


Internet Engineering Task Force  
Internet-Draft  
Intended status: Standards Track  
Expires: 26 October 2022

T. Lemon  
 (L. Qin)  
Apple Inc.  
24 April 2022

A 'Time Since Registration' Resource Record for Multicast DNS  
draft-tllq-tsr-01

## Abstract

This document defines a new DNS Resource Record (RR) to be used with multicast DNS. The new RR is used to communicate the time at which the set of RRsets on a domain name were first registered.

## 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 26 October 2022.

## Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.



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.

This works well for devices acting on their own behalf. However, in the case of advertising proxies, it works poorly: typically an advertising proxy is proxying the contents of its proxy database using mDNS. The source of truth for information in that database is some host that has registered with the proxy, for example using the Service Replication Protocol (SRP).

In the case of an advertising proxy proxying an SRP database, what we want is not the oldest information, but the newest. When the SRP client is able to continue registering with the same SRP server, this works well. However, if SRP is being managed using anycast registration, there is no guarantee that an SRP client will register with the same server each time.

When the SRP client registers with a different server, the behavior we expect 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 seen on the network, as separate services.

This creates a new burden on consumers of that service: they need to parse through the whole list of services of that type, using metadata from the TXT record in the registration if needed to determine that service A and service A' are the same service.

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

## [2.](#) Time Since Registered Resource Record

The Time Since Registered (TSR) RR is attached to the name for which the TSR RR is asserting a registration time. The TSR RR contains the time in seconds since the most recent registration that has been received. This time is computed at the time that the mDNS message is transmitted, and can be treated by the receiver as relative to the current time.

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

When probing, an mDNS registrar reports the TSR for the name for which it is probing. When an mDNS Registrar receives a probe, it checks to see if it has any registration that conflicts with the probe announcement. If it does, it compares its internal TSR with the TSR reported in the probe. If the TSR in the probe is more recent than the internal TSR, the internal registration is marked as stale, and the registrar does not respond to the probe. If the TSR in the probe is older than the internal TSR, the registrar reports a conflict as usual.

Note that because TSR computations are affected by network latency, comparisons can't be considered accurate. It is therefore necessary to tolerate some degree of error. As a general rule, a probe containing a TSR that arrives at a registrar for which the timestamp comparison is close to zero should be assumed to be more recent than the registrar's copy: since the registrar already has a registration, that registration most likely arrived before the registration that triggered the probe.

The Service Registration Protocol uses DNS update, and it takes a significant amount of time for a DNS update client to abandon one DNS server for another, so in the absence of significant congestion-related jitter in packet arrival times, it should never be the case that two SRP proxies receive an SRP update at the same time from the same client. Given that SRP generally does not operate across network infrastructure operator boundaries, such delays are unlikely. Also, if such a situation does occur, the updates should contain the

same data, and therefore should not be seen by the mDNS registrar as being in conflict.

When a probe succeeds, the registrar that did the probe then announces the new service. Registrars receiving this announcement that have internal registrations that conflict with it, which are marked stale, then remove the internal registration and report this event to the proxy that did the registration.

#### [4.](#) Internal Handling of TSR records

The TSR record that is sent on the wire is expressed in seconds relative to the time of 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 will refer to it as the TSR timestamp. When sending a TSR RR, the registrar computes the difference between the TSR timestamp, which must always be in the past, and the current system time. This difference is converted to seconds, and that value

is then sent as 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 systems may be able to record shorter or longer time differences, but because of this expectation of recentness, mDNS registrars should never 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 will treat the TSR record as part of the registration. Since the TSR record is only sent in probes, it will never be erroneously reported to any client that is browsing for services. If a legacy mDNS registrar and an mDNS registrar that supports TSR both advertise the same service, the conflict resolution rules described in [RFC6762](#) will be followed.

#### [7.](#) When to Use TSR

TSR is only relevant for mDNS proxies. It SHOULD NOT be used by regular (non-proxy) mDNS registrants. An mDNS registrant that is a proxy MUST explicitly request that a TSR be used for conflict resolution. mDNS registrars MUST NOT record a TSR timestamp unless the registrant has specifically requested it.

## [8.](#) Registrant API considerations

When an mDNS proxy registers a service and requests the use of a TSR timestamp, 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 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 server 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 TSR timestamp.

## [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. In the case of proxied mDNS registrations, an attacker can in fact deny service by superseding existing registrations.

However, such an attacker could achieve the same effect simply by responding to probes with conflict announcements. Furthermore, such an attack would cause noticeable problems on the network which the network operator would then take steps to correct.



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