Internet Engineering Task Force T. Lemon
Internet-Draft S. Cheshire
Intended status: Informational Apple Inc.
Expires: January 14, 2021 July 13, 2020

Service Registration Protocol for DNS-Based Service Discovery
draft-ietf-dnssd-srp-04

Abstract

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

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

DNS-Based Service Discovery [RFC6763] is a component of Zero Configuration Networking [RFC6760] [ZC] [I-D.cheshire-dnssd-roadmap].

This document describes an enhancement to DNS-Based Service Discovery [RFC6763] that allows services to automatically register their services using the DNS protocol rather than using Multicast DNS [RFC6762] (mDNS). There is already a large installed base of DNS-SD clients that can discover services using the DNS protocol. This extension makes it much easier to take advantage of this existing functionality.

This document is intended for three audiences: implementors of software that provides services that should be advertised using DNS-SD, implementors of 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 intended to provide sufficient information to allow interoperable implementation of the registration protocol.

DNS-Based Service Discovery (DNS-SD) allows services to advertise the fact that they provide service, and to provide the information required to access that service. 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.

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

The DNS-SD specification [RFC6763], Section 10 ("Populating the DNS with Information"), briefly discusses ways that services can publish their information in the DNS namespace. In the case of mDNS, it allows services to publish their information on the local link, using names in the ".local" namespace, which makes their services directly discoverable by peers attached to that same local link.

RFC6763 also allows clients to discover services using the DNS protocol [RFC1035]. This can be done by having a system administrator manually configure service information in the DNS, but manually populating DNS authoritative server databases is costly and potentially error-prone, and requires a knowledgable 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 [I-D.ietf-dnssd-hybrid] provides one way to automatically populate the DNS namespace, but is only appropriate on networks where services are easily advertised using mDNS. This document describes a solution more suitable for networks where multicast is inefficient, or where sleepy devices are common, by supporting both offering of services, and discovery of services, using unicast.

2. Service Registration Protocol

Services that implement SRP use DNS Update [RFC2136] [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" [RFC7228].

Full-featured hosts are either configured manually with a registration domain, or use the "dr._dns-sd._udp.<domain>" query ([RFC6763] Section 11) to learn the default registration domain from the network. RFC6763 says to discover the registration domain using either ".local" or a network-supplied domain name for <domain>. Services using SRP MUST use the domain name received through the DHCPv4 Domain Name option ([RFC2132] section 3.17), if available, or the Neighbor Discovery DNS Search List option [RFC8106]. If the DNS Search List option contains more than one domain name, it MUST NOT be used. If neither option is available, the Service Registration protocol is not available on the local network.

Manual configuration of the registraton domain can be done either by querying the list of available registration zones ("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(s) 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 [I-D.ietf-dnssd-push]. Having discovered the enclosing DNS zone, they query for the "_dnssd-srp._tcp<zone>" SRV record to discover the server to which they should send DNS updates. Hosts that support SRP updates using TLS use the "_dnssd-srp-tls._tcp<zone>" SRV record instead.

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

SRP clients on constrained networks are expected to receive from the network a list of SRP servers with which to register. It is the responsibility of a Constrained-Node Network supporting SRP to provide one or more SRP server addresses. It is the responsibility of the SRP server supporting a Constrained-Node Network 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 internally to names with broader visibility.

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

Networks that are not constrained networks can more complicated topologies at the Internet layer. Nodes connected to such networks can be assumed to be able to do DNSSD service registration domain discovery. Such networks are generally able to provide registration domain discovery and routing. By requiring the use of TCP, the possibility of off-network spoofing is eliminated.

We will discuss several parts to this process: how to know what to publish, how to know where to publish it (under what name), how to publish it, how to secure its publication, and how to maintain the information once published.

2.1. What to publish

We refer to the DNS Update message sent by services using SRP as an SRP update. Three types of updates appear in an SRP update: Service Discovery records, Service Description records, and Host Description records.

- Service Discovery records are one or more PTR RRs, mapping from the generic service type (or subtype) to the specific Service Instance Name.

