

Time Since Received

(preventing friendly conflicts)

Ted Lemon <elemon@apple.com>

Topics

- Operational experience so far
- Adversarial versus Cooperative conflict detection
- Useless probing
- Double transmission because dual stack
- RFC 6762 update?

TSR Operational experience

- When it works, it works
- Lots of ways for it not to work
 - Timing races
 - Incomplete set of records in probe
 - Specific to names, not update record sets
- Requires different conflict detection behavior
 - Needed for normal probes, but
 - Simultaneous probe behavior must also change
- But there is a basic theme in this...

Adversarial versus Cooperative Conflict Detection

- TSR as currently designed assumes multiple non-cooperating services
- But we need it because we're getting conflicts while cooperating
- Can we do better by accounting for the fact that we are cooperating, without running into problems in the non-cooperating case?

How to signal cooperation

- Right now we notice if records are in conflict by first checking to see if all records on the name are the same in the probe and in the authority database
- If we detect a conflict, we use TSR to figure out which update has priority, if we have it for both
- If TSR doesn't help us, we compare RR data
- This means that we are effectively treating each RR conflict separately, and each name separately, when in fact they are all part of the same update.
- We could include in the TSR record a hash of the update. This hash would be the same for all updated records.

Benefits

- We no longer have to keep records together
- Currently, if we don't include *all* the records on a name in the probe, this will be seen as conflicting with the complete set of records in the authority database, even when the probe and the authority database represent the same SRP update
- This allows us to loosen the requirement that all records for a given TSR be in the same probe (right?)

What about updates

- If we get an SRP renewal that just updates the received time without updating anything else, how do we signal that?
- One option would be to include the old hash and the new hash along with the TSR. If I see a record in the authority database with either hash, it's okay
- What if it's somewhat an update and somewhat a renew?
- Can we use the “old hash, new hash” approach here as well?
 - Yes: these are all from the same SRP client, so they can't be in conflict
 - No: ???

TSR is only in probes

- An additional issue is that currently TSR is only in probes.
- We did this because we didn't want clients to see a change in behavior
- Unfortunately this means that if we see a conflict in an answer, we can't do any sort of reasoning based on TSR
- Is there a compromise?
- Can we e.g. include TSR in answers, in the authority section?
- What would this break?

Useless Probing

- Separately, but related, when we have multiple advertising proxies, we wind up sending a lot of /really/ useless mDNS traffic
- This is expensive in terms of airtime if nothing else
- During SRP peer synchronization, this could be a /large/ burst of multicast traffic.
- It's possible that these bursts may result in smart WiFi access points blocking mDNS traffic (we can't prove this, but we suspect this is happening)
- This then reduces the reliability of mDNS generally (if true)

Double Transmission because of Dual Stack

- Whenever we send an mDNS message, we send it once per layer 3 stack, and we retransmit three (four?) times
- This again is a lot of unnecessary traffic
- Can we do anything about this?
- E.g., only transmit IPv4 packets if we get an IPv4 packet?
- Something cleverer?
 - Types of mDNS registrars:
 - General-purpose devices
 - Laptop, phone, speaker, set-top box
 - Peripherals that never get updated
 - Printers

RFC6762bis?

- Is what we actually need here RFC6762bis?
- We are seeing issues that RFC6762 tried to address, but didn't fully anticipate
- Maybe rather than just adding a new record, we need to re-strategize
- I don't want to boil the ocean, but experience thus far with TSR shows that TSR, while sensible, was a bit naive
- We are seeing some very real operational problems with 6762 that I think we can solve

Proposed way forward

- Fix TSR so that we think it works
- Write a sample 6762 update, which I think would just be a “additional considerations” document rather than a full rewrite (but fight me if you disagree)