Stream: Internet Engineering Task Force (IETF)
RFC: 9665
Category: Standards Track
Published: October 2024
ISSN: 2070-1721
Authors: T. Lemon *Apple Inc.*
S. Cheshire *Apple Inc.*

RFC 9665
Service Registration Protocol for DNS-Based Service Discovery

Abstract

The Service Registration Protocol (SRP) for DNS-based Service Discovery (DNS-SD) uses the standard DNS Update mechanism to enable DNS-SD using only unicast packets. This makes it possible to deploy DNS-SD without multicast, which greatly improves scalability and improves performance on networks where multicast service is not an optimal choice, particularly IEEE 802.11 (Wi-Fi) and IEEE 802.15.4 networks. DNS-SD Service registration uses public keys and SIG(0) to allow services to defend their registrations.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9665.

Copyright Notice

Table of Contents

1. Introduction

DNS-SD [RFC6763] is a component of Zero Configuration Networking [RFC6760] [ZC] [ROADMAP].

This document describes an enhancement to DNS-SD that allows servers to register the services they offer using the DNS protocol over unicast rather than using Multicast DNS (mDNS) [RFC6762]. There is already a large installed base of DNS-SD clients that can discover services using the DNS protocol (e.g., Android, Windows, Linux, Apple operating systems).

This document is intended for three audiences: Implementers of software that provides services that should be advertised using DNS-SD, implementers of authoritative DNS servers that will be used in contexts where DNS-SD registration is needed, and administrators of networks where DNS-SD service is required. The document is expected to provide sufficient information to allow interoperable implementation of the Service Registration Protocol.

DNS-SD allows servers to publish the information required to access the services they provide. DNS-SD clients can then discover the set of services of a particular type that are available. They can then select a service from among those that are available and obtain the information required to use it. Although DNS-SD using the DNS protocol can be more efficient and versatile than using mDNS, it is not common in practice because of the difficulties associated with updating authoritative DNS services with service information.

The existing practice for updating DNS zones is either to enter new data manually or to use DNS Update [RFC2136]. Unfortunately, DNS Update requires either:

- that the authoritative DNS server automatically trust updates or
- that the DNS Update requester have some kind of shared secret or public key that is known to the authoritative DNS server and can be used to authenticate the update.

Furthermore, DNS Update can be a fairly chatty process, requiring multiple roundtrips with different conditional predicates to complete the update process.

The Service Registration Protocol (SRP) adds a set of default heuristics for processing DNS updates that eliminates the need for conditional predicates. Instead, the SRP registrar (an authoritative DNS server that supports SRP Updates) has a set of default predicates that are applied to the update; and the update either succeeds entirely or fails in a way that allows the requester to know what went wrong and construct a new update.

SRP also adds a feature called "First Come, First Served Naming" (or "FCFS Naming"), which allows the requester to:

- claim a name that is not yet in use, and
- authenticate, using SIG(0) [RFC2931], both the initial claim (to ensure it has not been modified in transit) and subsequent updates (to ensure they come from the same entity that performed the initial claim).

This prevents a new service instance from "stealing" a name that is already in use: A second SRP requester attempting to claim an existing name will not possess the SIG(0) key used by the first requester to claim it. Because of this, its claim will be rejected. This will force it to choose a new name.

It is important to understand that "authenticate" here just means that we can tell that an update came from the same source as the original registration. We have not established trust. This has important implications for what we can and can't do with data the SRP requester sends us. You will notice as you read this document that we only support adding a very restricted set of records, and the content of those records is further constrained.

The reason for this is precisely that we have not established trust. So, we can only publish information that we feel safe in publishing even though we do not have any basis for trusting the requester. We reason that mDNS [RFC6762] allows arbitrary hosts on a single IP link to advertise services [RFC6763], relying on whatever service is advertised to provide authentication as a part of its protocol rather than in the service advertisement.

This is considered reasonably safe because it requires physical presence on the network in order to advertise. An off-network mDNS attack is simply not possible. Our goal with this specification is to impose similar constraints. Therefore, you will see in Section 3.3.1 that a very restricted set of records with a very restricted set of relationships are allowed. You will also see in Section 6.1 that we give advice on how to prevent off-network attacks.

This leads us to the disappointing observation that this protocol is not a mechanism for adding arbitrary information to DNS zones. We have not evaluated the security properties of adding, for example, an SOA record, an MX record, or a CNAME record; therefore, these are forbidden. Future updates to this specification might include analyses for other records and extend the set of records and/or record content that can be registered here. Or it might require establishment of trust, and add an authorization model to the authentication model we now have. But that is work for a future document.

Finally, SRP adds the concept of a "lease" [RFC9664], analogous to leases in DHCP [RFC2131] [RFC8415]. The SRP registration itself has a lease that may be on the order of two hours; if the requester does not renew the lease before it has elapsed, the registration is removed. The claim on the name can have a longer lease so that another requester cannot claim the name, even though the registration has expired.

The Service Registration Protocol for DNS-SD specified in this document provides a reasonably secure mechanism for publishing this information. Once published, these services can be readily discovered by DNS-SD clients using standard DNS lookups.

Section 10 of the DNS-SD specification [RFC6763] briefly discusses ways that servers can advertise the services they provide in the DNS namespace. In the case of mDNS, it allows servers to advertise their services on the local link, using names in the "local." namespace, which makes their services directly discoverable by peers attached to that same local link.

DNS-SD [RFC6763] also allows clients to discover services by using the DNS protocol over traditional unicast [RFC1035]. This can be done by having a system administrator manually configure service information in the DNS; however, manually populating DNS authoritative server databases is costly and potentially error-prone and requires a knowledgeable network administrator. Consequently, although all DNS-SD client implementations of which we are aware support DNS-SD using DNS queries, in practice it is used much less frequently than mDNS.

The Discovery Proxy [RFC8766] provides one way to automatically populate the DNS namespace but is only appropriate on networks where services are easily advertised using mDNS. The present document describes a solution more suitable for networks where multicast is inefficient, or where sleepy devices are common, by supporting the use of unicast for both the offering of and the discovery of services.

2. Conventions and Terminology Used in This Document

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Strictly speaking, fully qualified domain names end with a period. In DNS zone files and other similar contexts, if the final period is omitted, then a name may be treated incorrectly as relative to some other parent domain. This document follows the formal DNS convention, ending fully

qualified domain names with a period ("."). When this document mentions domain names such as "local." and "default.service.arpa.", the final period is part of the domain name and does not indicate the end of a sentence as it would in normal prose.

3. Service Registration Protocol

Services that implement SRP use DNS Update [RFC2136] with SIG(0) [RFC3007] to publish service information in the DNS. Two variants exist: One for full-featured hosts and one for devices designed for Constrained-Node Networks (CNNs) [RFC7228]. An SRP registrar is most likely an authoritative DNS server or is a source of data for one or more authoritative DNS servers. There is no requirement that the authoritative DNS server that is receiving SRP Updates be the same authoritative DNS server that is answering queries that return records that have been registered. For example, an SRP registrar could be the "hidden primary" that is the source of data for a fleet of secondary authoritative DNS servers.

3.1. Protocol Variants

3.1.1. Full-Featured Hosts

Full-featured hosts either are configured manually with a registration domain or discover the default registration domain automatically using the Domain Enumeration process described in Section 11 of the DNS-SD specification [RFC6763]. If this process does not produce a default registration domain, the SRP registrar is not discoverable on the local network using this mechanism. Other discovery mechanisms are possible, but they are out of scope for this document.

Configuration of the registration domain can be done either:

- by querying the list of available registration domains ("r._dns-sd._udp") and allowing the user to select one from the UI, or
- by any other means appropriate to the particular use case being addressed.

Full-featured devices construct the names of the SRV, TXT, and PTR records describing their service or services as subdomains of the chosen service registration domain. For these names, they then discover the zone apex of the closest enclosing DNS zone using SOA queries as described in Section 6.1 of the DNS Push Notification specification [RFC8765]. Having discovered the enclosing DNS zone, they query for the "_dnssd-srp._tcp.<zone>" SRV record to discover the SRP registrar to which they can send SRP Updates. Hosts that support SRP Updates using TLS use the "_dnssd-srp-tls._tcp.<zone>" SRV record instead.

Examples of full-featured hosts include devices such as home computers, laptops, powered peripherals with network connections (such as printers and home routers), and even battery-operated devices such as mobile phones that have long battery lives.

3.1.2. Constrained Hosts

For devices designed for CNNs [RFC7228], some simplifications are available. Instead of being configured with (or discovering) the service registration domain, the special-use domain name [RFC6761] "default.service.arpa." is used. The details of how SRP registrars are discovered will be specific to the constrained network; therefore, we do not suggest a specific mechanism here.

