Network Working Group
Request for Comments: 2782
Obsoletes: 2052
Category: Standards Track

A. Gulbrandsen
Troll Technologies
P. Vixie
Internet Software Consortium
L. Esibov
Microsoft Corp.

February 2000

A DNS RR for specifying the location of services (DNS SRV)

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document describes a DNS RR which specifies the location of the server(s) for a specific protocol and domain.

Overview and rationale

Currently, one must either know the exact address of a server to contact it, or broadcast a question.

The SRV RR allows administrators to use several servers for a single domain, to move services from host to host with little fuss, and to designate some hosts as primary servers for a service and others as backups.

Clients ask for a specific service/protocol for a specific domain (the word domain is used here in the strict RFC 1034 sense), and get back the names of any available servers.

Note that where this document refers to "address records", it means A RR's, AAAA RR's, or their most modern equivalent.

Definitions

The key words "MUST", "MUST NOT", "SHOULD", "SHOULD NOT" and "MAY" used in this document are to be interpreted as specified in [BCP 14]. Other terms used in this document are defined in the DNS specification, RFC 1034.

Applicability Statement

In general, it is expected that SRV records will be used by clients for applications where the relevant protocol specification indicates that clients should use the SRV record. Such specification MUST define the symbolic name to be used in the Service field of the SRV record as described below. It also MUST include security considerations. Service SRV records SHOULD NOT be used in the absence of such specification.

Introductory example

If a SRV-cognizant LDAP client wants to discover a LDAP server that supports TCP protocol and provides LDAP service for the domain example.com., it does a lookup of

```
_ldap._tcp.example.com
```

as described in [ARM]. The example zone file near the end of this memo contains answering RRs for an SRV query.

Note: LDAP is chosen as an example for illustrative purposes only, and the LDAP examples used in this document should not be considered a definitive statement on the recommended way for LDAP to use SRV records. As described in the earlier applicability section, consult the appropriate LDAP documents for the recommended procedures.

The format of the SRV RR

Here is the format of the SRV RR, whose DNS type code is 33:

```
_Service._Proto.Name TTL Class SRV Priority Weight Port Target
```

(There is an example near the end of this document.)

Service
: The symbolic name of the desired service, as defined in Assigned Numbers [STD 2] or locally. An underscore (_) is prepended to the service identifier to avoid collisions with DNS labels that occur in nature.

Some widely used services, notably POP, don't have a single universal name. If Assigned Numbers names the service indicated, that name is the only name which is legal for SRV lookups. The Service is case insensitive.

Proto
: The symbolic name of the desired protocol, with an underscore (_) prepended to prevent collisions with DNS labels that occur in nature. _TCP and _UDP are at present the most useful values for this field, though any name defined by Assigned Numbers or locally may be used (as for Service). The Proto is case insensitive.

Name
: The domain this RR refers to. The SRV RR is unique in that the name one searches for is not this name; the example near the end shows this clearly.

TTL
: Standard DNS meaning [RFC 1035].

Class
: Standard DNS meaning [RFC 1035]. SRV records occur in the IN Class.

Priority
: The priority of this target host. A client MUST attempt to contact the target host with the lowest-numbered priority it can reach; target hosts with the same priority SHOULD be tried in an order defined by the weight field. The range is 0-65535. This is a 16 bit unsigned integer in network byte order.

Weight
: A server selection mechanism. The weight field specifies a relative weight for entries with the same priority. Larger weights SHOULD be given a proportionately higher probability of being selected. The range of this number is 0-65535. This is a 16 bit unsigned integer in network byte order. Domain administrators SHOULD use Weight 0 when there isn't any server selection to do, to make the RR easier to read for humans (less noisy). In the presence of records containing weights greater than 0, records with weight 0 should have a very small chance of being selected.

  In the absence of a protocol whose specification calls for the use of other weighting information, a client arranges the SRV RRs of the same Priority in the order in which target hosts,

specified by the SRV RRs, will be contacted. The following algorithm SHOULD be used to order the SRV RRs of the same priority:

To select a target to be contacted next, arrange all SRV RRs (that have not been ordered yet) in any order, except that all those with weight 0 are placed at the beginning of the list.

Compute the sum of the weights of those RRs, and with each RR associate the running sum in the selected order. Then choose a uniform random number between 0 and the sum computed (inclusive), and select the RR whose running sum value is the first in the selected order which is greater than or equal to the random number selected. The target host specified in the selected SRV RR is the next one to be contacted by the client. Remove this SRV RR from the set of the unordered SRV RRs and apply the described algorithm to the unordered SRV RRs to select the next target host. Continue the ordering process until there are no unordered SRV RRs. This process is repeated for each Priority.

Port
The port on this target host of this service. The range is 0-65535. This is a 16 bit unsigned integer in network byte order. This is often as specified in Assigned Numbers but need not be.

