Use Domain Name System and IP Version 6

What You Will Learn

The introduction of IP Version 6 (IPv6) into an enterprise environment requires some changes both in the provisioned Domain Name System (DNS) data and in the way the data is provisioned. This document explains the changes needed.

To use IPv6 with DNS, you need to perform a series of tasks, explained here. The document also explains some DNS-specific terms and processes, but the reader of this document is expected to already have a working DNS set up for IPv4 and a basic knowledge of DNS. For more information about DNS, see the For More Information section at the end of this document.

Basic Steps

You need to perform the following nine steps to use IPv6 with DNS:

1. Add AAAA records in your DNS server for the hostnames of the devices that can be reached through the IPv6 protocol.
2. Add pointer (PTR) records in your DNS server for the IP addresses of the devices that can be reached through the IPv6 protocol.
3. Enable IPv6 access to the authoritative DNS servers. Be sure that TCP/53 and UDP/53 can be accessed through IPv6.
4. Enable IPv6 connectivity to the external full-service resolvers that send DNS queries to authoritative servers in the world.
5. Make sure that the full-service resolver is configured with both IPv4 and IPv6 glue for the root servers in the world.
6. Enable IPv6 on the recursive resolver so that it responds to DNS requests over IPv6 as well as IPv4.
7. Enable IPv6 on the node that sends queries so that it can send DNS requests to the recursive resolver.
8. Configure the stub resolver on the node that sends queries so that it uses IPv6 to send DNS queries, either statically or using Dynamic Host Configuration Protocol Version 6 (DHCPv6).
9. Review policies for flows and make sure that both TCP/53 and UDP/53 can be accessed over IPv4 and IPv6.

These steps are discussed in more detail later in this document.

DNS Basics and Terminology Used in This Document

The Domain Name System, or DNS, is both the namespace and database that defines certain operations such as lookups, and the protocol used between a client and a server to implement this distributed lookup mechanism. In the protocol, the client sends a request (often called a query) and gets back a response (or answer). The request is sent either to a preconfigured IP address of a server or to a server for which the IP address has been discovered through earlier requests.

Normally, DNS involves three hosts: the client that runs an application that needs the address for given a hostname, the intermediary server that responds to this query and acts as a proxy, and the authoritative server that holds the authoritative data.

Queries can be sent either with a request from the client that the server provide recursion, or without such a request. If recursion is requested, the server can choose to deny this request. A client that runs an application the needs responses normally runs a resolver, which always requests recursion. This so-called stub resolver is configured (often through DHCP) with the IP address of the intermediary server that acknowledges this request. This intermediary is configured with the IP addresses of the root servers in the world and implements recursion by repeatedly sending the queries first to the root servers and then to whomever the root server refers the query. The intermediary (also called the full-service resolver) sends the queries without requesting recursion.

An intermediary server can use a forwarder, in turn using another intermediary server for all its queries. This process can be performed in many steps.