SRP requesters on CNNs are expected to receive, from the network, a list of SRP registrars with which to register. It is the responsibility of a CNN supporting SRP to provide one or more registrar addresses. It is the responsibility of the registrar supporting a CNN to handle the updates appropriately. In some network environments, updates may be accepted directly into a local "default.service.arpa." zone, which has only local visibility. In other network environments, updates for names ending in "default.service.arpa." may be rewritten by the registrar to names with broader visibility. Domain name rewriting should be performed as appropriate for the network environment in question. Some suggested techniques for how domain names can be translated from a locally scoped name to a domain name with larger scope can be found in the discussion of data translation for names in Multicast DNS answers in Section 5.5 of the Discovery Proxy specification [RFC8766].

3.1.3. Why two variants?

The reason for these different variants is that low-power devices that typically use CNNs may have very limited battery capacity. The series of DNS lookups required to discover an SRP registrar and then communicate with it will increase the energy required to advertise a service; for low-power devices, the additional flexibility this provides does not justify the additional use of energy. It is also fairly typical of such networks that some network service information is obtained as part of the process of joining the network; thus, this can be relied upon to provide nodes with the information they need.

Networks that are not CNNs can have more complicated topologies at the IP layer. Nodes connected to such networks can be assumed to be able to do DNS-SD service registration domain discovery. Such networks are generally able to provide registration domain discovery and routing. This creates the possibility of off-network spoofing, where a device from a foreign network registers a service on the local network in order to attack devices on the local network. To prevent such spoofing, TCP is required for such networks.

3.2. Protocol Details

We will discuss several parts to this process:

- how to know what to publish (Section 3.2.1),
- how to know where to publish it (under what name) (Section 3.2.2),
- how to publish it (Section 3.2.3),
- how to secure its publication (Section 3.2.4), and
- how to maintain the information once published (Section 5).

3.2.1. What to Publish

SRP Updates are sent by SRP requesters to SRP registrars. Three types of instructions appear in an SRP Update: Service Discovery instructions, Service Description instructions, and Host Description instructions. These instructions are made up of DNS Update Resource Records (RRs) that are either adds or deletes. The types of records that are added, updated, and removed in each of these instructions, as well as the constraints that apply to them, are described in Section 3.3. An SRP Update is a DNS Update message [RFC2136] that is constructed so as to meet the constraints described in that section. The following is a brief overview of what is included in a typical SRP Update:

- Service Discovery PTR RR(s) for service(s), which map from a generic service type (or subtype(s)) to a specific service instance name [RFC6763].
- For each service instance name, an SRV RR, one or more TXT RRs, and a KEY RR. Although, in principle, DNS-SD Service Description records can include other record types with the same service instance name, in practice, they rarely do. Currently, SRP does not permit other record types. The KEY RR is used to support FCFS Naming and has no specific meaning for DNS-SD lookups. SRV records for all services described in an SRP Update point to the same hostname.
- There is always exactly one hostname in a single SRP Update. A DNS Update containing more than one hostname is not an SRP Update. The hostname has one or more address RRs (AAAA or A) and a KEY RR (used for FCFS Naming). Depending on the use case, an SRP requester may be required to suppress some addresses that would not be usable by hosts discovering the service through the SRP registrar. The exact address record suppression behavior required may vary for different types of SRP requesters. Some suggested policies for suppressing unusable records can be found in Section 5.5.2 of the Discovery Proxy specification [RFC8766].

The DNS-Based Service Discovery specification [RFC6763] describes the details of what each of these RR types mean, with the exception of the KEY RR, which was defined in the specification for how to store Diffie-Hellman Keys in the DNS [RFC2539]. These specifications should be considered the definitive sources for information about what to publish; the reason for summarizing this here is to provide the reader with enough information about what will be published that the service registration process can be understood at a high level without first learning the full details of DNS-SD. Also, the "service instance name" is an important aspect of FCFS Naming, which we describe later on in this document.

3.2.2. Where to Publish It

Multicast DNS (mDNS) uses a single namespace, "local.". Subdomains of "local." are specific to the local link on which they are advertised. This convenience is not available for DNS-SD using the DNS protocol: Services must exist in some specific DNS namespace that is chosen either by the network operator or automatically.

As described above, full-featured devices are responsible for knowing the domain in which to register their services. Such devices **MAY** optionally support configuration of a registration domain by the operator of the device. However, such devices **MUST** support registration domain discovery as described in Section 11 of the DNS-SD specification [RFC6763].

Devices made for CNNs register in the special-use domain name [RFC6761] "default.service.arpa." and let the SRP registrar handle rewriting that to a different domain if necessary, as described in Section 3.1.2.

3.2.3. How to Publish It

It is possible to send a DNS Update message that does several things at once: For example, it's possible in a single transaction to add or update a single Host Description while also adding or updating the RRs comprising the Service Description(s) for one or more service instance(s) available on that host and adding or updating the RRs comprising the Service Discovery instruction(s) for those service instance(s).

An SRP Update takes advantage of this: It is implemented as a single DNS Update message that contains a service's Service Discovery records, Service Description records, and Host Description records.

Updates done according to this specification are somewhat different from normal DNS Updates [RFC2136] where the update process could involve many update attempts. You might first attempt to add a name if it doesn't exist; if that fails, then in a second message you might update the name if it does exist but matches certain preconditions. Because the Service Registration Protocol described in this document uses a single transaction, some of this adaptability is lost.

In order to allow updates to happen in a single transaction, SRP Updates do not include update prerequisites. The requirements specified in Section 3.3 are implicit in the processing of SRP Updates; thus, there is no need for the SRP requester to put in any explicit prerequisites.

3.2.3.1. How the DNS-SD Service Registration Process Differs from DNS Update

DNS-SD Service Registration uses the DNS Update specification [RFC2136] with some additions:

- It implements FCFS Naming, protected using SIG(0) [RFC2931].
- It enforces policy about what updates are allowed.
- It optionally performs rewriting of "default.service.arpa." to some other domain.
- It optionally performs automatic population of the address-to-name reverse mapping domains.
- An SRP registrar is not required to implement general DNS Update prerequisite processing.
- CNN SRP requesters are allowed to send updates to the generic domain "default.service.arpa.".

3.2.3.2. Retransmission Strategy

The DNS protocol, including DNS updates, can operate over UDP or TCP. When using UDP, reliable transmission must be guaranteed by retransmitting if a DNS UDP message is not acknowledged in a reasonable interval. Section 4.2.1 of the DNS specification [RFC1035] provides some guidance on this topic, as does Section 1 of the IETF document describing common DNS implementation errors [RFC1536]. Section 3.1.3 of the UDP Usage Guidelines document [RFC8085] also provides useful guidance that is particularly relevant to DNS.

3.2.3.3. Successive Updates

SRP does not require that every update contain the same information. When an SRP requester needs to send more than one SRP Update to the SRP registrar, it **SHOULD** combine these into a single SRP Update, when possible, subject to DNS message size limits and link-specific size limits (e.g., an IEEE 802.15.4 network will perform poorly when asked to deliver a packet larger than about 500 bytes). If the updates do not fit into a single SRP Update, then the SRP requester **MUST** send subsequent SRP Updates sequentially: Until an earlier SRP Update has been acknowledged, the requester **MUST NOT** send any subsequent SRP Updates. If a configuration change occurs while an outstanding SRP Update is in flight, the SRP registrar **MUST** defer sending a new SRP Update for that change until the previous SRP Update has completed.

3.2.4. How to Secure It

DNS Update messages can be secured using secret key transaction signatures (TSIG) [RFC8945]. This approach uses a secret key shared between the DNS Update requester (which issues the update) and the authoritative DNS server (which authenticates it). This model does not work for automatic service registration.

The goal of securing the DNS-SD Registration Protocol is to provide the best possible security given the constraint that service registration has to be automatic. It is possible to layer more operational security on top of what we describe here, but FCFS Naming is already an improvement over the security of mDNS.

3.2.4.1. FCFS Naming

FCFS Naming provides a limited degree of security. A server that registers its service using SRP is given ownership of a name for an extended period of time based on a lease specific to the key used to authenticate the SRP Update, which may be longer than the lease associated with the registered RRs. As long as the registrar remembers the name and the public key corresponding to the private key used to register RRs on that name, no other SRP requester can add or update the information associated with that name. If the SRP requester fails to renew its service registration before the KEY lease expires (Section 4 of the DNS Update Lease specification [RFC9664]) its name is no longer protected. FCFS Naming is used to protect both the Service Description and the Host Description.