Target
The domain name of the target host. There MUST be one or more address records for this name, the name MUST NOT be an alias (in the sense of RFC 1034 or RFC 2181). Implementors are urged, but not required, to return the address record(s) in the Additional Data section. Unless and until permitted by future standards action, name compression is not to be used for this field.

A Target of "." means that the service is decidedly not available at this domain.

Domain administrator advice

Expecting everyone to update their client applications when the first server publishes a SRV RR is futile (even if desirable). Therefore SRV would have to coexist with address record lookups for existing protocols, and DNS administrators should try to provide address records to support old clients:

- Where the services for a single domain are spread over several hosts, it seems advisable to have a list of address records at the same DNS node as the SRV RR, listing reasonable (if perhaps

suboptimal) fallback hosts for Telnet, NNTP and other protocols likely to be used with this name. Note that some programs only try the first address they get back from e.g. gethostbyname(), and we don't know how widespread this behavior is.

- Where one service is provided by several hosts, one can either provide address records for all the hosts (in which case the round-robin mechanism, where available, will share the load equally) or just for one (presumably the fastest).

- If a host is intended to provide a service only when the main server(s) is/are down, it probably shouldn't be listed in address records.

- Hosts that are referenced by backup address records must use the port number specified in Assigned Numbers for the service.

- Designers of future protocols for which "secondary servers" is not useful (or meaningful) may choose to not use SRV's support for secondary servers. Clients for such protocols may use or ignore SRV RRs with Priority higher than the RR with the lowest Priority for a domain.

Currently there's a practical limit of 512 bytes for DNS replies. Until all resolvers can handle larger responses, domain administrators are strongly advised to keep their SRV replies below 512 bytes.

All round numbers, wrote Dr. Johnson, are false, and these numbers are very round: A reply packet has a 30-byte overhead plus the name of the service ("_ldap._tcp.example.com" for instance); each SRV RR adds 20 bytes plus the name of the target host; each NS RR in the NS section is 15 bytes plus the name of the name server host; and finally each A RR in the additional data section is 20 bytes or so, and there are A's for each SRV and NS RR mentioned in the answer. This size estimate is extremely crude, but shouldn't underestimate the actual answer size by much. If an answer may be close to the limit, using a DNS query tool (e.g. "dig") to look at the actual answer is a good idea.

The "Weight" field

Weight, the server selection field, is not quite satisfactory, but the actual load on typical servers changes much too quickly to be kept around in DNS caches. It seems to the authors that offering administrators a way to say "this machine is three times as fast as that one" is the best that can practically be done.

The only way the authors can see of getting a "better" load figure is asking a separate server when the client selects a server and contacts it. For short-lived services an extra step in the connection establishment seems too expensive, and for long-lived services, the load figure may well be thrown off a minute after the connection is established when someone else starts or finishes a heavy job.

Note: There are currently various experiments at providing relative network proximity estimation, available bandwidth estimation, and similar services. Use of the SRV record with such facilities, and in particular the interpretation of the Weight field when these facilities are used, is for further study. Weight is only intended for static, not dynamic, server selection. Using SRV weight for dynamic server selection would require assigning unreasonably short TTLs to the SRV RRs, which would limit the usefulness of the DNS caching mechanism, thus increasing overall network load and decreasing overall reliability. Server selection via SRV is only intended to express static information such as "this server has a faster CPU than that one" or "this server has a much better network connection than that one".

The Port number

Currently, the translation from service name to port number happens at the client, often using a file such as /etc/services.

Moving this information to the DNS makes it less necessary to update these files on every single computer of the net every time a new service is added, and makes it possible to move standard services out of the "root-only" port range on unix.

Usage rules

A SRV-cognizant client SHOULD use this procedure to locate a list of servers and connect to the preferred one:

Do a lookup for QNAME=_service._protocol.target, QCLASS=IN, QTYPE=SRV.

If the reply is NOERROR, ANCOUNT>0 and there is at least one SRV RR which specifies the requested Service and Protocol in the reply:

If there is precisely one SRV RR, and its Target is "." (the root domain), abort.

```
            Else, for all such RR's, build a list of (Priority, Weight,
            Target) tuples

            Sort the list by priority (lowest number first)

            Create a new empty list

            For each distinct priority level
                While there are still elements left at this priority
                level

                    Select an element as specified above, in the
                    description of Weight in "The format of the SRV
                    RR" Section, and move it to the tail of the new
                    list

            For each element in the new list

                query the DNS for address records for the Target or
                use any such records found in the Additional Data
                section of the earlier SRV response.

                for each address record found, try to connect to the
               (protocol, address, service).

        else

            Do a lookup for QNAME=target, QCLASS=IN, QTYPE=A

            for each address record found, try to connect to the
           (protocol, address, service)
```

Notes:

- Port numbers SHOULD NOT be used in place of the symbolic service or protocol names (for the same reason why variant names cannot be allowed: Applications would have to do two or more lookups).