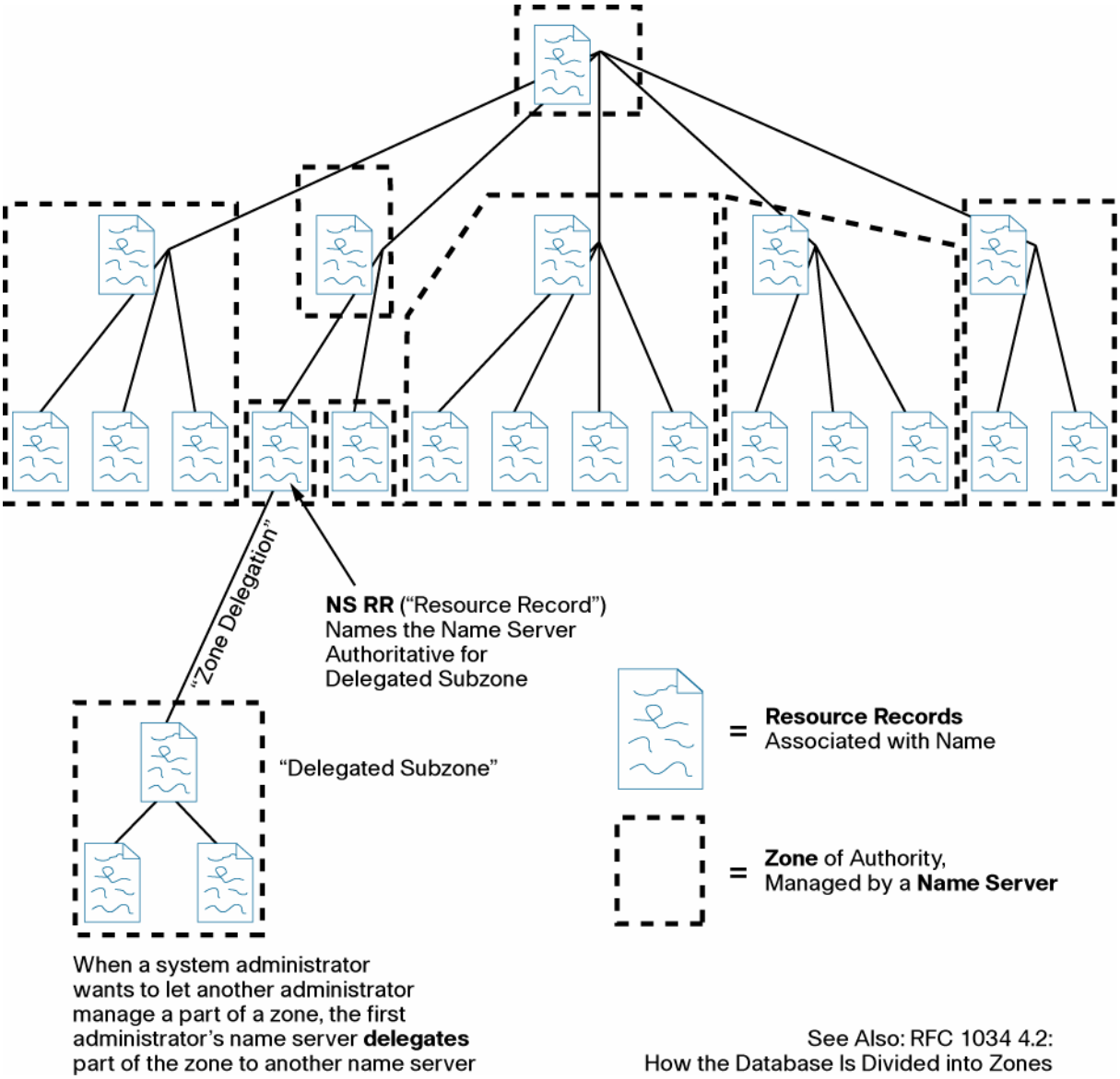
When a client sends a request to a server and the client does not request recursion, instead of responding with a response the server may send back one or more same server (NS) records. This record includes the hostname of a name server that, as far as the responding server knows, has the answer to the query. The client then resends the query to the host to which the name server record refers, and this may in turn result in a response with name server records. This repeated querying is called recursion.

When a name server record (NS) is sent back to, and received by the client, the client must look up the IP address to which the hostname refers. This is because the next query is to be sent to the host which the NS record refers to. If the hostname is in the delegated zone, a difficulty occurs: only the server the IP address refers to knows the answer to the query regarding hostname to IP address mapping. In this special case, the server that returned the name server records, in addition to the name server records that refer to the hostname, also includes the IP addresses of the hostnames that are in the delegated zone. These mappings from hostnames to IP addresses are called glue records, or just glue.

Using DNS and IPv6

In DNS, a namespace consists of zones, which are delegated to a number of authoritative name servers, as shown in Figure 1.

Figure 1. The Namespace in DNS, with Zones and Delegation Points



This namespace has names for everything that has an IP address; these names are in the form of mappings from names to IP addresses. In addition, there is data that maps IP addresses to names (not always the same name that maps from the name to the IP address) and other records that are used in specific applications. There is also data that is used by the DNS itself, such as the name server resource record (NS) that specifies a delegation of a zone or the RRSIG resource record used in DNS Security Extensions (DNSSEC).