3.2.5. SRP Requester Behavior

3.2.5.1. Public/Private Key Pair Generation and Storage

The requester generates a public/private key pair (Section 6.6). This key pair **MUST** be stored in stable storage; if there is no writable stable storage on the SRP requester, the SRP requester **MUST** be preconfigured with a public/private key pair in read-only storage. This key pair **MUST** be unique to the device. A device with rewritable storage **SHOULD** retain this key indefinitely. When the device changes ownership, it may be appropriate for the former owner to erase the old key pair, which would then require the new owner to install a new one. Therefore, the SRP requester on the device **SHOULD** provide a mechanism to erase the key (for example, as the result of a "factory reset") and to generate a new key.

Note that when a new key is generated, this will prevent the device from registering with the name associated with the old key in the same domain where it had previously registered. So, implicit in the generation of a new key is the generation of a new name; this can be done either proactively when regenerating a key or when the SRP update produces a name conflict.

The policy described here for managing keys assumes that the keys are only used for SRP. If a key that is used for SRP is also used for other purposes, the policy described here is likely to be insufficient. The policy stated here is **NOT RECOMMENDED** in such a situation: a policy appropriate to the full set of uses for the key must be chosen. Specifying such a policy is out of scope for this document.

When sending DNS updates, the requester includes a KEY record containing the public portion of the key in each Host Description Instruction and each Service Description Instruction. Each KEY record **MUST** contain the same public key. The update is signed using SIG(0), using the private key that corresponds to the public key in the KEY record. The lifetimes of the records in the update are set using the EDNS(0) Update Lease option [RFC9664].

The format of the KEY resource record in the SRP Update is defined in the IETF specification for DNSSEC Resource Records [RFC4034]. Because the KEY RR used in SIG(0) is not a zone-signing key, the flags field in the KEY RR **MUST** be all zeroes.

The KEY record in Service Description updates **MAY** be omitted for brevity; if it is omitted, the SRP registrar **MUST** behave as if the same KEY record that is given for the Host Description is also given for each Service Description for which no KEY record is provided. Omitted KEY records are not used when computing the SIG(0) signature.

3.2.5.2. Name Conflict Handling

"Add" operations for both Host Description RRs and Service Description RRs can have names that result in name conflicts. Service Discovery record "Add" operations cannot have name conflicts. If any Host Description or Service Description record is found by the SRP registrar to have a conflict with an existing name, the registrar will respond to the SRP Update with a YXDomain RCODE [RFC2136], indicating that the desired name is already owned by a different SIG(0) key. In this case, the SRP requester **MUST** choose a new name or give up.

There is no specific requirement for how the SRP requester should choose a new name. Typically, however, the requester will append a number to the preferred name. This number could be sequentially increasing or could be chosen randomly. One existing implementation attempts several sequential numbers before choosing randomly. For instance, it might try host.default.service.arpa., then host-1.default.service.arpa., then host-2.default.service.arpa., then host-31773.default.service.arpa.

3.2.5.3. Record Lifetimes

The lifetime of the DNS-SD PTR, SRV, A, AAAA, and TXT records [RFC6763] uses the LEASE field of the Update Lease option and is typically set to two hours. Thus, if a device is disconnected from the network, it does not continue to appear for too long in the user interfaces of devices looking for instances of that service type.

The lifetime of the KEY records is set using the KEY-LEASE field of the Update Lease Option and **SHOULD** be set to a much longer time, typically 14 days. The result being that even though a device may be temporarily unplugged -- disappearing from the network for a few days -- it makes a claim on its name that lasts much longer.

Therefore, even if a device is unplugged from the network for a few days, and its services are not available for that time, no other device can come along and claim its name the moment it disappears from the network. In the event that a device is unplugged from the network and permanently discarded, then its name is eventually cleaned up and made available for reuse.

3.2.5.4. Compression in SRV Records

Although the original SRV specification [RFC2782] requires that the target hostname in the rdata of an SRV record not be compressed in DNS queries and responses, an SRP requester **MAY** compress the target in the SRV record, since an SRP Update is neither a DNS query nor a DNS response. The motivation for *not* compressing is not stated in the SRV specification but is assumed to be because a recursive resolver (caching server) that does not understand the format of the SRV record might store it as binary data without decoding a compression pointer embedded with the target hostname field and thus return nonsensical rdata in response to a query. This concern does not apply in the case of SRP. An SRP registrar needs to understand SRV records in order to validate the SRP Update. Compression of the target can save space in the SRP Update, so we want SRP requesters to be able to assume that the registrar will handle this. Therefore, SRP registrars **MUST** support compression of SRV RR targets.

Note that this document does not update the SRV specification [RFC2782]: Authoritative DNS servers still **MUST NOT** compress SRV record targets. The requirement to accept compressed SRV records in updates only applies to SRP registrars, and SRP registrars that are also authoritative DNS servers still **MUST NOT** compress SRV record targets in DNS responses. We note also that Multicast DNS [RFC6762] similarly compresses SRV records in mDNS messages.

In addition, we note that an implementer of an SRP requester might update existing code that creates SRV records or compresses DNS messages so that it compresses the target of an SRV record. Care must be taken if such code is used both in requesters and in authoritative DNS servers that the code only compresses in the case where a requester is generating an SRP Update.

3.2.5.5. Removing Published Services

3.2.5.5.1. Removing All Published Services

To remove all the services registered to a particular hostname, the SRP requester transmits an SRP Update for that hostname with an Update Lease option that has a LEASE value of zero. The SRP Update **MUST** contain exactly one Host Description Instruction that contains exactly one "Delete All RRsets From A Name" instruction for the hostname and no "Add to an RRSet" instructions for that hostname. If the registration is to be permanently removed, KEY-LEASE **SHOULD** also be zero. Otherwise, it **SHOULD** be set to the same value it had previously; this holds the name in reserve for when the SRP requester is once again able to provide the service.

This method of removing services is intended for the case where the requester is going offline and does not want any of its services to continue being advertised.

To support this, when removing a hostname, an SRP registrar **MUST** remove all service instances pointing to that hostname and all Service Discovery PTR records pointing to those service instances, even if the SRP requester doesn't list them explicitly. If the KEY lease time is nonzero, the SRP registrar **MUST NOT** delete the KEY records for these SRP requesters.

3.2.5.5.2. Removing Some Published Services

In some use cases, a requester may need to remove a specific service without removing its other services. For example, a device may shut down its remote screen access (_rfb._tcp) service while retaining its command-line login (_ssh._tcp) service. This can be accomplished in one of two ways:

1. To simply remove a specific service, the requester sends a valid SRP Update with a Service Description Instruction (Section 3.3.1.2) containing a single "Delete All RRsets From A Name" update to the service instance name. The SRP Update **SHOULD** include Service Discovery Instructions (Section 3.3.1.1) consisting of "Delete An RR From An RRset" updates [RFC2136] that delete any Service Discovery PTR records whose target is the service instance name. However, even in the absence of such Service Discovery Instructions, the SRP registrar **MUST** delete any Service Discovery PTR records that point to the deleted service instance name.
2. When deleting one service instance while simultaneously creating a new service instance with a different service instance name, an alternative is to perform both operations using a single SRP Update. In this case, the old service is deleted as in the first alternative. The new service is added, just as it would be in an update that wasn't deleting the old service. Because both the removal of the old service and the add of the new service consists of a valid Service Discovery Instruction and a valid Service Description Instruction, the update as a whole is a valid SRP Update and will result in the old service being removed and the new one added; or, to put it differently, the SRP Update will result in the old service being replaced by the new service.

It is perhaps worth noting that if a service is being updated without the service instance name changing (for example, when only the target port in the SRV record is being updated), then that SRP Update will look very much like the second alternative above. The PTR record in the Service Discovery Instruction will be the same for both the "Delete An RR From An RRset" update and

the "Add To An RRset" update [RFC2136]. Since the removal of the old service and the addition of the new service are both valid SRP Update operations, the combined operation is a valid SRP Update operation. The SRP registrar does not need to include code to recognize this special case and does not need to take any special actions to handle it correctly.

Whichever of these two alternatives is used, the hostname lease will be updated with the lease time provided in the SRP update. In neither of these cases is it permissible to delete the hostname. All services must point to a hostname. If a hostname is to be deleted, this must be done using the method described in Section 3.2.5.5.1, which deletes the hostname and all services that have that hostname as their target.

3.3. Validation and Processing of SRP Updates

3.3.1. Validation of DNS Update Add and Delete RRs

The SRP registrar first validates that the DNS Update message is a syntactically and semantically valid DNS Update message according to the usual DNS Update rules [RFC2136].

SRP Updates consist of a set of *instructions* that together add or remove one or more services. Each *instruction* consists of one or more delete update(s), or one or more add update(s), or some combination of both delete updates and add updates.

The SRP registrar checks each instruction in the SRP Update to see that it is either a Service Discovery Instruction, a Service Description Instruction, or a Host Description Instruction. Order matters in DNS updates. Specifically, deletes must precede adds for records that the deletes would affect; otherwise, the add will have no effect. This is the only ordering constraint: Aside from this constraint, updates may appear in whatever order is convenient when constructing the update.

