Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: January 3, 2018

S. Cheshire
Apple Inc.
T. Lemon
Nominum, Inc.

July 2, 2017

Service Registration Protocol for DNS-Based Service Discovery
draft-sctl-service-registration-00

Abstract

The DNS-SD Service Registration Protocol provides a way to perform DNS-Based Service Discovery using only unicast packets. This eliminates the dependency on Multicast DNS as the foundation layer, which has worked well in some environments, like the simplest of home networks, but not in others, like large enterprise networks (where multicast does not scale well to thousands of devices) and mesh networks (where multicast and broadcast are supported poorly, if at all). Broadly speaking, the DNS-SD Service Registration Protocol is DNS Update, with a few additions.

Status of This Memo

Copyright Notice

1. Introduction

DNS-Based Service Discovery [RFC6763] is a component of Zero Configuration Networking [RFC6760] [ZC] [Roadmap].

There are two facets of DNS-Based Service Discovery to consider: how relevant information makes its way into the DNS namespace (how a server offers its services to interested clients) and how clients access that information (how an interested client discovers and uses a service instance).

This document is concerned with the first of those two facets: how relevant information makes its way into the DNS namespace.

In the DNS-Based Service Discovery specification [RFC6763] Section 10 "Populating the DNS with Information" briefly discusses ways that relevant information can make its way into the DNS namespace. In the case of Multicast DNS [RFC6762], the relevant information trivially becomes visible in the ".local" namespace by virtue of devices answering for themselves. For unicast DNS names, ways that information makes its way into the DNS namespace include manual configuration of DNS zone files, possibly assisted using tools such as the "dns-sd -Z" command, automated tools such as a Discovery Proxy [DisProx], or explicit registration by the services themselves. It is the last option -- explicit registration by the services themselves -- that is the subject of this document.

2. Service Registration Protocol

The DNS-SD Service Registration Protocol is largely built on DNS Update [RFC2136] [RFC3007], with some additions.

When a device advertises services using Multicast DNS, the parent domain is implicitly ".local".

When a device advertises services in the traditional unicast DNS namespace, it needs to know the parent domain name for its services. This parent domain can be manually configured by a human operator, or learned from the network. In the DNS-SD specification [RFC6763] section 11, "Discovery of Browsing and Registration Domains (Domain Enumeration)", describes how a client device can learn a recommended default registration domain from the network.

In the remainder of this document, Section 3 covers cleanup of stale data, and Section 4 covers advertising services on behalf of devices that are sleeping to reduce power consumption.

The final question is security. Most dynamic DNS servers will not accept unauthenticated updates. In the case of manual configuration of registration domain by a human operator, the human operator can also configure an appropriate TSIG security key. In the case of automatic configuration via DNS-SD Domain Enumeration queries, it would be nice to also have zero-configuration security. While at first glance zero-configuration security may seem to be a self-contradiction, this document proposes a simple first-come first-served security mechanism, described below in Section 5.

3. Cleanup of Stale Data

The traditional DNS Update mechanisms [RFC2136] [RFC3007] implicitly assume they are being used by a human operator. If a human operator uses DNS Update (perhaps via the 'nsupdate' command) to create a record, then that record should stay created until the human operator decides to remove it.

The same assumptions do not apply to machine-generated records. If a mobile device creates one or more records using DNS Update, and later unceremoniously departs the network, then those stale records should eventually be removed.

The mechanism proposed here is modeled on DHCP. Just like a DHCP address lease, a record created using DNS Update has a lifetime. If the record is not refreshed before its lifetime expires, then the record is deleted.

When a client performs a DNS Update, it includes a EDNS(0) Update Lease option [DNS-UL]. The DNS Update Lease option indicates the requested lifetime of the records created or updated in the associated DNS Update message. In the DNS Update reply, the server returns its own EDNS(0) Update Lease option indicating the granted lifetime, which may be shorter, the same, or longer than the client requested. If the records are not refreshed before the granted lifetime expires, then the records are deleted.