When you introduce IPv6, you will use both IPv4 and IPv6 addresses in your network. Therefore, you need to add mappings from names to IPv6 addresses in parallel with the existing mapping from names to IPv4 addresses. One example of such a mapping, using the AAAA resource record type, is shown here:

```
www.ipv6.cisco.com. 86400 IN AAAA 2001:420:80:1::5
```

As you can see, a mapping from a name to an IPv6 address is performed using an AAAA resource record, with the IPv6 address given as a hexadecimal address (RFC 3596).

Task 1. Add AAAA records in your DNS server for the hostnames of the devices that can be reached through the IPv6 protocol.

Adding IPv6 addresses to DNS is important for the name servers themselves.

However, some applications also try to find the hostname given the IP address, and in IPv4 a PTR resource record type is used. In IPv6, PTR records are used as well, but the notation is different. The following example shows a PTR record for an IPv6 address according to the notation specified in RFC 3596. The address is hexadecimal, with a period between each hexadecimal digit. Note that the line is wrapped, and should be one line only.

```
5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.8.0.0.0.2.4.0.1.0.0.2.ip6.arpa. 900 IN
PTR www.ipv6.cisco.com.
```

Task 2. Add PTR records in your DNS server for the IP addresses of the devices that can be reached through the IPv6 protocol.

Having data in the namespace is not enough for successful DNS deployment in an IPv6 environment because that data is accessible only using the IPv4 protocol. This step is needed to start the use of IPv6 as a transport for the DNS protocol.

The DNS protocol is running over IP. It is defined to run over both TCP and User Datagram Protocol (UDP), and the server requests are sent to port 53 with random source ports. When introducing IPv6, you must also make sure that IPv6 can be used for the same DNS requests and responses for which you currently use IPv4. Figure 2 shows one example on how DNS can be deployed in an enterprise environment.

