



CHAPTER 14

Introduction to the Domain Name System

The Domain Name System (DNS) handles the growing number of Internet users. DNS translates names, such as `www.cisco.com`, into IP addresses, such as `192.168.40.0` (or the more extended IPv6 addresses), so that computers can communicate with each other. DNS makes using Internet applications, such as the World Wide Web, easy. The process is as if, when phoning your friends and relatives, you could autodial them based on their names instead of having to remember their phone numbers.

Related Topics

[How DNS Works, page 14-1](#)

[Domains, page 14-2](#)

[Nameservers, page 14-5](#)

[Reverse Nameservers, page 14-6](#)

[Authoritative and Caching DNS servers, page 14-7](#)

[High-Availability DNS, page 14-7](#)

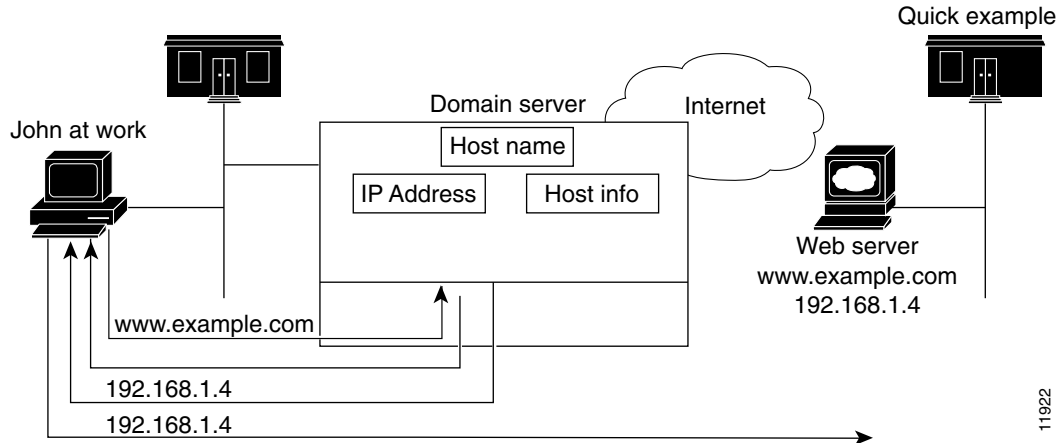
[About EDNS, page 14-7](#)

How DNS Works

To understand how DNS works, imagine a typical user, John, logging in to his computer. He launches his web browser so that he can view the website at a company, ExampleCo (see [Figure 14-1 on page 14-2](#)). He enters the name of their website—`http://www.example.com`. Then:

1. John's workstation sends a request to the DNS server about the IP address of `www.example.com`.
2. The DNS server checks its database to find that `www.example.com` corresponds to `192.168.1.4`.
3. The server returns this address to John's browser.
4. The browser uses the address to locate the website.
5. The browser displays the website on John's monitor.

Figure 14-1 Domain Names and Addresses

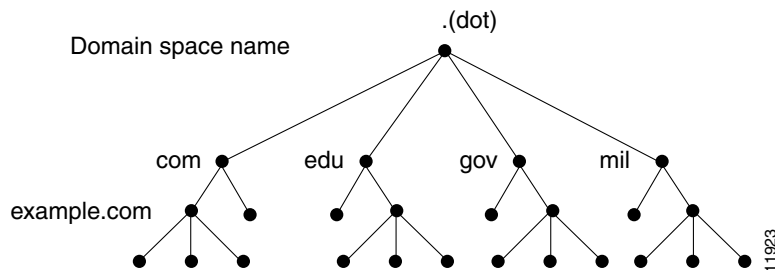


11922

Domains

John can access the ExampleCo website because his DNS server knows the `www.example.com` IP address. The server learned the address by searching through the domain namespace. DNS was designed as a tree structure, where each named domain is a node in the tree. The top-most node of the tree is the DNS root domain (`.`), under which there are subdomains, such as `.com`, `.edu`, `.gov`, and `.mil` (see Figure 14-2 on page 14-2).

Figure 14-2 Domain Name System Hierarchy



11923

The fully qualified domain name (FQDN) is a dot-separated string of all the network domains leading back to the root. This name is unique for each host on the Internet. The FQDN for the sample domain is `example.com.`, with its domain `example`, parent domain `.com`, and root domain `."` (`dot`).

