Workgroup: Internet Engineering Task Force
Published: 11 July 2022
Intended Status: Standards Track
Expires: 12 January 2023
A T. Lemon (L. Qin)
uApple Inc. Apple Inc.
t
h
o
r
s
:

Multicast DNS conflict resolution using the Time Since Received (TSR) RR

Abstract

This document specifies a new conflict resolution mechanism for DNS, for use in cases where the advertisement is being proxied, rather than advertised directly, e.g. when using a combined DNS-SD Advertising Proxy and SRP registrar. A new DNS RR is defined that communicates the time at which the set of resource records on a particular DNS owner name was most recently updated.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 12 January 2023.

Table of Contents

1. Introduction

Unlike the Domain Name System [RFC1034], with its authority servers and delegation of authority, Multicast DNS has no single source of authority. Because of this, mDNS has a mechanism, conflict resolution (Section 9 of [RFC6762]) for detecting and fixing conflicts in mDNS advertisements.

The current goal of mDNS conflict resolution is to prevent a new service being advertised from taking the place of an existing service with the same name that is already being advertised. This goal, however, assumes that the entity advertising an mDNS service is in fact authoritative for that service. In the case of an Advertising Proxy [I-D.sctl-advertising-proxy], this is not the case: the source of truth for the service being advertised is an SRP [I-D.ietf-dnssd-srp] client.

On a link with more than one SRP registrar, an SRP client may register with one SRP registrar, and then subsequently update its registration on a different SRP registrar. Both SRP registrars may be acting as advertising proxies. If so, the original server may still be advertising the old SRP registration using mDNS. If the information in the new SRP registration is identical to that in the old registration, this is not a problem. However if some information has changed (e.g., a new IP address has been added, or a TXT record updated), then the new registration will be seen to be in conflict with the old registration.

In the case of such a conflict, the current behavior of mDNS is for the older (stale) registration to win, and the newer (current) information to be discarded. This behavior, which is entirely correct for services that are advertising on their own behalf, is exactly wrong when a service registration is being proxied.

1.1. Current Behavior

When a new service is to be advertised, the server that wishes to advertise its service typically registers the service with a central mDNS registrar on the host on which it is running. This mDNS registrar may have an internal database of services already registered, and may detect a conflict with one of those services. This can be true whether the conflicting database entry is data for which the mDNS registrar is authoritative, or data it has received via mDNS and cached.

In the case of such a conflict, no network transaction is required: the mDNS registrar detects it locally. It addresses the conflict in one of two ways. The first alternative is that the mDNS registrar will report the conflict to the server as an error, which the server must fix. Alternatively, the server may have indicated that the mDNS mDNS registrar should automatically choose a new name for it, in which case the mDNS registrar does so automatically, without notifying the server.

Once any locally-detectable conflicts have been resolved, the mDNS registrar probes (see Section 8.1 of [RFC6762]) local network to see if any other host has already registered a service the conflicts with the proposed new service. If such a service is present on the network, the mDNS registrar follows the same process previously described, either reporting the error to the server or automatically choosing a new name.

The effect of this approach is that generally whichever server first registers a service under a particular name wins. If a server comes along later and registers the same service with conflicting information, the newcomer's information is rejected.

1.2. Problem Statement

The current behavior works well for services registering on their own behalf. However, for example in the case of an SRP registrar, it works poorly: an SRP registrar acting as an advertising proxy proxies the contents of its registration dataset(s) using mDNS. The source of truth for information in such datasets is whatever service has registered with the SRP registrar, not the SRP registrar itself.

In the case of an advertising proxy proxying an SRP dataset, what we want is not the oldest information, but the newest. When the SRP client is able to continue registering with the same SRP registrar, this works well: stale data is automatically removed and replaced with current data. However, if more than one SRP registrar is available, and for some reason the original SRP registrar with which the registration was completed is still operating but no longer reachable (e.g., in the case of a network partition), the SRP client will wind up registering with a different SRP registrar. Similarly, if the SRP service is being advertised using an anycast address, there is no guarantee that the SRP renewal will be delivered to the same SRP registrar.

When the SRP client registers with a different SRP registrar, the behavior we get with the current conflict resolution approach is that the SRP client will be given a new name, and both the old

(stale) advertisement (A) and the new (more recent) advertisement (A') will be discoverable as separate services.

This creates a new burden on consumers of such services: they need to parse through the whole list of services of their type, using metadata from the TXT record in the service instance data, if possible, to determine that service A and service A' are the same service. If no such information is present in the TXT record, the only way to determine that one of these two registrations is stale is to attempt to use the advertised service, which may no longer be reachable if, for example, the change that produced the conflict was an IP address change. When the SRP lease for the stale service expires, that service's advertisement will be removed, and the service will no longer be discoverable under the original name, even if the IP address hasn't changed.