Figure 2. Sample Layout of DNS in an Enterprise Network

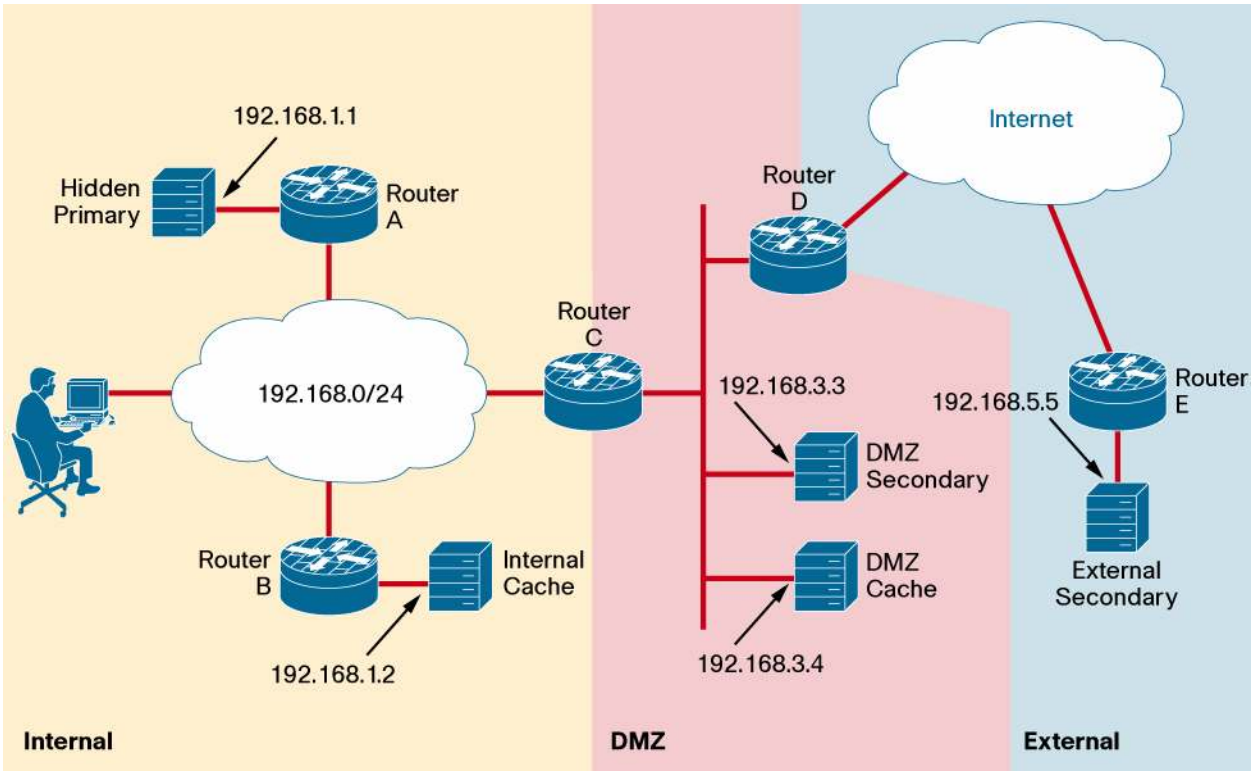


Figure 2 shows a normal layout of DNS in an enterprise network. It consists of three parts: DNS records publication, recursive resolvers, and policies related to DNS traffic in, for example, firewalls.

The DNS records you added to your zone in tasks 1 and 2 are added in the zone that originates on the hidden primary. From there, the records are distributed using zone transfers to the DMZ secondary and external secondary. Because the information is related to IPv6 addresses, you want those two servers that respond to requests from users on the Internet to be reachable through the IPv6 protocol. The first step to enable this communication is to enable IPv6 on those name servers.

Task 3. Enable IPv6 access to the authoritative DNS servers. Be sure that TCP/53 and UDP/53 can be accessed through IPv6.

As soon as tasks 1, 2, and 3 are completed, people will be able to send DNS requests to your DNS servers over IPv6 and get back, for example, IPv6 addresses for your web servers. However, if you also want your organization to be able to send email over IPv6 from your outgoing Simple Mail Transfer Protocol (SMTP) server to recipients that have only IPv6 addresses, then you must be able to send DNS requests over IPv6.

DNS, like SMTP, is not an end-to-end protocol. It uses proxies. When introducing IPv6, you can use this characteristic to make the implementation smoother. The client sending a DNS request sends the request to the full-service resolver of the client's choice (the internal cache in Figure 2). That resolver forwards the query to a different server (the DMZ cache in Figure 2), which resends the query to a number of authoritative servers until it gets a response. If you want to send responses over IPv6, it is sufficient to enable IPv6 from the recursive resolver that is, in fact, sending the requests.

Task 4. Enable IPv6 connectivity to the external full-service resolvers that send DNS queries to authoritative servers in the world.

Task 5. Make sure that the full-service resolver is configured with both IPv4 and IPv6 glue for the root servers in the world.

As soon as tasks 4 and 5 are complete, you can issue DNS queries from nodes on the enterprise internal network. Send the queries over IPv4 to the recursive resolvers, which will use either IPv4 or IPv6 as the carrier for the DNS requests they send to authoritative servers in the world.

When IPv6 is introduced on the internal network, the computers on the inside must send DNS queries over IPv6 to the internal recursive resolver (the internal cache in Figure 2). This process involves several steps: enabling IPv6 on the recursive resolver, enabling IPv6 on the local node, and configuring the stub resolver on the local node to use IPv6.

Task 6. Enable IPv6 on the recursive resolver so that it responds to DNS requests over IPv6 as well as IPv4.

Task 7. Enable IPv6 on the node that sends queries so that it can send DNS requests to the recursive resolver.