Because the SRP Update is a DNS update, it **MUST** contain a single entry in the Zone Section (what would be the Question Section in a traditional DNS message) that indicates the zone to be updated. Every delete and update in an SRP Update **MUST** be within the zone that is specified for the SRP Update.

3.3.1.1. Service Discovery Instruction

An instruction is a Service Discovery Instruction if it:

- consists of exactly one "Add To An RRSet" or exactly one "Delete An RR From An RRSet" RR update (Section 2.5 of the DNS Update specification [RFC2136]),
- which updates a PTR RR,
- the target of which is a service instance name
- for which name a Service Description Instruction is present in the SRP Update, and:
  - if the Service Discovery Instruction is an "Add To An RRSet" instruction, that Service Description Instruction contains a "Delete All RRsets From A Name" instruction for that service instance name followed by "Add To An RRset" instructions for the SRV and TXT records describing that service; or

- if the Service Discovery Instruction is a "Delete An RR From An RRSet" instruction, that Service Description Instruction contains a "Delete All RRsets From A Name" instruction for that service instance name with no following "Add To An RRset" instructions for the SRV and TXT records describing that service.

Note that there can be more than one Service Discovery Instruction for the same service name (the owner name of the Service Discovery PTR record) if the SRP requester is advertising more than one instance of the same service type or is changing the target of a PTR RR. When subtypes are being used (Section 7.1 of the DNS-SD specification [RFC6763]), each subtype is a separate Service Discovery Instruction. For each such PTR RR add or delete, the above constraints must be met.

3.3.1.2. Service Description Instruction

An instruction is a Service Description Instruction if, for the given service instance name, all of the following are true:

- It contains exactly one "Delete All RRsets From A Name" update for the service instance name (Section 2.5.3 of the DNS Update specification [RFC2136]).
- It contains zero or one "Add To An RRset" KEY RRs that, if present, contains the public key corresponding to the private key that was used to sign the message (if present, the KEY RR **MUST** match the KEY RR given in the Host Description).
- It contains zero or one "Add To An RRset" SRV RR.
- If an "Add To An RRSet" update for an SRV RR is present, there **MUST** be at least one "Add To An RRset" update for the corresponding TXT RR, and the target of the SRV RR **MUST** be the hostname given in the Host Description Instruction in the SRP Update, or
- If there is no "Add To An RRset" update for an SRV RR, then there **MUST** be no "Add To An RRset" updates for the corresponding TXT RR, and either:
- - the name to which the "Delete All RRsets From A Name" applies does not exist, or
  - there is an existing KEY RR on that name that matches the key with which the SRP Update was signed.

Service Description Instructions do not modify any other resource records.

An SRP registrar **MUST** correctly handle compressed names in the SRV target.

3.3.1.3. Host Description Instruction

Every SRP Update alway contains exactly one Host Description Instruction.

An instruction is a Host Description Instruction if, for the appropriate hostname, it contains the following:

- exactly one "Delete All RRsets From A Name" RR
- exactly one "Add To An RRset" RR that adds a KEY RR that contains the public key corresponding to the private key that was used to sign the message

- zero "Add To An RRset" operations (in the case of deleting a registration) or one or more "Add To An RRset" RRs of type A and/or AAAA (in the case of creating or updating a registration)

Host Description Instructions do not modify any other resource records.

A and/or AAAA records that are not of sufficient scope to be validly published in a DNS zone **MAY** be ignored by the SRP registrar, which could result in a Host Description effectively containing zero reachable addresses even when it contains one or more addresses.

For example, if an IPv4 link-local address [RFC3927] or an IPv6 link-local address [RFC4862] is provided by the SRP requester, the SRP registrar could elect not to publish this in a DNS zone. However, in some situations, the registrar might make the records available through a mechanism such as an advertising proxy only on the specific link from which the SRP Update originated. In such a situation, locally scoped records are still valid.

3.3.2. Valid SRP Update Requirements

An SRP Update **MUST** contain exactly one Host Description Instruction. Multiple Service Discovery updates and Service Description updates may be combined into a single single SRP Update along with a single Host Description update, as described in Section 3.2.3. A DNS Update message that contains any additional adds or deletes that cannot be identified as Service Discovery, Service Description, or Host Description Instructions is not an SRP Update. A DNS update that contains any prerequisites is not an SRP Update.

An SRP Update **MUST** include an EDNS(0) Update Lease option [RFC9664]. The LEASE time specified in the Update Lease option **MUST** be less than or equal to the KEY-LEASE time. A DNS update that does not include the Update Lease option, or that includes a KEY-LEASE value that is less than the LEASE value, is not an SRP Update.

When an SRP registrar receives a DNS Update message that is not an SRP update, it **MAY** process the update as normal DNS Update [RFC2136], including access control checks and constraint checks, if supported. Otherwise, the SRP registrar **MUST** reject the DNS Update with the Refused RCODE.

If the definitions of each of these instructions are followed carefully and the update requirements are validated correctly, many DNS Update messages that look very much like SRP Updates nevertheless will fail to validate. For example, a DNS update that contains an "Add To An RRset" instruction for a Service Name and an "Add to an RRset" instruction for a service instance name where the PTR record added to the Service Name does not reference the service instance name is not a valid SRP Update but may be a valid DNS Update.

3.3.3. FCFS Name and Signature Validation

Assuming that the SRP registrar has confirmed that a DNS Update message is a valid SRP Update (Section 3.3.2), it then checks that the name in the Host Description Instruction exists in the zone being updated. If so, then the registrar checks to see if the KEY record on that name is the same as the KEY record in the Host Description Instruction. The registrar performs the same check for the KEY records in any Service Description Instructions. For KEY records that were omitted from

Service Description Instructions, the KEY from the Host Description Instruction is used. If any existing KEY record corresponding to a KEY record in the SRP Update does not match the KEY record in the SRP Update (whether provided or taken from the Host Description Instruction), then the SRP registrar **MUST** reject the SRP Update with an YXDomain RCODE indicating that the desired name is already owned by a different SIG(0) key. This informs the SRP requester that it should select a different name and try again.

If the SRP Update is not in conflict with existing data in the zone being updated, the SRP registrar validates the SRP Update using SIG(0) against the public key in the KEY record of the Host Description Instruction. If the validation fails, the SRP Update is malformed, and the registrar **MUST** reject the SRP Update with the Refused RCODE. Otherwise, the SRP Update is considered valid and authentic and is processed as for a normal DNS Update [RFC2136].

KEY record updates omitted from Service Description Instruction(s) are processed as if they had been explicitly present. After the SRP Update has been applied, every Service Description that is updated **MUST** have a KEY RR, which **MUST** have the same valua as the KEY RR that is present in the Host Description to which the Service Description refers.

The IETF specification for DNSSEC Resource Records [RFC4034] states that the flags field in the KEY RR **MUST** be zero except for bit 7, which can be one in the case of a zone key. SRP requesters implementing this version of the SRP specification **MUST** set the flags field in the KEY RR to all zeroes. SRP registrars implementing this version of the SRP specification **MUST** accept and store the flags field in the KEY RR as received, without checking or modifying its value.

3.3.4. Handling of Service Subtypes

SRP registrars **MUST** treat the update instructions for a service type and all its subtypes as atomic. That is, when a service and its subtypes are being updated, whatever information appears in the SRP Update is the entirety of information about that service and its subtypes. If any subtype appeared in a previous update but does not appear in the current update, then the SRP registrar **MUST** remove that subtype.

There is intentionally no mechanism for deleting a single subtype individually. A delete of a service deletes all of its subtypes. To delete a single subtype individually, an SRP Update must be constructed that contains the service type and all subtypes for that service except for the subtype(s) to be deleted.

3.3.5. SRP Update Response

The status that is returned depends on the result of processing the update and can be either NoError, ServFail, Refused, or YXDomain. All other possible outcomes will already have been accounted for when applying the constraints that qualify the update as an SRP Update. The meanings of these responses are explained in Section 2.2 of the DNS Update specification [RFC2136].

In the case of a response other than NoError, Section 3.8 of the DNS Update specification [RFC2136] states that the authoritative DNS server is permitted to respond either with no RRs or to copy the RRs sent by the DNS Update client into the response. The SRP requester **MUST NOT**

attempt to validate any RRs that are included in the response. It is possible that a future SRP extension may include per-RR indications as to why the update failed, but at the time of writing this is not specified. So, if an SRP requester were to attempt to validate the RRs in the response, it might reject such a response, since it would contain RRs but probably not a set of RRs identical to what was sent in the SRP Update.