This document proposes an enhancement to the current conflict resolution algorithm for mDNS, which allows an mDNS proxy to report the time at which it received the registration it is newly advertising. This is done using a new Time Since Received RR, which is attached to the name of the registration.

2. Time Since Received Resource Record

The Time Since Received (TSR) RR is given the same owner name as the RRset or RRsets for which it is asserting a received time. When a service registration is successful, the mDNS registrar records the wall clock time at which the registration request was received. This may be the current time, or a time specified by a proxy service that is doing the registration. This time is only recorded if the service requesting the registration specifies it; otherwise, the time of receipt is not recorded.

The TSR RR contains the difference, in seconds, between the the time at which the TSR record is being generated and the time of receipt for recorded for that owner name. If this difference is greater than seven days (7 * 24 * 60 * 60), the mDNS registrar MUST use a value of seven days rather than the larger value.

The resource record is formatted as described in Section 3.2.1 of [RFC1035]. The RDATA consists of the time offset in the form of a 32-bit unsigned number in network byte order.

3. mDNS Registrar Behavior

3.1. When sending a probe

When probing, for each ANY RR in the question section of the probe query, if a time of receipt has been recorded for the owner name of that RR, the mDNS registrar generates a TSR record according to the method described in Section 2. This TSR RR is then added to the authority section of the query, along with the contents of all conflict-producing RRsets it has recorded on that owner name.

A conflict-producing RRset is an RRset with an RRtype that can produce a conflict. Most RRtypes can produce conflicts. RRtypes that do not produce conflicts include the PTR RRtype and the TSR RRtype.

This is new behavior: mDNS registrars that do not use the TSR record typically do not include all such records in the authority section. E.g., for a DNS-SD service instance, only the SRV record is included, not the TXT record, in order to conserve space in the probe packet. This doesn't work for TSR probes, because the mDNS registrar receiving the probe needs to be able to determine if there is a conflict by looking at the probe message alone.

3.2. When processing a probe

A probe is an mDNS query that includes an authority section.

When an mDNS registrar receives a probe query, it processes it as usual, unless it discovers a conflict. When data in the authority section of the probe that matches a particular RR in the question section, and that data conflicts with data in the mDNS registrar's own database, as described in Section 8.1 of [RFC6762], this is considered to be a probe conflict.

Note that TSR records should not exist in the authority database of any mDNS registrar. This would not make sense, because the TSR record encodes a relative time, and in order to generate a TSR record, the mDNS registrar needs an absolute (wall clock) time. They are also not conflict-producing records. So a TSR record can never be the basis for detecting a conflict.

After a probe conflict has been detected, the registrar checks the authority section of the probe query for a TSR record with the same owner name as the name of the question for which a potential conflict was detected. It also checks for a recorded time of receipt on that owner name in its own authority database. If no TSR record is found, or no recorded time of receipt is present, then the probe conflict is processed normally as described in sections 8.1 (paragraph 8), 8.2 and 9.

If both the TSR record in the authority section and the recorded time of receipt are found, then before doing normal processing of the probe conflict, the two times are compared. This is done by subtracting the value contained in the TSR record from the current wall clock time to determine the time of receipt of the data in the probe. If the time of receipt in the probe is more recent than (greater then) the time of receipt in the registrar's authority database, then the registrar sees this as losing the conflict and removes its authority records.

Otherwise, as when both times are not available for comparison, the probe conflict is processed normally as described in sections 8.1 and Section 8.2 of [RFC6762].

3.2.1. Processing multiple questions in a probe

Multicast DNS probes can contain more than one question. The question and authoritative data for each owner name are handled separately, as described above, so it may be that data on some owner names is handled based on the TSR conflict detection process, while data on other owner names is handled using the conflict detection mechanism described in Section 9 of [RFC6762].

3.2.2. The effect of network latency on time computations

Because TSR computations are affected by network latency, comparisons can't be considered accurate. It is therefore necessary to tolerate some amount of error. In practice, however, it should generally not be the case that two advertising proxies receive SRP updates from the same SRP client at nearly the same time. So it should always be the case either that there is a clear ordering to the timestamps, or that there is no conflict in the data. For example with anycast, a retransmission could go to a different SRP registrar, but in this case both servers would simultaneously receive identical data, so the close ordering or even equality of the timestamps should not affect the outcome.

3.3. When sending a reply

Records with the TSR RRtype MUST NOT be send in any section of an mDNS reply.

3.4. When processing a reply

TSR records are not allowed in replies. An mDNS registrar receiving an mDNS reply containing a TSR record MUST silently ignore the TSR record.