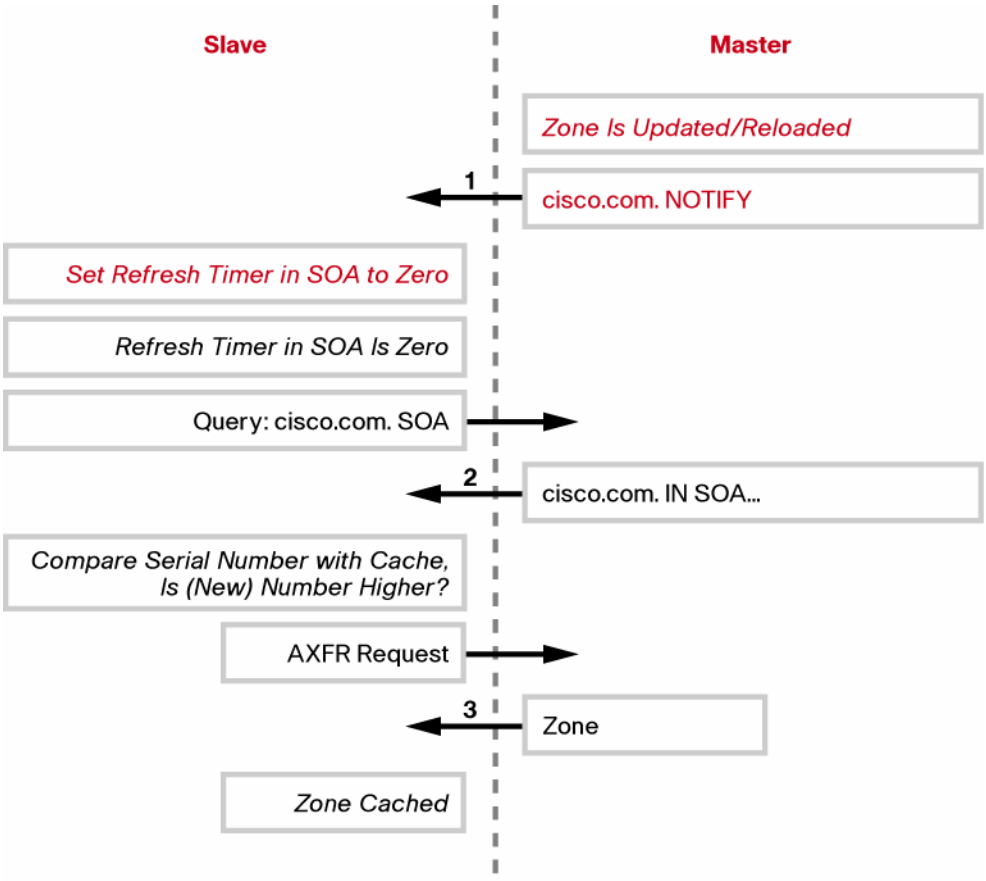
Of these tasks, the configuration of the stub resolver on the local node is the harder. The stub resolver can, of course, be configured statically, but most people want dynamic configuration. DNS resolver configuration for IPv4 is normally performed using DHCPv4: that is, the node sends a DHCP request and gets the resolver configuration back as part of the DHCP response. You can use DHCPv6 for IPv6, but DHCPv6 is not available in all operating systems. For example, Apple Mac OS X does not include DHCPv6 support. Those nodes that do not have DHCPv6 either must be statically configured or must use DHCPv4 and continue to use dual-stack techniques and send DNS queries over IPv4.

Task 8. Configure the stub resolver on the node that sends queries so that it uses IPv6 to send DNS queries, either statically or using DHCPv6.

After completing tasks 1 through 8, you are essentially done. However, you may have some policies in firewalls or routers (one of the routers A through E in Figure 2) that filter DNS over IPv6, TCP/53, or UDP/53. Stateful inspection is possible for DNS over IPv6 just as it is for DNS over IPv4, because in most cases sessions are initiated from the

inside toward the outside. In a few cases, however, when performing zone transfers (see Figure 3), the flows are initiated from the outside, and policies must be altered to enable this. Of the three flows in Figure 3, flows 2 and 3 are initiated toward the master server, which is the opposite direction from what is normally allowed in the flow inspection that protects the master server. However, the configuration is the same for IPv6 as for IPv4.

Figure 3. Flow of Packets When Performing Zone Transfer with DNS Notify



Task 9. Review policies for flows and make sure that both TCP/53 and UDP/53 can be accessed over IPv4 and IPv6.

In performing task 9, you should help ensure that modern DNS features are also allowed, such as large request and response sizes (greater than 512 bytes), EDNS0 extension, and support for unknown resource record types.

More complicated situations exist, specifically situations in which Network Address Translation (NAT) is used and in which an IPv6-only network requests access to IPv4-only nodes, but such situations are beyond the scope of this document.

For More Information

- *DNS and BIND, 5th Edition,* by Cricket Liu and Paul Albitz, O'Reilly Media, May 2006
- RFC 3596: DNS Extensions to Support IP Version 6, by S. Thomson, C. Huitema, V. Ksinant, and M. Souissi, October 2003 (format: TXT=14093 bytes)(obsoletes RFC 3152 and RFC 1886) (status: Draft Standard)

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Printed in USA C11-628652-00 10/10