- Service Description records are exactly one SRV RR, exactly one KEY RR, and one or more TXT RRs, all with the same name, the Service Instance Name ([RFC6763] section 4.1). In principle

Service Description records can include other record types, with the same Service Instance Name, though in practice they rarely do. The Service Instance Name MUST be referenced by one or more Service Discovery PTR records, unless it is a placeholder service registration for an intentionally non-discoverable service name.

- The Host Description records for a service are a KEY RR, used to claim exclusive ownership of the service registration, and one or more RRs of type A or AAAA, giving the IPv4 or IPv6 address(es) of the host where the service resides.

RFC 6763 describes the details of what each of these types of updates contains and is the definitive source 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 first-come, first-serve naming, which we describe later on in this document.

2.2. Where to publish it

Multicast DNS uses a single namespace, ".local", which is valid on the local link. This convenience is not available for DNS-SD using the DNS protocol: services must exist in some specific unicast namespace.

As described above, full-featured devices are responsible for knowing in what domain they should register their services. Devices made for Constrained-Node Networks register in the (proposed) special use domain name [RFC6761] "default.service.arpa", and let the SRP server handle rewriting that to a different domain if necessary.

2.3. How to publish it

It is possible to issue a DNS Update that does several things at once; this means that it's possible to do all the work of adding a PTR resource record to the PTR RRset on the Service Name, and creating or updating the Service Instance Name and Host Description, in a single transaction.

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 than regular DNS Updates as defined in RFC2136. The RFC2136 update process can 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 registration protocol 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 2.4.3 are implicit in the processing of SRP updates, and so there is no need for the service sending the SRP update to put in any explicit prerequisites.

2.3.1. How DNS-SD Service Registration differs from standard RFC2136 DNS Update

DNS-SD Service Registration is based on standard RFC2136 DNS Update, with some differences:

- It implements first-come first-served name allocation, 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 server is not required to implement general DNS Update prerequsite processing.

- Clients are allowed to send updates to the generic domain "default.service.arpa"

2.4. How to secure it

Traditional DNS update is secured using the TSIG protocol, which uses a secret key shared between the client (which issues the update) and the 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 what we describe here is an improvement over the security of mDNS. The goal is not to provide the level of security of a network managed by a skilled operator.

2.4.1. First-Come First-Served Naming

First-Come First-Serve naming provides a limited degree of security: a service that registers its service using DNS-SD Registration protocol is given ownership of a name for an extended period of time based on the key used to authenticate the DNS Update. As long as the registration service remembers the name and the key used to register that name, no other service can add or update the information associated with that. FCFS naming is used to protect both the Service Description and the Host Description.

2.4.1.1. Service Behavior

The service generates a public/private key pair. This key pair MUST be stored in stable storage; if there is no writable stable storage on the client, the client MUST be pre-configured with a public/private key pair in read-only storage that can be used. This key pair MUST be unique to the device.

When sending DNS updates, the service includes a KEY record containing the public portion of the key in each Host Description update and each Service Description update. 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 is set using the EDNS(0) Update Lease option [I-D.sekar-dns-ul].

The KEY record in Service Description updates MAY be omitted for brevity; if it is omitted, the SRP server 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.

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. This means that if a device is disconnected from the network, it does not appear in the user interfaces of devices looking for services of that type for too long.

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 of this is 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.

This means that 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 re-use.

2.4.2. Removing published services

To remove a service registration, the client retransmits its most recent update with an Update Lease option that has a LEASE value of zero. If the registration is to be permanently removed, KEY-LEASE should also be zero. Otherwise, it should have the same value it had previously; this holds the name in reserve for when the client is once again able to provide the service.

SRP clients are normally expected to remove all service instances when removing a host. However, in some cases a client may not have retained sufficient state to know that some service instance is pointing to a host that it is removing. Nevertheless, removing the host can be assumed to mean that all service instances pointing to it are no longer valid. Therefore, SRP servers MAY remove all service instances pointing to a host when a host is removed, even if the client doesn't remove them explicitly.

