23] S.E. Deering, *RFC 1112: Host extensions for IP multicasting*, August 1989.

[24] D. Waitzman, C. Partridge, S.E. Deering, *RFC 1075: Distance Vector Multicast Routing Protocol*, November 1988.

[25] J. Moy, *RFC 1584: Multicast Extensions to OSPF*, March 1994.

[26] Y. Rekhter, B. Moskowitz, D. Karrenberg, G.J. de Groot, E. Lear*, RFC 1918: Address Allocation for Private Internets*, February 1996.