- If a truncated response comes back from an SRV query, the rules described in [RFC 2181] shall apply.

- A client MUST parse all of the RR's in the reply.

- If the Additional Data section doesn't contain address records for all the SRV RR's and the client may want to connect to the target host(s) involved, the client MUST look up the address record(s). (This happens quite often when the address record has shorter TTL than the SRV or NS RR's.)

- Future protocols could be designed to use SRV RR lookups as the means by which clients locate their servers.

Fictional example

This example uses fictional service "foobar" as an aid in understanding SRV records. If ever service "foobar" is implemented, it is not intended that it will necessarily use SRV records. This is (part of) the zone file for example.com, a still-unused domain:

```
$ORIGIN example.com.
@               SOA server.example.com. root.example.com. (
                    1995032001 3600 3600 604800 86400 )
                NS  server.example.com.
                NS  ns1.ip-provider.net.
                NS  ns2.ip-provider.net.
; foobar - use old-slow-box or new-fast-box if either is
; available, make three quarters of the logins go to
; new-fast-box.
_foobar._tcp    SRV 0 1 9 old-slow-box.example.com.
                 SRV 0 3 9 new-fast-box.example.com.
; if neither old-slow-box or new-fast-box is up, switch to
; using the sysdmin's box and the server
                 SRV 1 0 9 sysadmins-box.example.com.
                 SRV 1 0 9 server.example.com.
server           A   172.30.79.10
old-slow-box     A   172.30.79.11
sysadmins-box    A   172.30.79.12
new-fast-box     A   172.30.79.13
; NO other services are supported
*._tcp          SRV  0 0 0 .
*._udp          SRV  0 0 0 .
```

In this example, a client of the "foobar" service in the "example.com." domain needs an SRV lookup of "_foobar._tcp.example.com." and possibly A lookups of "new-fast-box.example.com." and/or the other hosts named. The size of the SRV reply is approximately 365 bytes:

30 bytes general overhead
20 bytes for the query string, "_foobar._tcp.example.com."
130 bytes for 4 SRV RR's, 20 bytes each plus the lengths of "new-fast-box", "old-slow-box", "server" and "sysadmins-box" - "example.com" in the query section is quoted here and doesn't need to be counted again.
75 bytes for 3 NS RRs, 15 bytes each plus the lengths of "server", "ns1.ip-provider.net." and "ns2" - again, "ip-provider.net." is quoted and only needs to be counted once.
120 bytes for the 6 address records (assuming IPv4 only) mentioned by the SRV and NS RR's.

IANA Considerations

The IANA has assigned RR type value 33 to the SRV RR. No other IANA services are required by this document.

Changes from RFC 2052

This document obsoletes RFC 2052. The major change from that previous, experimental, version of this specification is that now the protocol and service labels are prepended with an underscore, to lower the probability of an accidental clash with a similar name used for unrelated purposes. Aside from that, changes are only intended to increase the clarity and completeness of the document. This document especially clarifies the use of the Weight field of the SRV records.

Security Considerations

The authors believe this RR to not cause any new security problems. Some problems become more visible, though.

- The ability to specify ports on a fine-grained basis obviously changes how a router can filter packets. It becomes impossible to block internal clients from accessing specific external services, slightly harder to block internal users from running unauthorized services, and more important for the router operations and DNS operations personnel to cooperate.

- There is no way a site can keep its hosts from being referenced as servers. This could lead to denial of service.

- With SRV, DNS spoofers can supply false port numbers, as well as host names and addresses. Because this vulnerability exists already, with names and addresses, this is not a new vulnerability, merely a slightly extended one, with little practical effect.

References

STD 2: Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700, October 1994.

RFC 1034: Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

RFC 1035: Mockapetris, P., "Domain names - Implementation and Specification", STD 13, RFC 1035, November 1987.

RFC 974: Partridge, C., "Mail routing and the domain system", STD 14, RFC 974, January 1986.

BCP 14: Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

RFC 2181: Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, July 1997.

RFC 2219: Hamilton, M. and R. Wright, "Use of DNS Aliases for Network Services", BCP 17, RFC 2219, October 1997.

BCP 14: Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

ARM: Armijo, M., Esibov, L. and P. Leach, "Discovering LDAP Services with DNS", Work in Progress.

KDC-DNS: Hornstein, K. and J. Altman, "Distributing Kerberos KDC and Realm Information with DNS", Work in Progress.

Acknowledgements

The algorithm used to select from the weighted SRV RRs of equal priority is adapted from one supplied by Dan Bernstein.

Authors' Addresses

Arnt Gulbrandsen
Troll Tech
Waldemar Thranes gate 98B
N-0175 Oslo, Norway

Fax: +47 22806380
Phone: +47 22806390
EMail: arnt@troll.no

Paul Vixie
Internet Software Consortium
950 Charter Street
Redwood City, CA 94063

Phone: +1 650 779 7001

Levon Esibov
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

EMail: levone@microsoft.com

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.