2.4.3. SRP Server Behavior

2.4.3.1. Validation of Adds

The SRP server first validates that the DNS Update is a syntactically and semantically valid DNS Update according to the rules specified in RFC2136.

SRP Updates consist of a set of Instructions that together add one or more services. Each instruction consists either of a single add, or a delete followed by an add. When an instruction contains a delete and an add, the delete MUST precede the add.

The SRP server checks each Instruction in the SRP update to see that it is either a Service Discovery update, a Service Description update, or a Host Description update. 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 question 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.

An Instruction is a Service Discovery Instruction if it contains

- exactly one "Add to an RRSet" ([RFC2136] Section 2.5.1) RR,
- which is a PTR RR,
- which points to a Service Instance Name
- for which a Service Description Instruction is present in the SRP Update.
- Service Discovery Instructions do not contain any deletes, and do not contain any other adds.

An Instruction is a Service Description Instruction if, for the appropriate Service Instance Name, it contains

- exactly one "Delete all RRsets from a name" update for the service instance name [RFC2136] Section 2.5.3,
- exactly one "Add to an RRset" SRV RR,
- zero or one "Add to an RRset" KEY RR that contains the public key corresponding to the private key that was used to sign the message (if present, the KEY MUST match the KEY RR given in the Host Description),
- one or more "Add to an RRset" TXT RRs,
- and the target of the SRV RR Add points to a hostname for which there is a Host Description Instruction in the SRP Update.
- Service Descriptions Instructions do not modify any other RRs.

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

- exactly one "Delete all RRsets from a name" RR,
- one or more "Add to an RRset" RRs of type A and/or AAAA,
- 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,
- there is a Service Instance Name Instruction in the SRP update for which the SRV RR that is added points to the hostname being updated by this update.
- Host Description updates do not modify any other records.

An SRP Update MUST include at least one Service Discovery Instruction, at least one Service Description Instruction, and exactly one Host Description Instruction. A DNS Update that does not is not an SRP update. A DNS Update that contains any other adds, any other deletes, or any prerequisites, is not an SRP update. Such messages should either be processed as regular RFC2136 updates, including access control checks and constraint checks, if supported, or else rejected with RCODE=REFUSED.

Note that if the definitions of each of these update types are followed carefully, this means that many things that look very much like SRP updates nevertheless are not. For example, a DNS update that contains an RRset Add to a Service Name and an RRset Add to a Service Instance Name, where the Service Name does not reference the Service Instance Name, is not a valid SRP update message, but may be a valid RFC2136 update.

Assuming that a DNS Update message has been validated with these conditions and is a valid SRP Update, the server checks that the name in the Host Description Instruction exists. If so, then the server checks to see if the KEY record on that name is the same as the KEY record in the Host Description Instruction. The server performs the same check for the KEY records in any Service Description Instrructions. 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 same record in the SRP Update (whether provided or taken from the Host Description Instruction), then the server MUST reject the SRP Update with the YXDOMAIN RCODE.

Otherwise, the server validates the SRP Update using SIG(0) on the public key in the KEY record of the Host Description update. If the validation fails, the server MUST reject the SRP Update with the REFUSED RCODE. Otherwise, the SRP update is considered valid and authentic, and is processed according to the method described in RFC2136.

KEY record updates omitted from Service Description update are processed as if they had been explicitly present: every Service Description that is updated MUST, after the update, have a KEY RR, and it must be the same KEY RR that is present in the Host Description to which the Service Description refers.

The status that is returned depends on the result of processing the update, and can be either SUCCESS or SERVFAIL: all other possible outcomes should already have been accounted for when applying the constraints that qualify the update as an SRP Update.

The server MAY add a Reverse Mapping that corresponds to the Host Description. This is not required because the Reverse Mapping serves no protocol function, but it may be useful for debugging, e.g. in annotating network packet traces or logs. In order for the server to add a reverse mapping update, it must be authoritative for the zone or have credentials to do the update. The client MAY also do a reverse mapping update if it has credentials to do so.