3.3.6. Optional Behavior

The SRP registrar **MAY** add a Reverse Mapping PTR record (described for IPv4 in Section 3.5 of [RFC1035] of the DNS specification [RFC1035] and for IPv6 in Section 2.5 of [RFC3596] of the later document updating DNS for IPv6 [RFC3596]) that corresponds to the Host Description. This is optional: The reverse mapping PTR record serves no essential protocol function. One reason to provide reverse mappings is that they can be used to annotate logs and network packet traces. In order for the registrar to do a reverse mapping update, it must be authoritative for the zone that would need to be updated or have credentials to do the update. The SRP requester **MAY** also do a reverse mapping update if it has credentials to do so.

The SRP registrar **MAY** apply additional criteria when accepting updates. In some networks, it may be possible to do out-of-band registration of keys and only accept updates from preregistered keys. In this case, an update for a key that has not been registered **SHOULD** be rejected with the Refused RCODE. When use of managed keys is desired, there are at least two benefits to doing this in conjunction with SRP rather than simply performing traditional DNS Updates using SIG(0) keys:

1. The same over-the-air registration protocol is used in both cases, so both use cases can be addressed by the same SRP requester implementation.
2. The Service Registration Protocol includes maintenance functionality not present with normal DNS updates.

Note that the semantics of using SRP in this way are different from the semantics of typical implementations of DNS Update. The KEY used to sign the SRP Update only allows the SRP requester to update records that refer to its Host Description. Implementations of a traditional DNS Update [RFC2136] do not normally provide a way to enforce a constraint of this type.

The SRP registrar could also have a dictionary of names or name patterns that are not permitted. If such a list is used, updates for service instance names that match entries in the dictionary are rejected with a Refused RCODE.

4. TTL Consistency

All RRs within an RRset are required to have the same TTL (required by Section 5.2 of the DNS Clarifications document [RFC2181]). In order to avoid inconsistencies, SRP places restrictions on TTLs sent by requesters and requires that SRP registrars enforce consistency.

Requesters sending SRP Updates **MUST** use consistent TTLs in all RRs within each RRset contained within an SRP Update.

SRP registrars **MUST** check that the TTLs for all RRs within each RRset contained within an SRP Update are the same. If they are not, the SRP update **MUST** be rejected with a Refused RCODE.

Additionally, when adding RRs to an RRset (for example, when processing Service Discovery records), the SRP registrar **MUST** use the same TTL on all RRs in the RRset. How this consistency is enforced is up to the implementation.

TTLs sent in SRP Updates are advisory: they indicate the SRP requester's guess as to what a good TTL would be. SRP registrars may override these TTLs. SRP registrars **SHOULD** ensure that TTLs are reasonable: neither too long nor too short. The TTL **SHOULD NOT** ever be longer than the lease time (Section 5.1). Shorter TTLs will result in more frequent data refreshes; this increases latency on the DNS-SD client side, increases load on any caching resolvers and on the authoritative DNS server, and also increases network load, which may be an issue for CNNs. Longer TTLs will increase the likelihood that data in caches will be stale. TTL minimums and maximums **SHOULD** be configurable by the operator of the SRP registrar.

5. Maintenance

5.1. Cleaning Up Stale Data

Because the DNS-SD Service Registration Protocol is automatic and not managed by humans, some additional bookkeeping is required. When an update is constructed by the SRP requester, it **MUST** include an EDNS(0) Update Lease Option [RFC9664]. The Update Lease Option contains two lease times: the Lease Time and the KEY Lease Time.

Similar to DHCP leases [RFC2131], these leases are promises from the SRP requester that it will send a new update for the service registration before the lease time expires. The Lease time is chosen to represent the duration after the update during which the registered records other than the KEY record can be assumed to be valid. The KEY lease time represents the duration after the update during which the KEY record can be assumed to be valid. The reasoning behind the different lease times is discussed in Sections 3.2.4.1 and 3.2.5.3.

SRP registrars may be configured with limits for these values. At the time of writing, a default limit of two hours for the Lease and 14 days for the SIG(0) KEY are thought to be good choices. Devices with limited battery that wake infrequently are likely to request longer leases; registrars that support such devices may need to set higher limits. SRP requesters that are going to continue to use names on which they hold leases **SHOULD** refresh them well before the lease ends in case the registrar is temporarily unavailable or under heavy load.

The lease time applies specifically to the hostname. All service instances, and all service entries for such service instances, depend on the hostname. When the lease on a hostname expires, the hostname and all services that reference it **MUST** be removed at the same time: It is never valid for a service instance to remain when the hostname it references has been removed. If the KEY record for the hostname is to remain, the KEY record for any services that reference it **MUST** also remain. However, the Service Discovery PTR record **MUST** be removed since it has no key associated with it and since it is never valid to have a Service Discovery PTR record for which there is no service instance on the target of the PTR record.

SRP registrars **MUST** also track a lease time per service instance. The reason being that a requester may re-register a hostname with a different set of services and not remember that some different service instance had previously been registered. In this case, when that service instance lease expires, the SRP registrar **MUST** remove the service instance, and any associated Service Discovery PTR records pointing to that service instance, (although the KEY record for the service instance **SHOULD** be retained until the KEY lease on that service expires). This is beneficial because it avoids stale services continuing to be advertised after the SRP requester has forgotten about them.

The SRP registrar **MUST** include an EDNS(0) Update Lease option in the response. The requester **MUST** check for the EDNS(0) Update Lease option in the response, and when deciding when to renew its registration the requester **MUST** use the lease times from that received option in place of the lease times that it originally requested from the registrar. The times may be shorter or longer than those specified in the SRP Update. The SRP requester must honor them in either case.

SRP requesters **SHOULD** assume that each lease ends N seconds after the update was first transmitted (where N is the granted lease duration). SRP registrars **SHOULD** assume that each lease ends N seconds after the update that was successfully processed was received. Because the registrar will always receive the update after the SRP requester sent it, this avoids the possibility of a race condition where the SRP registrar prematurely removes a service when the SRP requester thinks the lease has not yet expired. In addition, the SRP requester **MUST** begin attempting to renew its lease in advance of the expected expiration time, as required by the DNS Update Lease specification [RFC9664], to accomodate the situation where the clocks on the SRP requester and the SRP registrar do not run at precisely the same rate.

SRP registrars **MUST** reject updates that do not include an EDNS(0) Update Lease option. DNS authoritative servers that allow both SRP and non-SRP DNS updates **MAY** accept updates that don't include leases, but they **SHOULD** differentiate between SRP Updates and other updates and **MUST** reject updates that would otherwise be SRP Updates if they do not include leases.

The function of Lease times and the function of TTLs are completely different. On an authoritative DNS server, the TTL on a resource record is a constant. Whenever that RR is served in a DNS response, the TTL value sent in the answer is the same. The lease time is never sent as a TTL; its sole purpose is to determine when the authoritative DNS server will delete stale records. It is not an error to send a DNS response with a TTL of M when the remaining time on the lease is less than M.

6. Security Considerations

6.1. Source Validation

SRP Updates have no authorization semantics other than "First Come, First Served" (FCFS). Thus, if an attacker from outside the administrative domain of the SRP registrar knows the registrar's IP address, it can, in principle, send updates to the registrar that will be processed successfully. Therefore, SRP registrars **SHOULD** be configured to reject updates from source addresses outside of the administrative domain of the registrar.

For TCP updates, the initial SYN-SYN+ACK handshake prevents updates being forged by an off-path attacker. In order to ensure that this handshake happens, SRP registrars relying on three-way-handshake validation **MUST NOT** accept TCP Fast Open payloads [RFC7413]. If the network infrastructure allows it, an SRP registrar **MAY** accept TCP Fast Open payloads if all such packets are validated along the path, and the network is able to reject this type of spoofing at all ingress points.

For UDP updates from CNN devices, spoofing would have to be prevented with appropriate source address filtering on routers [RFC2827]. This would ordinarily be accomplished by measures such as those described in Section 4.5 of the IPv6 CE Router Requirements document [RFC7084]. For example, a stub router [SNAC-SIMPLE] for a CNN might only accept UDP updates from source addresses known to be on-link on that stub network and might further validate that the UDP update was actually received on the stub network interface and not the interface connected to the adjacent infrastructure link.

6.2. Other DNS Updates

Note that these rules only apply to the validation of SRP Updates. An authoritative DNS server that accepts updates from SRP requesters may also accept other DNS Update messages, and those DNS Update messages may be validated using different rules. However, in the case of an authoritative DNS server that accepts SRP updates, the intersection of the SRP Update rules and whatever other update rules are present must be considered very carefully.

For example, a normal authenticated DNS update to any RR that was added using SRP, but is authenticated using a different key, could be used to override a promise made by the SRP registrar to an SRP requester by replacing all or part of the service registration information with information provided by an authenticated DNS update requester. An implementation that allows both kinds of updates **SHOULD NOT** allow DNS Update requesters that are using different authentication and authorization credentials to update records added by SRP requesters.