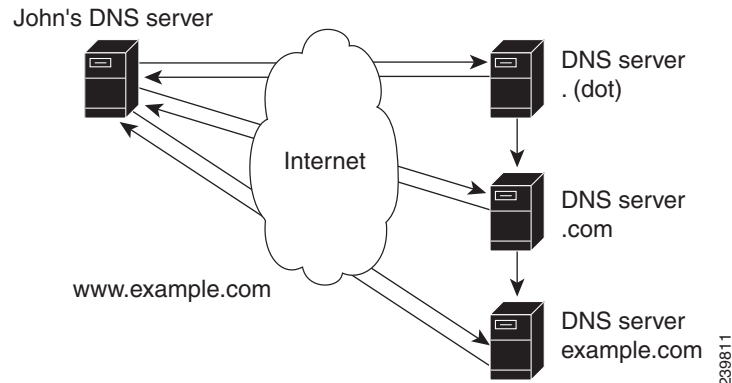
Related Topics

[Learning ExampleCo Address, page 14-3](#)
[Establishing a Domain, page 14-3](#)
[Difference Between Domains and Zones, page 14-3](#)

Learning ExampleCo Address

When John's workstation requests the IP address of the website `www.example.com` (see [Figure 14-3 on page 14-3](#)):

Figure 14-3 DNS Hierarchical Name Search



1. The local DNS server looks for the `www.example.com` domain in its database, but cannot find it, indicating that the server is not authoritative for this domain.
2. The server asks the authoritative root nameserver for the top-level (root) domain “.” (dot).
3. The root nameserver directs the query to a nameserver for the `.com` domain that knows about its subdomains.
4. The `.com` nameserver determines that `example.com` is one of its subdomains and responds with its server address.
5. The local server asks the `example.com` nameserver for the `www.example.com` location.
6. The `example.com` nameserver replies that its address is `192.168.1.4`.
7. The local server sends this address to John's Web browser.

Establishing a Domain

ExampleCo has a website that John could reach because it registered its domain with an accredited domain registry. ExampleCo also entered its domain name in the `.com` server database, and requested a network number, which defines a range of IP addresses.

In this case, the network number is `192.168.1.0`, which includes all assignable hosts in the range `192.168.1.1` through `192.168.1.254`. You can only have numbers 0 through 255 (2^8) in each of the address fields, known as octets. However, the numbers 0 and 255 are reserved for network and broadcast addresses, respectively, and are not used for hosts.

Difference Between Domains and Zones

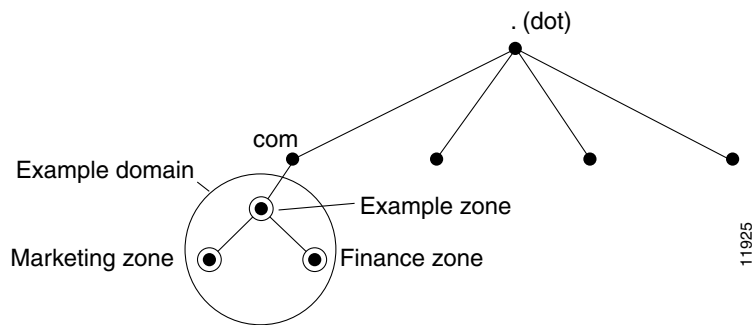
The domain namespace is divided into areas called zones that are points of delegation in the DNS tree. A zone contains all domains from a certain point downward, except those for which other zones are authoritative.

A zone usually has an authoritative nameserver, often more than one. In an organization, you can have many nameservers, but Internet clients can query only those that the root nameservers know. The other nameservers answer internal queries only.

The ExampleCo company registered its domain, example.com. It established three zones—example.com, marketing.example.com, and finance.example.com. ExampleCo delegated authority for marketing.example.com and finance.example.com to the DNS servers in the Marketing and Finance groups in the company. If someone queries example.com about hosts in marketing.example.com, example.com directs the query to the marketing.example.com nameserver.

In [Figure 14-4](#), the domain example.com includes three zones, with the example.com zone being authoritative only for itself.