The server 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 pre-registered keys. In this case, an update for a key that has not been registered should be rejected with the REFUSED RCODE.

There are at least two benefits to doing this rather than simply using normal SIG(0) DNS updates. First, the same registration protocol can be used in both cases, so both use cases can be addressed by the same service implementation. Second, the registration protocol includes maintenance functionality not present with normal DNS updates.

Note that the semantics of using SRP in this way are different than for typical RFC2136 implementations: the KEY used to sign the SRP update only allows the client to update records that refer to its Host Description. RFC2136 implementations do not normally provide a way to enforce a constraint of this type.

The server may 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 YXDOMAIN.

2.5. TTL Consistency

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

Services sending SRP updates MUST use consistent TTLs in all RRs within the SRP update.

SRP update servers MUST check that the TTLs for all RRs within the 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 server 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 client's guess as to what a good TTL would be. SRP servers may override these TTLs. SRP servers SHOULD ensure that TTLs are reasonable: neither too long nor too short. The TTL should never be longer than the lease time Section 2.6.1. Shorter TTLs will result in more frequent

data refreshes; this increases latency on the client side, and increases load on any caching resolvers and on the authoritative server. 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 server.

2.6. Maintenance

2.6.1. Cleaning up stale data

Because the DNS-SD registration protocol is automatic, and not managed by humans, some additional bookkeeping is required. When an update is constructed by the client, it MUST include include an EDNS(0) Update Lease Option [I-D.sekar-dns-ul]. The Update Lease Option contains two lease times: the Lease Time and the Key Lease Time.

These leases are promises, similar to DHCP leases [RFC2131], from the client that it will send a new update for the service registration before the lease time expires. The Lease time is chosen to represent the time after the update during which the registered records other than the KEY record should be assumed to be valid. The Key Lease time represents the time after the update during which the KEY record should be assumed to be valid.

The reasoning behind the different lease times is discussed in the section on first-come, first-served naming Section 2.4.1. SRP servers may be configured with limits for these values. A default limit of two hours for the Lease and 14 days for the SIG(0) KEY are currently thought to be good choices. Clients that are going to continue to use names on which they hold leases should update well before the lease ends, in case the registration service is unavailable or under heavy load.

The SRP server MUST include an EDNS(0) Update Lease option in the response if the lease time proposed by the service has been shortened or lengthened. The service MUST check for the EDNS(0) Update Lease option in the response and MUST use the lease times from that option in place of the options that it sent to the server when deciding when to update its registration. The times may be shorter or longer than those specified in the SRP update; the client must honor them in either case.

Clients should assume that each lease ends N seconds after the update was first transmitted, where N is the lease duration. Servers should assume that each lease ends N seconds after the update that was successfully processed was received. Because the server will always

receive the update after the client sent it, this avoids the possibility of misunderstandings.

SRP servers MUST reject updates that do not include an EDNS(0) Update Lease option. Dual-use servers MAY accept updates that don't include leases, but SHOULD differentiate between SRP updates and other updates, and MUST reject updates that would otherwise be SRP updates updates if they do not include leases.

Lease times have a completely different function than TTLs. 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 'n' when the remaining time on the lease is less than 'n'.

2.6.2. Sleep Proxy

Another use of SRP is for devices that sleep to reduce power consumption.

In this case, in addition to the DNS Update Lease option [I-D.sekar-dns-ul] described above, the device includes an EDNS(0) OWNER Option [I-D.cheshire-edns0-owner-option].

The EDNS(0) Update Lease option constitutes a promise by the device that it will wake up before this time elapses, to renew its registration and thereby demonstrate that it is still attached to the network. If it fails to renew the registration by this time, that indicates that it is no longer attached to the network, and its registration (except for the KEY in the Host Description) should be deleted.