6.3. Risks of Allowing Arbitrary Names to be Registered in SRP Updates

It is possible to set up SRP Updates for a zone that is also used for non-DNS-SD records. For example, imagine that you set up SRP service for example.com. SRP requesters can now register names like "www" or "mail" or "smtp" in this domain. In addition, SRP Updates using FCFS Naming can insert names that are obscene or offensive into the zone. There is no simple solution to these problems. However, we have two recommendations to address this problem:

- Do not provide SRP service in organization-level zones. Use subdomains of the organizational domain for DNS-SD. This does not prevent registering names as mentioned above but does ensure that genuinely important names are not accidentally claimed by SRP requesters. So, for example, the zone "dnssd.example.com." could be used instead of "example.com." for SRP Updates. Because of the way that DNS-browsing domains are discovered, there is no need for the DNS-SD discovery zone that is updated by SRP to have a user-friendly or important-sounding name.
- Configure a dictionary of names that are prohibited. Dictionaries of common obscene and offensive names are no doubt available and can be augmented with a list of typical "special"

names like "www", "mail", "smtp", and so on. Lists of names are generally available or can be constructed manually. Names rejected due to this should return a Refused RCODE, indicating to the SRP requester that it should not append or increment a number at the end of the name and then try again, since this would likely result in an infinite loop. If a name is considered unacceptable because it is obscene or offensive, adding a number on the end is unlikely to make the name acceptable.

6.4. Security of Local Service Discovery

Local links can be protected by managed services such as RA Guard [RFC6105], but multicast services like DHCP [RFC2131], DHCPv6 [RFC8415], and IPv6 Neighbor Discovery [RFC4861] are, in most cases, not authenticated and can't be controlled on unmanaged networks, such as home networks and small office networks where no network management staff are present. In such situations, the SRP service has comparatively fewer potential security exposures and, hence, is not the weak link. This is discussed in more detail in Section 3.2.4.

The fundamental protection for networks of this type is the user's choice of what devices to add to the network. Work is being done in other working groups and standards bodies to improve the state of the art for network on-boarding and device isolation (e.g., Manufacturer Usage Descriptions [RFC8520] provide a means for constraining what behaviors are allowed for a device in an automatic way), but such work is out of scope for this document.

6.5. SRP Registrar Authentication

This specification does not provide a mechanism for validating responses from SRP registrars to SRP requesters. In principle, a KEY RR could be used by a non-CNN SRP requester to validate responses from the registrar, but this is not required, nor do we specify a mechanism for determining which key to use.

In addition, for DNS-over-TLS connections, out-of-band key pinning as described in Section 4.2 of the DNS-over-TLS specification [RFC7858] could be used for authentication of the SRP registrar, e.g., to prevent man-in-the-middle attacks. However, the use of such keys is impractical for an unmanaged service registration protocol; hence, it is out of scope for this document.

6.6. Required Signature Algorithm

For validation, SRP registrars **MUST** implement the ECDSAP256SHA256 signature algorithm. SRP registrars **SHOULD** implement the algorithms that are listed in Section 3.1 of the DNSSEC Cryptographic Algorithms specification [RFC8624], in the validation column of the table, that are numbered 13 or higher and that have a "**MUST**", "**RECOMMENDED**", or "**MAY**" designation in the validation column of the table. SRP requesters **MUST NOT** assume that any algorithm numbered lower than 13 is available for use in validating SIG(0) signatures.

7. Privacy Considerations

Because DNS-SD SRP Updates can be sent off-link, the privacy implications of SRP are different from those for mDNS responses. SRP Requester implementations that are using TCP **SHOULD** also use DNS-over-TLS [RFC7858] if available. SRP registrar implementations **MUST** offer TLS support. Because there is no mechanism for sharing keys, validation of DNS-over-TLS keys is not possible; DNS-over-TLS is used only for Opportunistic Privacy, as documented in Section 4.1 of the DNS-over-TLS specification [RFC7858].

SRP requesters that are able to use TLS **SHOULD NOT** fall back to TCP. Since all SRP registrars are required to support TLS, whether to use TLS is entirely the decision of the SRP requester.

Public keys can be used as identifiers to track hosts. SRP registrars **MAY** elect not to return KEY records for queries for SRP registrations. To avoid DNSSEC validation failures, an SRP registrar that signs the zone for DNSSEC but refuses to return a KEY record **MUST NOT** store the KEY record in the zone itself. Because the KEY record isn't in the zone, the nonexistence of the KEY record can be validated. If the zone is not signed, the authoritative DNS server **MAY** instead return a negative non-error response (either NXDOMAIN or no data).

8. Domain Name Reservation Considerations

This section specifies considerations for systems involved in domain name resolution when resolving queries for names ending with ".service.arpa.". Each item in this section addresses some aspect of the DNS or the process of resolving domain names that would be affected by this special-use allocation. Detailed explanations of these items can be found in Section 5 of the Special-Use Domain Names specification [RFC6761].

8.1. Users

The current proposed use for "service.arpa." does not require special knowledge on the part of the user. While the "default.service.arpa." subdomain is used as a generic name for registration, users are not expected to see this name in user interfaces. In the event that it does show up in a user interface, it is just a domain name and requires no special treatment by the user.

8.2. Application Software

Application software does not need to handle subdomains of "service.arpa." specially. "service.arpa." **SHOULD NOT** be treated as more trustworthy than any other insecure DNS domain, simply because it is locally served (or for any other reason). It is not possible to register a PKI certificate for a subdomain of "service.arpa." because it is a locally served domain name. So, no such subdomain can be considered to be uniquely identifying a particular host, as would be required for such a PKI certificate to be issued. If a subdomain of "service.arpa." is returned by an API or entered in an input field of an application, PKI authentication of the endpoint being identified by the name will not be possible. Alternative methods and practices for authenticating such endpoints are out of scope for this document.

8.3. Name Resolution APIs and Libraries

Name resolution APIs and libraries **MUST NOT** recognize names that end in "service.arpa." as special and **MUST NOT** treat them as having special significance, except that it may be necessary that such APIs not bypass the locally discovered recursive resolvers.

One or more IP addresses for recursive resolvers will usually be supplied to the SRP requester through router advertisements or DHCP. For an administrative domain that uses subdomains of "service.arpa.", the recursive resolvers provided by that domain will be able to answer queries for subdomains of "service.arpa.". Other (non-local) resolvers will not, or they will provide answers that are not correct within that administrative domain.

A host that is configured to use a resolver other than one that has been provided by the local network may not be able to resolve or may receive incorrect results for subdomains of "service.arpa.". In order to avoid this, hosts **SHOULD** use the resolvers that are locally provided for resolving "service.arpa." names, even when they are configured to use other resolvers for other names.

8.4. Recursive Resolvers

There are two considerations for recursive resolvers (also known as "caching DNS servers" or "recursive DNS servers") that follow this specification:

1. For correctness, recursive resolvers at sites using 'service.arpa.' must, in practice, transparently support DNSSEC queries: queries for DNSSEC records and queries with the DNSSEC OK (DO) bit set (Section 3.2.1 of the DNSSEC specification [RFC4035]). DNSSEC validation [RFC9364] is a best current practice: Although validation is not required, a caching recursive resolver that does not validate answers that can be validated may cache invalid data. In turn, this would prevent validating stub resolvers from successfully validating answers. Hence, as a practical matter, recursive resolvers at sites using "service.arpa." should do DNSSEC validation.
2. Unless configured otherwise, recursive resolvers and DNS proxies **MUST** behave following the rules prescribed for Iterative Resolvers in Section 3 of the IETF Locally Served DNS Zones document [RFC6303]. That is, queries for "service.arpa." and subdomains of "service.arpa." **MUST NOT** be forwarded, with one important exception: a query for a DS record with the DO bit set **MUST** return the correct answer for that question, including correct information in the authority section that proves that the record is nonexistent.

   So, for example, a query for the NS record for "service.arpa." **MUST NOT** result in that query being forwarded to an upstream cache nor to the authoritative DNS server for ".arpa.". However, to provide accurate authority information, a query for the DS record **MUST** result in forwarding whatever queries are necessary. Typically, this will just be a query for the DS record since the necessary authority information will be included in the authority section of the response if the DO bit is set.

8.5. Authoritative DNS Servers

No special processing of "service.arpa." is required for authoritative DNS server implementations. It is possible that an authoritative DNS server might attempt to check the authoritative DNS servers for "service.arpa." for a delegation beneath that name before answering authoritatively for such a delegated name. In such a case, because the name always has only local significance, there will be no such delegation in the "service.arpa." zone; therefore, the authoritative DNS server would refuse to answer authoritatively for such a zone. An authoritative DNS server that implements this sort of check **MUST** be configurable so that either it does not do this check for the "service.arpa." domain or it ignores the results of the check.