4. Internal Handling of TSR records

The TSR record that is sent on the wire is expressed in seconds relative to the time of receipt of the registration. In order to derive a TSR record, the registrar must remember the time at which the registration occurred. This time is recorded as an absolute time, not a relative time. We refer to this as the time of receipt. When sending a TSR RR, the registrar computes the difference between the current time and the time of receipt, which must always be in the past. This difference, which should be a positive integer, is converted to seconds, and that unsigned value is then used to synthesize the TSR RR.

5. Timeliness of Conflict Resolution

It is expected that if a conflict exists, it will be recent, and will be resolved quickly. Different hosts may be able to record shorter or longer time differences. However, because of this expectation of recentness, mDNS registrars should never need to report a TSR of longer than seven days. It's reasonable to expect that every mDNS implementation should be able to remember time intervals of at least seven days.

6. Legacy Registrars

An mDNS registrar that does not support TSR and receives a probe containing a TSR record will treat the TSR record as part of the authoritative data being probed, and will see that data as a conflict. This will produce the correct behavior: the non-implementing registrar will respond with its authoritative data, and the probing registrar will either see a conflict (and treat the probe as having failed) or will see no conflict, since the non-implementing registrar will not have sent a TSR record.

mDNS registrars that support the TSR record MUST NOT send any TSR record in an mDNS response. Consequently, non-implementing mDNS registrars will never see a TSR record other than in a probe, which we have shown will be handled correctly.

7. When to Use TSR

TSR is only relevant for mDNS proxies. Regular (non-proxy) mDNS registrants are not expected to use it, since it will produce the wrong behavior for this use case. An mDNS registrant that is a proxy MUST explicitly request that a TSR be used for conflict resolution. mDNS registrars MUST NOT record a time of receipt unless the registrant has specifically requested it.

8. Registrant API considerations

When an mDNS proxy registers a service and requests the use of a time of receipt, the proxy MUST specify when it received the registration. In order to support this, the API is required not only to allow the registrant to specify that TSR conflict resolution is wanted, but must also provide a way for the proxy to specify an absolute time at which the registration was received.

This is important, for example, in the case of SRP Replication [I-D.lemon-srp-replication], where an SRP registrar may receive a registration from a peer during startup synchronization. This registration will have occurred at some significant amount of time in the past, and so it would be incorrect for the mDNS proxy receiving the registration to use the time that the mDNS proxy registers the service as the time of receipt.

9. Security Considerations

The TSR RR is an optimization: it ameliorates an edge case for mDNS proxies. A malicious host on the same link could use the TSR RR to win conflict resolution processes. However, because TSR is only used by proxies, this technique will not work for normal mDNS service registrations: in that case, normal mDNS conflict resolution is done, and the attacker gains no benefit from using TSR.

Whether or not an mDNS registration has a recorded time of receipt, an attacker can deny service by announcing its own conflicting data and then answering the subsequent probe as described in Section 9 of [RFC6762]. Because it does not include a TSR record in its authority section, it can win the simultaneous conflict resolution process that follows its bogus announcement.

So the TSR-based conflict resolution process creates no new vulnerability. Addressing the existing vulnerability is out of scope for this document. Protocols that rely on mDNS MUST NOT assume that mDNS service is secure or private. If security (authentication, authorization and/or secrecy) are needed, these must be provided at the application layer, or by using DNSSEC rather than mDNS for service discovery.

10. IANA Considerations

IANA is requested to allocate a new RR Type from the DNS Resource Record (RR) TYPEs registry for the 'Time Since Received' Resource Record. The type shall be 'TSR'. The value shall be allocated by IANA. The meaning shall be 'Multicast DNS Time Since Received". Reference shall refer to this document, once published. There is no template specified, and IANA shall determine the registration date.

11. Informative References

12. Normative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <https://www.rfc-editor.org/info/rfc1034>.

[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>.

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

[I-D.lemon-srp-replication] Lemon, T., "Automatic Replication of DNS-SD Service Registration Protocol Zones", Work in Progress, Internet-Draft, draft-lemon-srp-replication-01, 7 November 2021, <https://datatracker.ietf.org/doc/html/draft-lemon-srp-replication-01>.

[I-D.sctl-advertising-proxy] Cheshire, S. and T. Lemon, "Advertising Proxy for DNS-SD Service Registration Protocol", Work in Progress, Internet-Draft, draft-sctl-advertising-proxy-02, 12 July 2021, <https://datatracker.ietf.org/doc/html/draft-sctl-advertising-proxy-02>.

[I-D.ietf-dnssd-srp] Lemon, T. and S. Cheshire, "Service Registration Protocol for DNS-Based Service Discovery", Work in Progress, Internet-Draft, draft-ietf-dnssd-srp-13, 24 April 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-dnssd-srp-13>.

Authors' Addresses

Ted Lemon
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

Email: mellon@fugue.com

Liang Qin
Apple Inc.
One Apple Park Way

Cupertino, California 95014
United States of America

Email: Leonqin0101@gmail.com
Additional contact information:

Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America