The EDNS(0) OWNER Option indicates that the device will be asleep, and will not be receptive to normal network traffic. When a DNS server receives a DNS Update with an EDNS(0) OWNER Option, that signifies that the SRP server should set up a proxy for any IPv4 or IPv6 address records in the DNS Update message. This proxy should send ARP or ND messages claiming ownership of the IPv4 and/or IPv6 addresses in the records in question. In addition, proxy should answer future ARP or ND requests for those IPv4 and/or IPv6 addresses, claiming ownership of them. When the DNS server receives a TCP SYN or UDP packet addressed to one of the IPv4 or IPv6 addresses for which it proxying, it should then wake up the sleeping device using the information in the EDNS(0) OWNER Option. At present version 0 of the OWNER Option specifies the "Wake-on-LAN Magic

Packet" that needs to be sent; future versions could be extended to specify other wakeup mechanisms.

Note that although the authoritative DNS server that implements the SRP function need not be on the same link as the sleeping host, the Sleep Proxy must be on the same link.

It is not required that sleepy nodes on a Constrained-Node Network support sleep proxy. Such devices may have different mechanisms for dealing with sleep and wakeup. An SRP registration for such a device will be useful regardless of the mechanism whereby messages are delivered to the sleepy end device. For example, the message might be held in a buffer for an extended period of time by an intermediate device on a mesh network, and then delivered to the device when it wakes up. The exact details of such behaviors are out of scope for this document.

3. Security Considerations

3.1. Source Validation

SRP updates have no authorization semantics other than first-come, first-served. This means that if an attacker from outside of the administrative domain of the server knows the server's IP address, it can in principle send updates to the server that will be processed successfully. Servers should therefore be configured to reject updates from source addresses outside of the administrative domain of the server.

For Anycast updates, this validation must be enforced by every router that connects the Constrained-Device Network to the unconstrained portion of the network. For TCP updates, the initial SYN-SYN+ACK handshake prevents updates being forged by an off-network attacker. In order to ensure that this handshake happens, Service Discovery Protocol servers MUST NOT accept TCP Fast Open payloads.

Note that these rules only apply to the validation of SRP updates. A server that accepts updates from DNS-SD registration protocol clients may also accept other DNS updates, and those DNS updates may be validated using different rules. However, in the case of a DNS service 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 RFC2136 update to any RR that was added using SRP, but that is authenticated using a different key, could be used to override a promise made by the registration protocol, by replacing all or part of the service registration

information with information provided by a different client. An implementation that allows both kinds of updates should not allow updates to records added by SRP updates using different authentication and authorization credentials.

3.2. SIG(0) signature validation

This specification does not provide a mechanism for validating responses from DNS servers to SRP clients. In the case of Constrained Network/Constrained Node clients, such validation isn't practical because there's no way to establish trust. In principle, a KEY RR could be used by a non-constrained SRP client to validate responses from the server, but this is not required, nor do we specify a mechanism for determining which key to use.

3.3. Required Signature Algorithm

For validation, SRP Servers MUST implement the ECDSAP256SHA256 signature algorithm. SRP servers SHOULD implement the algorithms specified in [I-D.ietf-dnsop-algorithm-update] section 3.1, in the validation column of the table, starting with algorithm number 13. SRP clients MUST NOT assume that any algorithm numbered lower than 13 is available for use in validating SIG(0) signatures.

4. Privacy Considerations

Because DNSSD SRP updates can be sent off-link, the privacy implications of SRP are different than for multicast DNS responses. Host implementations that are using TCP SHOULD also use TLS if available. Server implementations MUST offer TLS support. The use of TLS with DNS is described in [RFC7858] and [RFC8310].

Hosts that implement TLS support SHOULD NOT fall back to TCP; since servers are required to support TLS, it is entirely up to the host implementation whether to use it.

5. Delegation of 'service.arpa.'

In order to be fully functional, there must be a delegation of 'service.arpa.' in the '.arpa.' zone [RFC3172]. This delegation should be set up as was done for 'home.arpa', as a result of the specification in [RFC8375]Section 7.

6. IANA Considerations

6.1. Registration and Delegation of 'service.arpa' as a Special-Use Domain Name

IANA is requested to record the domain name 'service.arpa.' in the Special-Use Domain Names registry [SUDN]. IANA is requested, with the approval of IAB, to implement the delegation requested in Section 5.