DNS servers may be configured to refuse DNS Updates that do not include a DNS Update Lease option.

4. Sleep Proxy

Another use of Service Registration Protocol is for devices that sleep to reduce power consumption.

In this case, in addition to the DNS Update Lease option [DNS-UL] described above, the device includes an EDNS(0) OWNER Option [Owner].

The DNS Update Lease option constitutes a promise by the device that it will wake up before this time elapses, to renew its records and thereby demonstrate that it is still attached to the network. If it fails to renew the records by this time, that indicates that it is no longer attached to the network, and its records 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 DNS server should act as a proxy for any IPv4 or IPv6 address records in the DNS Update message. This means that the DNS server should send ARP or ND messages claiming ownership of the IPv4 and/or IPv6 addresses in the records in question. In addition, the DNS server 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.

5. First-Come First-Served Naming

In some environments, such as home networks with an appropriate border gateway, it may be preferable to have some limited security on the protected internal network rather than no security at all.

Users have shown limited willingness to endure complicated configuration for their networked home devices. It is rare for home users to change even the factory-default name for their wireless printer, so it's questionable whether it's reasonable to expect them to configure passwords or security keys.

This document presents a zero-configuration first-come first-served naming mechanism.

Instead of requiring a preconfigured key installed by manual administration, a new device optimistically creates its DNS Service Discovery records, plus a DNS SIG(0) public key, using a DNS Update signed with its DNS SIG(0) private key.

The DNS server validates the signature on the message using the SIG(0) key already stored on the name, if present, and otherwise with the key sent in the update, if the requested name is not yet present. The server may check that the two public keys are the same before validating, and refuse the update if they are not, to avoid the cost of verifying the signature.

The lifetime of the DNS-SD PTR, SRV and TXT records [RFC6763] is typically set to two hours. That way, if a device is disconnected from the network, its stale data does not persist for too long, advertising a service that is not accessible.

However, the lifetime of its DNS SIG(0) public key 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 way, even if a device is unplugged from the network for a few days, and its services are not available for that time, no other rogue device can come along and immediately claim its name the moment it disappears from the network. It takes a much longer time before an abandoned name becomes available for re-use.

When using this first-come first-served security mechanism, the server accepting or rejecting the updates utilizes knowledge of the DNS-Based Service Discovery semantics [RFC6763]. Specifically, for all records aside from PTR records, the update must be validly signed

using the SIG(0) key with the same DNS resource record owner name (the name on the left in a traditional textual zone file). For additions or deletions of PTR records, the update must be validly signed using the SIG(0) key with the same DNS resource record owner name as the rdata in the PTR record (the name on the right in a traditional textual zone file).

6. Security Considerations

To be completed.

7. References

7.1. Normative References

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

7.2. Informative References

[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, <http://www.rfc-editor.org/info/rfc2136>.

[RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, DOI 10.17487/RFC3007, November 2000, <http://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, <http://www.rfc-editor.org/info/rfc6760>.

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

[DisProx] Cheshire, S., "Discovery Proxy for Multicast DNS-Based Service Discovery", draft-ietf-dnssd-hybrid-06 (work in progress), March 2017.

[DNS-UL] Sekar, K., "Dynamic DNS Update Leases", draft-sekar-dns-ul-01 (work in progress), August 2006.

[Roadmap] Cheshire, S., "Service Discovery Road Map", draft-cheshire-dnssd-roadmap-00 (work in progress), July 2017.

[Owner] 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.

Authors' Addresses

Stuart Cheshire
Apple Inc.
1 Infinite Loop
Cupertino, California 95014
USA

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

Ted Lemon
Nominum, Inc.
800 Bridge Parkway
Redwood City, California 94065
United States of America

Phone: +1 650 381 6000
Email: ted.lemon@nominum.com