8.6. DNS Server Operators

DNS server operators **MAY** configure an authoritative DNS server for "service.arpa." for use with SRP. The operator for the DNS servers that are authoritative for "service.arpa." in the global DNS will configure any such DNS servers as described in Section 9.

8.7. DNS Registries/Registrars

"service.arpa." is a subdomain of the "arpa." top-level domain, which is operated by IANA under the authority of the Internet Architecture Board (IAB) [RFC3172]. There are no other DNS registrars for "arpa.".

9. Delegation of "service.arpa."

The owner of the "arpa." zone, at the time of writing the IAB [IAB-ARPA], has added a delegation of "service.arpa." in the "arpa." zone [RFC3172], following the guidance provided in Section 7 of the "home.arpa." specification [RFC8375].

10. IANA Considerations

10.1. Registration and Delegation of "service.arpa." as a Special-Use Domain Name

IANA has recorded the domain name "service.arpa." in the "Special-Use Domain Names" registry [SUDN]. IANA has implemented the delegation requested in Section 9.

10.2. Addition of "service.arpa." to the Locally-Served Zones Registry

IANA has also added a new entry to the "Transport-Independent Locally-Served Zones Registry" registry of the "Locally-Served DNS Zones" group [LSDZ]. The entry is for the domain "SERVICE.ARPA." with the description "DNS-SD Service Registration Protocol Special-Use Domain" and lists this document as the reference.

10.3. Subdomains of "service.arpa."

This document only makes use of the "default.service.arpa." subdomain of "service.arpa." Other subdomains are reserved for future use by DNS-SD or related work. IANA has created the "service.arpa. Subdomain" registry [SUB]. The IETF has change control for this registry. New entries may be added either as a result of Standards Action (Section 4.9 of [RFC8126]) or with IESG Approval (Section 4.10 of [RFC8126]), provided that the values and their meanings are documented in a permanent and readily available public specification, in sufficient detail so that interoperability between independent implementations is possible.

IANA has grouped the "service.arpa. Subdomain" registry with the "Locally-Served DNS Zones" group. The registry is a table with three columns: the subdomain name (expressed as a fully qualified domain name), a brief description of how it is used, and a reference to the document that describes its use in detail.

This initial contents of this registry are as follows:

Subdomain Name	Description	Reference
default.service.arpa.	Default domain for SRP Updates	RFC 9665

Table 1

10.4. Service Name Registrations

IANA has added two new entries to the "Service Name and Transport Protocol Port Number Registry" [PORT]. The following subsections contain tables with the fields required by Section 8.1.1 of IANA's Procedures for Service Name allocation [RFC6335].

10.4.1. "dnssd-srp" Service Name

Field Name	Value
Service Name	dnssd-srp
Transport Protocol	tcp
Assignee	IESG <iesg@ietf.org>
Contact	IETF Chair <chair@ietf.org>
Description	DNS-SD Service Discovery
Reference	RFC 9665
Port Number	None

Field Name	Value
Service Code	None

Table 2

10.4.2. "dnssd-srp-tls" Service Name

Field Name	Value
Service Name	dnssd-srp-tls
Transport Protocol	tcp
Assignee	IESG <iesg@ietf.org>
Contact	IETF Chair <chair@ietf.org>
Description	DNS-SD Service Discovery (TLS)
Reference	RFC 9665
Port Number	None
Service Code	None

Table 3

10.5. Anycast Address

IANA has allocated an IPv6 anycast address from the "IANA IPv6 Special-Purpose Address Registry" [IPv6], similar to the Port Control Protocol [RFC6887] anycast address [RFC7723]. The purpose of this allocation is to provide a fixed anycast address that can be commonly used as a destination for SRP Updates when no SRP registrar is explicitly configured. The initial values for the registry are as follows:

Attribute	Value
Address Block	2001:1::3/128
Name	DNS-SD Service Registration Protocol Anycast Address
RFC	RFC 9665
Allocation Date	2024-04
Termination Date	N/A
Source	True

Attribute	Value
Destination	True
Forwardable	True
Globally Reachable	True
Reserved-by-Protocol	False

Table 4

11. References

11.1. Normative References

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <https://www.rfc-editor.org/info/rfc1035>.

[RFC1536] Kumar, A., Postel, J., Neuman, C., Danzig, P., and S. Miller, "Common DNS Implementation Errors and Suggested Fixes", RFC 1536, DOI 10.17487/RFC1536, October 1993, <https://www.rfc-editor.org/info/rfc1536>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2136] Vixie, P., Ed., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, DOI 10.17487/RFC2136, April 1997, <https://www.rfc-editor.org/info/rfc2136>.

[RFC2181] Elz, R. and R. Bush, "Clarifications to the DNS Specification", RFC 2181, DOI 10.17487/RFC2181, July 1997, <https://www.rfc-editor.org/info/rfc2181>.

[RFC2539] Eastlake 3rd, D., "Storage of Diffie-Hellman Keys in the Domain Name System (DNS)", RFC 2539, DOI 10.17487/RFC2539, March 1999, <https://www.rfc-editor.org/info/rfc2539>.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, DOI 10.17487/RFC2782, February 2000, <https://www.rfc-editor.org/info/rfc2782>.

[RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <https://www.rfc-editor.org/info/rfc2931>.

[RFC3172] Huston, G., Ed., "Management Guidelines & Operational Requirements for the Address and Routing Parameter Area Domain ("arpa")", BCP 52, RFC 3172, DOI 10.17487/RFC3172, September 2001, <https://www.rfc-editor.org/info/rfc3172>.

[RFC3596] Thomson, S., Huitema, C., Ksinant, V., and M. Souissi, "DNS Extensions to Support IP Version 6", STD 88, RFC 3596, DOI 10.17487/RFC3596, October 2003, <https://www.rfc-editor.org/info/rfc3596>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC6303] Andrews, M., "Locally Served DNS Zones", BCP 163, RFC 6303, DOI 10.17487/RFC6303, July 2011, <https://www.rfc-editor.org/info/rfc6303>.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <https://www.rfc-editor.org/info/rfc6763>.

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <https://www.rfc-editor.org/info/rfc8085>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8375] Pfister, P. and T. Lemon, "Special-Use Domain 'home.arpa.'", RFC 8375, DOI 10.17487/RFC8375, May 2018, <https://www.rfc-editor.org/info/rfc8375>.

[RFC8624] Wouters, P. and O. Sury, "Algorithm Implementation Requirements and Usage Guidance for DNSSEC", RFC 8624, DOI 10.17487/RFC8624, June 2019, <https://www.rfc-editor.org/info/rfc8624>.

[RFC8765] Pusateri, T. and S. Cheshire, "DNS Push Notifications", RFC 8765, DOI 10.17487/RFC8765, June 2020, <https://www.rfc-editor.org/info/rfc8765>.

[RFC9364] Hoffman, P., "DNS Security Extensions (DNSSEC)", BCP 237, RFC 9364, DOI 10.17487/RFC9364, February 2023, <https://www.rfc-editor.org/info/rfc9364>.

[RFC9664] Cheshire, S. and T. Lemon, "An EDNS(0) Option to Negotiate Leases on DNS Updates", RFC 9664, DOI 10.17487/RFC9664, October 2024, <https://www.rfc-editor.org/info/rfc9664>.

11.2. Informative References

[IAB-ARPA] "Internet Architecture Board statement on the registration of special use names in the ARPA domain", March 2017, <https://www.iab.org/documents/correspondence-reports-documents/2017-2/iab-statement-on-the-registration-of-special-use-names-in-the-arpa-domain/>.

[IPv6] IANA, "IANA IPv6 Special-Purpose Address Registry", <https://www.iana.org/assignments/iana-ipv6-special-registry>.

[LSDZ] IANA, "Locally-Served DNS Zones", <https://www.iana.org/assignments/locally-served-dns-zones>.

[PORT] IANA, "Service Name and Transport Protocol Port Number Registry", <https://www.iana.org/assignments/service-names-port-numbers>.

[RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <https://www.rfc-editor.org/info/rfc2131>.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <https://www.rfc-editor.org/info/rfc2827>.

[RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, DOI 10.17487/RFC3007, November 2000, <https://www.rfc-editor.org/info/rfc3007>.

[RFC3927] Cheshire, S., Aboba, B., and E. Guttman, "Dynamic Configuration of IPv4 Link-Local Addresses", RFC 3927, DOI 10.17487/RFC3927, May 2005, <https://www.rfc-editor.org/info/rfc3927>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <https://www.rfc-editor.org/info/rfc4862>.

[RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, DOI 10.17487/RFC6105, February 2011, <https://www.rfc-editor.org/info/rfc6105>.

