Running a local copy of the DNS root zone

APRICOT 2017

IoT is great, except when it is not (secured)

- The October 2016 DDoS attack against Dyn was yet another wake up call.
- An effective attack against very well-architected and distributed DNS infrastructure.
- With DDoS, the lines of targets and victims are blurred.

Root Server System

- The root server system is also well-architected and vastly distributed.
- Organizational diversity has significant advantages. 12 operators of the 13 root name servers.
- There have been prior attacks and the root servers remain a significant target.

Further Ensuring Access to the Root Zone

Should resolver operators run a copy of the root zone locally?

- This has been suggested before in RFC 7706, though its motivations were different:
  - Decreasing access time to root servers
  - Privacy concerns

Technical Options

- On resolver, but without DNSSEC
- On resolver, with DNSSEC
- Off resolver, with DNSSEC

Note Well

- The functionality described is nameserver software agnostic.
- The config examples shown are BIND specific due to its ability to act as a recursive as well as authoritative server.
- Configs heavily borrowed from RFC 7706.
- The same functionality can be achieved with Unbound/NSD, Knot Resolver/Knot DNS, etc.

Zone transfers (axfr) required

- All options described require regular zone transfers of the entire root zone.
- The root zone is updated at least once per day. ixfr is not allowed. Each update requires the transfer of the entire zone, currently ~1.25 MB.
- Not all root servers permit xfrs.

On resolver, but without DNSSEC

```
zone "." {
    type slave;
    file "rootzone.db";
    notify no;
    masters {
            192.228.79.201;        # b.root-servers.net
            192.33.4.12;           # c.root-servers.net
            192.5.5.241;           # f.root-servers.net
            192.112.36.4;          # g.root-servers.net
            193.0.14.129;          # k.root-servers.net
            192.0.47.132;          # xfr.cjr.dns.icann.org
            192.0.32.132;          # xfr.lax.dns.icann.org
            2001:500:84::b;        # b.root-servers.net
            2001:500:2f::f;        # f.root-servers.net
            2001:7fd::1;           # k.root-servers.net
            2620:0:2830:202::132;  # xfr.cjr.dns.icann.org
            2620:0:2d0:202::132;   # xfr.lax.dns.icann.org
            };
};
```

On resolver, but without DNSSEC

- With the prior config, all responses for queries to the root would now be authoritative (the AA bit will be set). To our knowledge, this does not change client behavior.
- This is the simplest config to achieve the desired result but leaves you with no DNSSEC validation.

On resolver, with DNSSEC

```
view root {
    match-destinations { 127.12.12.12; };
    zone "." {
        type slave;
        file "rootzone.db";
        notify no;
        masters {
            192.228.79.201; # b.root-servers.net
            192.33.4.12;    # c.root-servers.net
            192.5.5.241;    # f.root-servers.net
            192.0.47.132;   # xfr.cjr.dns.icann.org
            2001:500:84::b; # b.root-servers.net
            2001:500:2f::f; # f.root-servers.net
            2001:7fd::1;    # k.root-servers.net
            2620:0:2830:202::132;  # xfr.cjr.dns.icann.org
            2620:0:2d0:202::132;  # xfr.lax.dns.icann.org
        };
    };
};
```

On resolver, with DNSSEC (continued)

```
view recursive {
    dnssec-validation auto;
    allow-recursion { any; };
    recursion yes;
    zone "." {
        type static-stub;
        server-addresses { 127.12.12.12; };
    };
 };
```

On resolver, with DNSSEC (continued)

- Why did you have to go and introduce views?!

From RFC 7706:

Validation: When using separate views or separate instances, the DS records in the slaved zone will be validated as the zone data is accessed by the recursive server. When using the same view, this validation does not occur for the slaved zone.

Off resolver, with DNSSEC

- Run one or more additional nameservers that slave the root zone and serve it authoritatively.
- The auth config if using BIND would match slide 8.

Off resolver, with DNSSEC

- On your resolvers, use this config to forward queries to the on net auths:

```
zone "." {
type static-stub;
server-addresses { w.x.y.z; };
};
```

Off resolver, with DNSSEC

- Follows best practice of separating recursive and authoritative DNS functions.
- No views!
- However, this does require additional instances of nameservers.

Recommendations

- Maintaining a local copy of the root zone does provide additional resiliency against DDoS on the root server system as well as providing the benefits intended with RFC 7706. However, it does introduce additional areas of breakage and could make debugging of issues more difficult. Consider all of this before making a decision to maintain a local root copy.

Recommendations

- Of the 3 methods reviewed, ISC currently recommends the off resolver method.
- We feel it is the most straightforward configuration-wise even as it requires additional nameserver instances.

Questions

?

Thank You!

ewinstead@isc.org

info@isc.org