IANA is further requested to add a new entry to the "Transport-Independent Locally-Served Zones" subregistry of the the "Locally-Served DNS Zones" registry[LSDZ]. The entry will be for the domain 'service.arpa.' with the description "DNS-SD Registration Protocol Special-Use Domain", listing this document as the reference.

6.2. 'dnssd-srp' Service Name

IANA is also requested to add a new entry to the Service Names and Port Numbers registry for dnssd-srp with a transport type of tcp. No port number is to be assigned. The reference should be to this document, and the Assignee and Contact information should reference the authors of this document. The Description should be as follows:

Availability of DNS Service Discovery Service Registration Protocol Service for a given domain is advertised using the "_dnssd-srp._tcp.<domain>." SRV record gives the target host and port where DNSSD Service Registration Service is provided for the named domain.

6.3. 'dnssd-srp-tls' Service Name

IANA is also requested to add a new entry to the Service Names and Port Numbers registry for dnssd-srp with a transport type of tcp. No port number is to be assigned. The reference should be to this document, and the Assignee and Contact information should reference the authors of this document. The Description should be as follows:

Availability of DNS Service Discovery Service Registration Protocol Service for a given domain over TLS is advertised using the "_dnssd-srp-tls._tcp.<domain>." SRV record gives the target host and port where DNSSD Service Registration Service is provided for the named domain.

6.4. Anycast Address

IANA is requested to allocate an IPv6 Anycast address from the IPv6 Special-Purpose Address Registry, similar to the Port Control Protocol anycast address, 2001:1::1. This address is referred to

within the document as TBD1, and the document should be updated to reflect the address that was allocated.

7. Acknowledgments

Thanks to Toke Hoeiland-Joergensen for a thorough technical review, to Tamara Kemper for doing a nice developmental edit, Tim Wattenberg for doing a service implementation at the Montreal Hackathon at IETF 102, Tom Pusateri for reviewing during the hackathon and afterwards, and [...] more reviewers to come, hopefully.

8. References

8.1. Normative References

[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>.

[I-D.sekar-dns-ul]
Cheshire, S. and T. Lemon, "Dynamic DNS Update Leases", draft-sekar-dns-ul-02 (work in progress), August 2018.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, DOI 10.17487/RFC2132, March 1997, <https://www.rfc-editor.org/info/rfc2132>.

[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>.

[RFC8106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 8106, DOI 10.17487/RFC8106, March 2017, <https://www.rfc-editor.org/info/rfc8106>.

[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>.

[I-D.ietf-dnsop-algorithm-update]
Wouters, P. and O. Sury, "Algorithm Implementation Requirements and Usage Guidance for DNSSEC", draft-ietf-dnsop-algorithm-update-10 (work in progress), April 2019.

[SUDN] "Special-Use Domain Names Registry", July 2012, <https://www.iana.org/assignments/special-use-domain-names/special-use-domain-names.xhtml>.

[LSDZ] "Locally-Served DNS Zones Registry", July 2011, <https://www.iana.org/assignments/locally-served-dns-zones/locally-served-dns-zones.xhtml>.

8.2. Informative 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>.

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

[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>.

[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>.

[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>.

[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>.

[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>.

[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>.

[RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <https://www.rfc-editor.org/info/rfc8310>.

[I-D.ietf-dnssd-hybrid] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", draft-ietf-dnssd-hybrid-10 (work in progress), March 2019.

[I-D.ietf-dnssd-push] Pusateri, T. and S. Cheshire, "DNS Push Notifications", draft-ietf-dnssd-push-25 (work in progress), October 2019.

[I-D.cheshire-dnssd-roadmap] Cheshire, S., "Service Discovery Road Map", draft-cheshire-dnssd-roadmap-03 (work in progress), October 2018.

[I-D.cheshire-edns0-owner-option] Cheshire, S. and M. Krochmal, "EDNS0 OWNER Option", draft-cheshire-edns0-owner-option-01 (work in progress), July 2017.

