

DNS Resource Records

Document ID: 12684

Introduction

Prerequisites

- Requirements

- Components Used

- Conventions

Start of Authority

- SOA Record Data Fields

Name Service

Address

Host Information

Text

Mail Exchange

Canonical Name

Pointer

Related Information

Introduction

Resource Records define data types in the Domain Name System (DNS). Resource Records identified by RFC 1035 are stored in binary format internally for use by DNS software. But resource records are sent across a network in text format while they perform zone transfers. This document discusses some of the more important types of Resource Records.

Note: There are a number of other record types that are no longer actively supported. These include mail destination (MD), mail forwarder (MF), mail group (MG), mailbox or mail list information (MINFO), mail rename (MR), and NULL. You can obtain a full list of DNS Record Types from IANA DNS Parameters .

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

Start of Authority

At the top level of a domain, the name database must contain a Start of Authority (SOA) record. This SOA record identifies what is the best source of information for data within the domain. SOA contains the current

version of the DNS database, and various other parameters that define the behavior of a particular DNS server.

There must be exactly one SOA record for each nameserver domain (each subdomain). This applies to the subdomains of IN-ADDR.ARPA (reverse domains). A region of namespace that has a separate SOA is known as a zone.

The format for this record is seen in this output. The value listed for the time intervals in this SOA are those recommended by RFC 1537 .

```
DOMAIN.NAME.      IN      SOA      Hostname.Domain.Name. Mailbox.Domain.Name. (
1                ;      serial number
86400            ;      refresh in seconds (24 hours)
7200             ;      retry in seconds (2 hours)
2592000         ;      expire in seconds (30 days)
345600)         ;      TTL in seconds (4 days)
```

The SOA record for the fictional foo.edu might look something like this:

```
FOO.EDU.          IN      SOA      FOO.EDU. Joe_Smith.Foo.EDU. (
910612          ;      serial number
28800           ;      refresh in 8 hours
7200            ;      retry in 2 hours
604800          ;      expire in 7 days
86400 )         ;      TTL is 1 day
```

SOA Record Data Fields

This list provides an explanation of the data fields in the fictional SOA record.

- **DOMAIN.NAME.** The name of the domain to which the SOA record pertains. Note the trailing dot (.). This signifies that no suffix is to be appended to the name.
- **IN** The class of the DNS record. IN stands for "Internet."
- **SOA** The type of DNS record, the Start of Authority in this example.
- **Hostname.Domain.Name.** The "origin field" needs to contain the host name of the primary name server for this zone, the host where the authoritative data resides.
- **Mailbox.Domain.Name.** The mailbox of the individual responsible for (name service for) this domain. In order to translate this field into a usable E-mail address, replace the first dot (.) with an @ (at-sign). In this example, if there are problems with foo.edu, send an E-mail to Joe_Smith@foo.edu.
- **Serial number** The serial number of the current version of the DNS database for this domain. The serial number is the means by which other nameservers realize that your database has been updated. This serial number starts at 1 and must be a monotonically increasing integer. Do not put a decimal point into the serial number, as this can yield confusing and unpleasant results. Some DNS administrators use the date last modified as the serial number, in the format YYMMDDHHMM, others simply increment the serno by a small number every time the database is updated. The half parenthesis that occurs before the serno and closes after the minimum Time to Live (TTL) number allows the SOA to span multiple lines.

When a secondary nameserver for the foo.edu domain contacts the primary nameserver to check if there has been a change to the primary's DNS database, and if the secondary should do a zone transfer, it compares its own serial number against that of the primary nameserver.

If the serial number of the secondary nameserver is higher than that of the primary, a zone transfer does not occur. If the serial number of the primary nameserver is a higher number, the secondary nameserver performs a zone transfer and updates its own DNS database.

The other numeric fields are known as the TTL fields. These control the frequency with which nameservers poll each other to get information updates (for example, how long the data is cached, and so on).

- **Refresh** Tells the secondary nameserver how often to poll the primary nameserver and how often to check for a serial number change. This interval effects how long it takes for DNS changes made on the primary nameserver to propagate.
- **Retry** The interval per second at which the secondary nameserver tries to reconnect with the primary nameserver, in the event that it failed to connect at the Refresh interval.
- **Expire** The number of seconds after which a secondary nameserver needs to "expire" the data of the primary nameserver, if it fails to reconnect to the primary nameserver.
- **TTL** The default value that applies to all records in the DNS database on a nameserver. Each DNS resource record can have a TTL value configured. The default TTL of the SOA record is only used if a particular resource record does not have an explicit value configured. This value is supplied by authoritative nameservers (primary and secondary nameservers for a particular zone) when they respond to DNS queries.