[RFC6335] Cotton, M., Eggert, L., Touch, J., Westerlund, M., and S. Cheshire, "Internet Assigned Numbers Authority (IANA) Procedures for the Management of the Service Name and Transport Protocol Port Number Registry", BCP 165, RFC 6335, DOI 10.17487/RFC6335, August 2011, <https://www.rfc-editor.org/info/rfc6335>.

[RFC6760] Cheshire, S. and M. Krochmal, "Requirements for a Protocol to Replace the AppleTalk Name Binding Protocol (NBP)", RFC 6760, DOI 10.17487/RFC6760, February 2013, <https://www.rfc-editor.org/info/rfc6760>.

[RFC6761] Cheshire, S. and M. Krochmal, "Special-Use Domain Names", RFC 6761, DOI 10.17487/RFC6761, February 2013, <https://www.rfc-editor.org/info/rfc6761>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>.

[RFC6887] Wing, D., Ed., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)", RFC 6887, DOI 10.17487/RFC6887, April 2013, <https://www.rfc-editor.org/info/rfc6887>.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <https://www.rfc-editor.org/info/rfc7084>.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <https://www.rfc-editor.org/info/rfc7228>.

[RFC7413] Cheng, Y., Chu, J., Radhakrishnan, S., and A. Jain, "TCP Fast Open", RFC 7413, DOI 10.17487/RFC7413, December 2014, <https://www.rfc-editor.org/info/rfc7413>.

[RFC7723] Kiesel, S. and R. Penno, "Port Control Protocol (PCP) Anycast Addresses", RFC 7723, DOI 10.17487/RFC7723, January 2016, <https://www.rfc-editor.org/info/rfc7723>.

[RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 8415, DOI 10.17487/RFC8415, November 2018, <https://www.rfc-editor.org/info/rfc8415>.

[RFC8520] Lear, E., Droms, R., and D. Romascanu, "Manufacturer Usage Description Specification", RFC 8520, DOI 10.17487/RFC8520, March 2019, <https://www.rfc-editor.org/info/rfc8520>.

[RFC8766] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", RFC 8766, DOI 10.17487/RFC8766, June 2020, <https://www.rfc-editor.org/info/rfc8766>.

[RFC8945] Dupont, F., Morris, S., Vixie, P., Eastlake 3rd, D., Gudmundsson, O., and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", STD 93, RFC 8945, DOI 10.17487/RFC8945, November 2020, <https://www.rfc-editor.org/info/rfc8945>.

[ROADMAP] Cheshire, S., "Service Discovery Road Map", Work in Progress, Internet-Draft, draft-cheshire-dnssd-roadmap-03, 23 October 2018, <https://datatracker.ietf.org/doc/html/draft-cheshire-dnssd-roadmap-03>.

[SNAC-SIMPLE] Lemon, T. and J. Hui, "Automatically Connecting Stub Networks to Unmanaged Infrastructure", Work in Progress, Internet-Draft, draft-ietf-snac-simple-06, 4 November 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-snac-simple-06>.

[SUB] IANA, "service.arpa Subdomain", <https://www.iana.org/assignments/locally-served-dns-zones/locally-served-dns-zones>.

[SUDN] IANA, "Special-Use Domain Names", <https://www.iana.org/assignments/special-use-domain-names>.

[ZC] Steinberg, D.H. and S. Cheshire, "Zero Configuration Networking: The Definitive Guide", O'Reilly Media, Inc., ISBN 9780596101008, December 2005.

Appendix A. Using Standard Authoritative DNS Servers Compliant with RFC 2136 to Test SRP Requesters

For testing, it may be useful to set up an authoritative DNS server that does not implement SRP. This can be done by configuring the authoritative DNS server to listen on the anycast address or by advertising it in the "_dnssd-srp._tcp.<zone>" and "_dnssd-srp-tls._tcp.<zone>" SRV records. It must be configured to be authoritative for "default.service.arpa." and to accept updates from hosts on local networks for names under "default.service.arpa." without authentication since such authoritative DNS servers will not have support for FCFS authentication (Section 3.2.4.1).

An authoritative DNS server configured in this way will be able to successfully accept and process SRP Updates from requesters that send SRP updates. However, no prerequisites will be applied; this means that the test authoritative DNS server will accept internally inconsistent SRP Updates and will not stop two SRP Updates sent by different services that claim the same name or names from overwriting each other.

Since SRP Updates are signed with keys, validation of the SIG(0) algorithm used by the requester can be done by manually installing the requester's public key on the authoritative DNS server that will be receiving the updates. The key can then be used to authenticate the SRP Update and can be used as a requirement for the update. An example configuration for testing SRP using BIND 9 is given in Appendix C.

Appendix B. How to Allow SRP Requesters to Update Standard Servers Compliant with RFC 2136

Ordinarily, CNN SRP Updates sent to an authoritative DNS server that implements standard DNS Update [RFC2136] but not SRP will fail because the zone being updated is "default.service.arpa." and because no authoritative DNS server that is not an SRP registrar would normally be configured to be authoritative for "default.service.arpa.". Therefore, a requester that sends an SRP Update can tell that the receiving authoritative DNS server does not support SRP but does support standard DNS Update [RFC2136] because the RCODE will either be NotZone, NotAuth, or Refused or because there is no response to the update request (when using the anycast address).

In this case, a requester **MAY** attempt to register itself using normal DNS updates [RFC2136]. To do so, it must discover the default registration zone and the authoritative DNS server designated to receive updates for that zone, as described earlier, using the _dns-update._udp SRV record. It can then send the update to the port and host pointed to by the SRV record, and it is expected to use appropriate prerequisites to avoid overwriting competing records. Such updates are out of scope for SRP, and a requester that implements SRP **MUST** first attempt to use SRP to register itself and only attempt to use backwards capability with normal DNS Update [RFC2136] if that fails. Although the owner name of the SRV record for DNS Update (_dns-update._udp) specifies UDP, it is also possible to use TCP, and TCP **SHOULD** be required to prevent spoofing.

Appendix C. Sample BIND 9 Configuration for "default.service.arpa."

```
zone "default.service.arpa." {
  type primary;
  file "/etc/bind/primary/service.db";
  allow-update { key demo.default.service.arpa.; };
};
```

Figure 1: Zone Configuration in named.conf

```
$TTL 57600  ; 16 hours
@                   IN SOA       ns postmaster (
                      2951053287 ; serial
                      3600       ; refresh (1 hour)
                      1800       ; retry   (30 minutes)
                      604800     ; expire  (1 week)
                      3600       ; minimum (1 hour)
                      )
                    NS           ns
ns                  AAAA         2001:db8:0:2::1

$TTL 3600   ; 1 hour

; Autoconguration bootstrap records
_dnssd-srp._tcp     SRV 0 0 53   ns
_dnssd-srp-tls._tcp SRV 0 0 853  ns

; Service Discovery Instruction
_ipps._tcp          PTR          demo._ipps._tcp

; Service Description Instruction
demo._ipps._tcp     SRV 0 0 631  demohost
                    TXT          ""

; Host Description Instruction
demohost            AAAA         2001:db8:0:2::2
                    KEY 0 3 13 (
                      qweEmaaq0FAWok5//ftuQtZgiZoiFSUsm0srWREdywQU
                      9dpvtOhrdKWUuPT3uEFF5TZU6B4q1z1I662GdaUwqg==
                      ); alg = ECDSAP256SHA256 ; key id = 14495
```

Figure 2: Example Zone File

Acknowledgments

Thanks to Toke Høiland-Jørgensen, Jonathan Hui, Esko Dijk, Kangping Dong, and Abtin Keshavarzian for their thorough technical reviews. Thanks to Kangping and Abtin as well for testing the document by doing an independent implementation. Thanks to Tamara Kemper for doing a nice developmental edit, Tim Wattenberg for doing an SRP requester proof-of-concept implementation at the Montreal Hackathon at IETF 102, and Tom Pusateri for reviewing during the hackathon and afterwards. Thanks to Esko for a really thorough second Last Call review. Thanks also to Nathan Dyck, Gabriel Montenegro, Kangping Dong, Martin Turon, and Michael Cowan for their detailed second last call reviews. Thanks to Patrik Fältström, Dhruv Dhody, David Dong, Joey Salazar, Jean-Michel Combes, and Joerg Ott for their respective directorate reviews. Thanks to Paul Wouters for a *really* detailed IESG review! Thanks also to the other IESG members who provided comments or simply took the time to review the document.

Authors' Addresses

Ted Lemon
Apple Inc.
One Apple Park Way
Cupertino, CA 95014
United States of America
Email: mellon@fugue.com

Stuart Cheshire
Apple Inc.
One Apple Park Way
Cupertino, CA 95014
United States of America
Phone: +1 408 974 3207
Email: cheshire@apple.com