[ZC] Cheshire, S. and D. Steinberg, "Zero Configuration Networking: The Definitive Guide", O'Reilly Media, Inc. , ISBN 0-596-10100-7, December 2005.

Appendix A. Testing using standard RFC2136-compliant servers

It may be useful to set up a DNS server for testing that does not implement SRP. This can be done by configuring the server to listen on the anycast address, or advertising it in the _dnssd-srp._tcp.<zone> SRV and _dnssd-srp-tls._tcp.<zone> record. 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 servers will not have support for FCFS authentication Section 2.4.1.

A server configured in this way will be able to successfully accept and process SRP updates from services that send SRP updates. However, no prerequisites will be applied, and this means that the test server will accept internally inconsistent SRP updates, and will not stop two SRP updates, sent by different services, that claim the same name(s), from overwriting each other.

Since SRP updates are signed with keys, validation of the SIG(0) algorithm used by the client can be done by manually installing the client public key on the DNS server that will be receiving the updates. The key can then be used to authenticate the client, 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 services to update standard RFC2136-compliant servers

Ordinarily SRP updates will fail when sent to an RFC 2136-compliant server that does not implement SRP because the zone being updated is "default.service.arpa", and no DNS server that is not an SRP server should normally be configured to be authoritative for "default.service.arpa". Therefore, a service that sends an SRP update can tell that the receiving server does not support SRP, but does support 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 service MAY attempt to register itself using regular RFC2136 DNS updates. To do so, it must discover the default registration zone and the DNS server designated to receive updates for that zone, as described earlier, using the _dns-update._udp SRV record. It can then make the update using the port and host pointed to by the SRV record, and should use appropriate prerequisites to avoid overwriting competing records. Such updates are out of scope for SRP, and a service that implements SRP MUST first attempt to use SRP to register itself, and should only attempt to use RFC2136 backwards compatibility if that fails. Although the owner name for the SRV record specifies the UDP protocol for updates, it is also possible to use TCP, and TCP should be required to prevent spoofing.

Appendix C. Sample BIND9 configuration for default.service.arpa.

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

Zone Configuration in named.conf

```
$ORIGIN .
$TTL 57600  ; 16 hours
default.service.arpa IN SOA          ns3.default.service.arpa.
                            postmaster.default.service.arpa. (
                2951053287 ; serial
                3600       ; refresh (1 hour)
                1800       ; retry (30 minutes)
                604800     ; expire (1 week)
                3600       ; minimum (1 hour)
)
                        NS           ns3.default.service.arpa.
                        SRV 0 0 53   ns3.default.service.arpa.
$ORIGIN default.service.arpa.
$TTL 3600   ; 1 hour
_ipps._tcp              PTR          demo._ipps._tcp
$ORIGIN _ipps._tcp.default.service.arpa.
demo                    TXT          "0"
                        SRV 0 0 9992 demo.default.service.arpa.
$ORIGIN _udp.default.service.arpa.
$TTL 3600   ; 1 hour
_dns-update             PTR          ns3.default.service.arpa.
$ORIGIN _tcp.default.service.arpa.
_dnssd-srp              PTR          ns3.default.service.arpa.
$ORIGIN default.service.arpa.
$TTL 300    ; 5 minutes
ns3                     AAAA         2001:db8:0:1::1
$TTL 3600   ; 1 hour
demo                    AAAA         2001:db8:0:2::1
                        KEY 513 3 13 (
                           qweEmaaq0FAWok5//ftuQtZgiZoiFSUsm0srWREdywQU
                           9dpvtOhrdKWUuPT3uEFF5TZU6B4q1z1I662GdaUwqg==
                        ); alg = ECDSAP256SHA256 ; key id = 15008
                        AAAA    ::1
```

Example Zone file

Authors' Addresses

Ted Lemon
Apple Inc.
One Apple Park Way
Cupertino, California 95014
USA

Email: mellon@fugue.com

Stuart Cheshire
Apple Inc.
One Apple Park Way
Cupertino, California 95014
USA

Phone: +1 408 974 3207
Email: cheshire@apple.com