Name Service

Each subdomain that is separately nameserved must have at least one corresponding Name Service (NS) record. Name servers use NS records to find each other.

An NS record takes this format:

```
DOMAIN.NAME.      IN      NS      Hostname.Domain.Name.
```

The value of an NS record for a domain is the name of the nameserver for that domain. You need to list an NS record for each primary or secondary nameserver for a domain.

Address

The Address record (A record) yields an IPv4 address that corresponds to a host name. There can be multiple IP addresses that correspond to a single hostname, there can also be multiple hostnames each of which maps to the same IP address.

An 'A' record takes this format:

```
Host.domain.name.      IN      A      xx.xx.xx.xx(IPv4 address)
```

There must be a valid 'A' record in the DNS for the Host.domain.name in order for a command, such as the **telnet host.domain.name** command, to work (or there must be a CNAME that points to a hostname with a valid 'A' record).

Note: DNS extensions for supporting IPv6 addresses are dealt with by RFC 1886 .

Host Information

The Host Information (HINFO) record can be set up to give hardware type and operating system (OS) information about each host. Its presence is optional, but having the information available can be useful.

There can be only one 'HINFO' record per host name.

A 'HINFO' record takes this format:

```
Host.DOMAIN.NAME.      IN      HINFO    "CPU type" "Operating System"
```

Note: Both the CPU type and OS fields are mandatory. If you want to leave one of these fields blank, specify it as " " (a blank space enclosed by double quotes). You cannot use just a pair of double quotes [""].

Note: The official machine names that you need for HINFO are found in RFC 1700 . RFC 1700 lists useful information such as /etc/services values, Ethernet manufacturer hardware addresses, and HINFO defaults.

Text

The Text (TXT) record allows you to associate any arbitrary text with a hostname. Some substandard implementations of the **bind** command do not support the 'TXT' record. However, some substandard implementations of the **bind** command do support a bogus record type called 'UINFO' that does the same thing. Cisco recommends you use only the 'TXT' record type.

You can have multiple 'TXT' records for a single host name.

A 'TXT' record takes this format:

```
Host.DOMAIN.NAME.  IN  TXT  "system manager: melvin@host.domain.name"
                  IN  TXT  "melasu"
```

Mail Exchange

A zone can have one or more Mail Exchange (MX) records. These records point to hosts that accept mail messages on behalf of the host. A host can be an 'MX' for itself. MX records need not point to a host in the same zone.

An 'MX' record takes this format:

```
Host.domain.name.      IN      MX      nn      Otherhost.domain.name.
                       IN      MX      nn      Otherhost2.domain.name.
```

The 'MX' preference numbers **nn** (value 0 to 65535) signify the order in which mailers select 'MX' records when they attempt mail delivery to the host. The lower the 'MX' number, the higher the host is in priority.

Canonical Name

The Canonical Name (CNAME) record is used to define an alias hostname.

A CNAME record takes this format:

```
alias.domain.name.    IN      CNAME   otherhost.domain.name.
```

This defines `alias.domain.name` as an alias for the host whose canonical (standard) name is `otherhost.domain.name`.

Note: A hostname that exists as a CNAME cannot have any other DNS records applied to it. For example, if your domain is called `philosophy.arizona.edu`, and it is separately nameserved (so that it has its own SOA and NS records), then you cannot give `philosophy.arizona.edu` a CNAME record. In order to send an E-mail to `anyuser@philosophy.arizona.edu`, you need to use MX and/or A records.

Pointer

Pointer records are the opposite of A Records and are used in Reverse Map zone files to map an IP address to a host name. Unlike the other SOA records, Pointer (PTR) records are used only in reverse (IN-ADDR.ARPA) domains. There must be exactly one PTR record for each Internet address. For example, if the host `gadzooks.poetry.arizona.edu` has an IP address of `128.196.47.55`, then there must be a PTR record for it in this format:

```
55.47.196.128.IN-ADDR.ARPA.      IN      PTR      gadzooks.poetry.arizona.edu.
```

Reverse domains contain mainly PTR records (plus SOA and NS records at the top).

The Berkeley `r`-utilities use the value of the PTR record for hostname authentication. Although DNS specifies that case is not significant in hostnames, be aware that some operating systems are sensitive to hostname case.

Related Information

- [Understanding the Domain Name System](#)
- [Resource Records](#)
- [Cisco Network Registrar](#)
- [Cisco IP Routing Technical Support](#)
- [Technical Support & Documentation – Cisco Systems](#)

All contents are Copyright © 2006–2007 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement.

Updated: Jul 18, 2007

Document ID: 12684