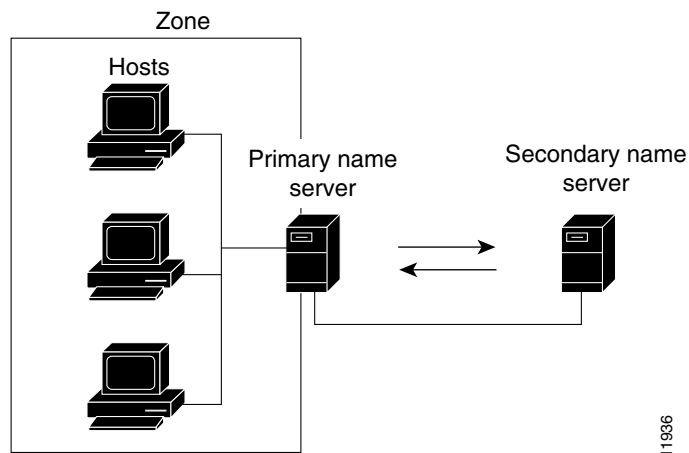
Figure 14-4 Example.com With Delegated Subdomains



ExampleCo could choose not to delegate authority to its subdomains. In that situation, the example.com domain is a zone that is authoritative for the subdomains for marketing and finance. The example.com server answers all outside queries about marketing and finance.

As you begin to configure zones by using Cisco Prime Network Registrar, you must configure a nameserver for each zone. Each zone has one primary server, which loads the zone contents from a local configuration database. Each zone can also have any number of secondary servers, which load the zone contents by fetching the data from the primary server. [Figure 14-5](#) shows a configuration with one secondary server.

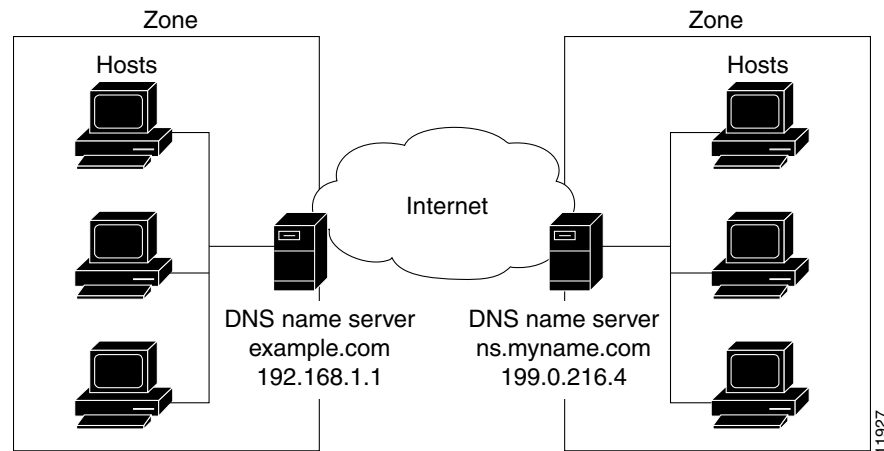
Figure 14-5 Primary and Secondary Servers for Zones



Nameservers

DNS is based on a client/server model. In this model, nameservers store data about a portion of the DNS database and provide it to clients that query the nameserver across the network. Nameservers are programs that run on a physical host and store zone data. As administrator for a domain, you set up a nameserver with the database of all the resource records (RRs) describing the hosts in your zone or zones (see [Figure 14-6 on page 14-5](#)).

Figure 14-6 Client/Server Name Resolution



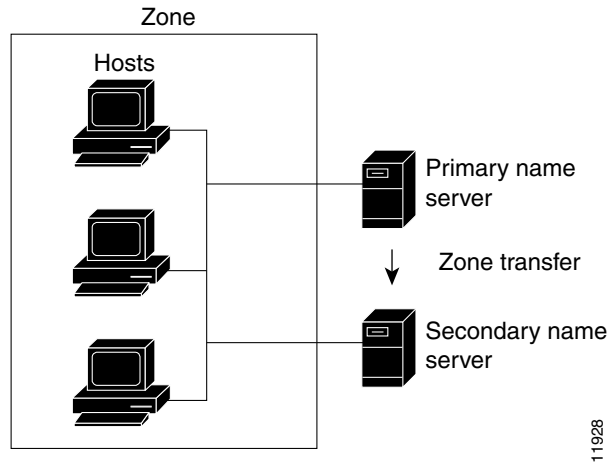
The DNS servers provide name-to-address translation, or name resolution. They interpret the information in a fully qualified domain name (FQDN) to find its address.

Each zone must have one primary nameserver that loads the zone contents from a local database, and a number of secondary servers, which load a copy of the data from the primary server (see [Figure 14-7 on page 14-6](#)). This process of updating the secondary server from the primary server is called a zone transfer.

Even though a secondary nameserver acts as a kind of backup to a primary server, both types of servers are authoritative for the zone. They both learn about hostnames in the zone from the zone authoritative database, not from information learned while answering queries. Clients can query both servers for name resolution.

As you configure the Cisco Prime Network Registrar DNS nameserver, you specify what role you want the server to perform for a zone—primary, secondary, or caching-only. The type of server is meaningful only in context to its role. A server can be a primary for some zones and a secondary for others. It can be a primary or secondary only, or it can serve no zones and just answer queries by means of its cache.

In Cisco Prime Network Registrar, the authoritative and caching services are separated and are handled by two separate servers. The authoritative server holds authoritative zone data and responds only to queries for which it is authoritative. The caching server is the recursive/caching server and does not contain any authoritative zone data.

Figure 14-7 DNS Zone Transfer

To configure the:

- Primary nameserver, see the [“Managing Primary DNS Servers”](#) section on page 15-4.
- Secondary server, see the [“Managing Secondary Servers”](#) section on page 15-14.

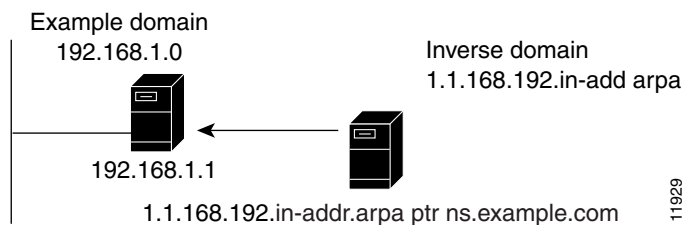
Reverse Nameservers

The DNS servers described so far perform name-to-address resolution. They can do this easily by searching through their database for the correct address, because they index all the data by name. However, there are times when you need address-to-name resolution so that you can interpret certain output, such as computer log files.

Finding a domain name when you only know the address, however, would require searching the entire namespace. DNS solves this problem by supporting a domain namespace that uses addresses as names, known as the `in-addr.arpa` domain. This reverse zone contains subdomains for each network based on the network number. For consistency and natural grouping, the four octets of a host number are reversed.

The IP address as a domain name appears backward, because the name is in leaf-to-root order. For example, the ExampleCo example domain network number is 192.168.1.0. Its reverse zone is 1.168.192.in-addr.arpa. If you only know the DNS server address (192.168.1.1), the query to the reverse domain would find the host entry 1.1.168.192.in-addr.arpa that maps back to example.com.

Reverse domains are handled through Pointer (PTR) RRs, as indicated in [Figure 14-8](#).

Figure 14-8 Reverse Domains

Authoritative and Caching DNS servers

Starting from release 8.0, the DNS server functionality is enhanced to provide separate DNS servers for authorization and caching. With this enhancement, Cisco Prime Network Registrar supports DNS64, DNSSEC, full IPv6, and has improved caching performance.

High-Availability DNS

Because there can be only one primary DNS server per zone, you risk the failure of dynamic updates if the primary DNS server goes down. These updates can occur on the primary DNS server only; a secondary DNS server cannot record these changes, but must forward them to the primary. To solve this problem, a second primary server can become a hot standby that shadows the main primary. This is called High-Availability (HA) DNS (see the [Chapter 19, “Configuring High-Availability DNS Servers”](#)). Both servers in this failover configuration must synchronize so that their primary zones and related attributes are identical. Cisco Prime Network Registrar provides settings on the main server to identify the main and backup for synchronization, and the timeout period to go over into failover mode.

About EDNS

To send a DNS message above 512 bytes over UDP, you need to use an extension of the DNS protocol known as Extended DNS (EDNS). The EDNS protocol expands the number of flags, label types, and return codes available to the DNS protocol. A version of EDNS specified by RFC 2671 is known as EDNS0. EDNS uses a pseudo resource record known as OPT Resource Record (OPT RR). OPT RR differentiates conventional DNS from EDNS. OPT RRs appear only in the route transmission between DNS clients and servers, they do not appear in the zone files or caches. A DNS endpoint that marks a DNS packet as EDNS must insert an OPT RR in the additional data section of the DNS request or response.

The DNS server supports all EDNS0 extensions. You can modify the UDP payload size of the DNS server. The minimum UDP payload size of the DNS server is 512 bytes and the maximum is 4 KB, with the default set to 4 KB.



Note

The DNS Server can handle requests from clients that do not support EDNS0, however, the DNS server is not permitted to use any extended capabilities, when it handles requests from clients that do not support EDNS0. The response to client requests are inserted into a default 512 byte message. To notify clients that the server supports EDNS0, an OPT RR is inserted into the additional section of the DNS message by the server.

