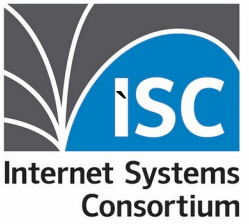
What’s new in BIND 9.20 and what’s coming up next?

Colin Vidal, ISC, May 23 2025



BIND 9.20: what's new

- DNSSEC configuration updates
  - `dnssec-policy` is now the only option to manage a signed zone
    → `dnssec-keymgr` and `auto-dnssec` are removed
  - HSM support is configured through `dnssec-policy`:

```
dnssec-policy {
    pkcs11-uri <quoted_string>;
};
```

- Support of Catalog Zone schema version 2
- PROXYv2 is available also for DNS over TCP and DNS over TLS
- Statistics channels are showing incoming zone transfer in progress
  http://<ip>:<port>/xml/v3/xfrins
  http://<ip>:<port>/json/v1/xfrins

BIND 9.20: USDT probes

(User Statically-Defined Tracing probes)

Instrumentation of production binaries of BIND 9

User-space program

```
result = foo();
FIRE_PROBE_FOO_ENDS(result);
...
result = bar();
FIRE_PROBE_BAR_ENDS(result);
```

Kernel module (built/loaded from `stap`)

```
void fooends(int result) {
     printf("foo ends with value %d\n",
            result);
}

void barends(int result) {
    printf("bar ends with value %d\n",
           result);
}
```

BIND 9.20 adds probes support for `rwlock` and incoming zone transfer flows

How-To: https://gitlab.isc.org/isc-projects/bind9/-/wikis/User-space-Probing-in-BIND-9

BIND 9.20: USDT probes

(User Statically-Defined Tracing probes)

Example using `perf` and analyzing incoming transfer

```
# perf buildid-cache --add /usr/local/lib/libdns.so
# pref list | grep xfrin
[… choose the one you’re interested in …]
# perf probe –add="sdt_libdns:xfrin_recv_start"
# perf record -e sdt_libdns:xfrin_recv_start -aR sleep 5
[… trigger zone transfer in named …]
# perf script
isc-loop-0005 2332966 [002] 940053.314601:
                        sdt_libdns:xfrin_recv_start: (7f37d5d28d44)
                                                     arg1=139877381898240
                                                     arg2=139877381899456
                                                     arg3=0
```

BIND 9.20: Extended DNS Error

- Response policy zone EDEs 15, 16, 17, 18

```
options {
  response-policy {
     zone "example.com." ede none|blocked|censored|filtered|prohibited;
  };
};
```

- From 9.20.6
  - Unsupported RRSIG algorithm (EDE 1)
  - Unsupported DNSKEY digest (EDE 2)
  - Multiple EDEs (up to 3) are supported in the same DNS response
- From 9.20.8
  - Signature expired (EDE 7)
  - Signature not yet valid (EDE 8)
  - Not authoritative (EDE 20)

BIND 9.20: a new cache and zone DB

qp-trie

https://dotat.at/prog/qp/README.html

- A key-value store which is:
  - Transactional
  - Particularly suited for DNS
- BIND9 implementation of qp-trie is a foundation to a lock-free database:
  - Use Userspace RCU for updates

    (userspace-rcu is now a mandatory library to build BIND 9)
  - But currently the values stored in the DB use locking
- Replaced red-black tree (which was using locking)

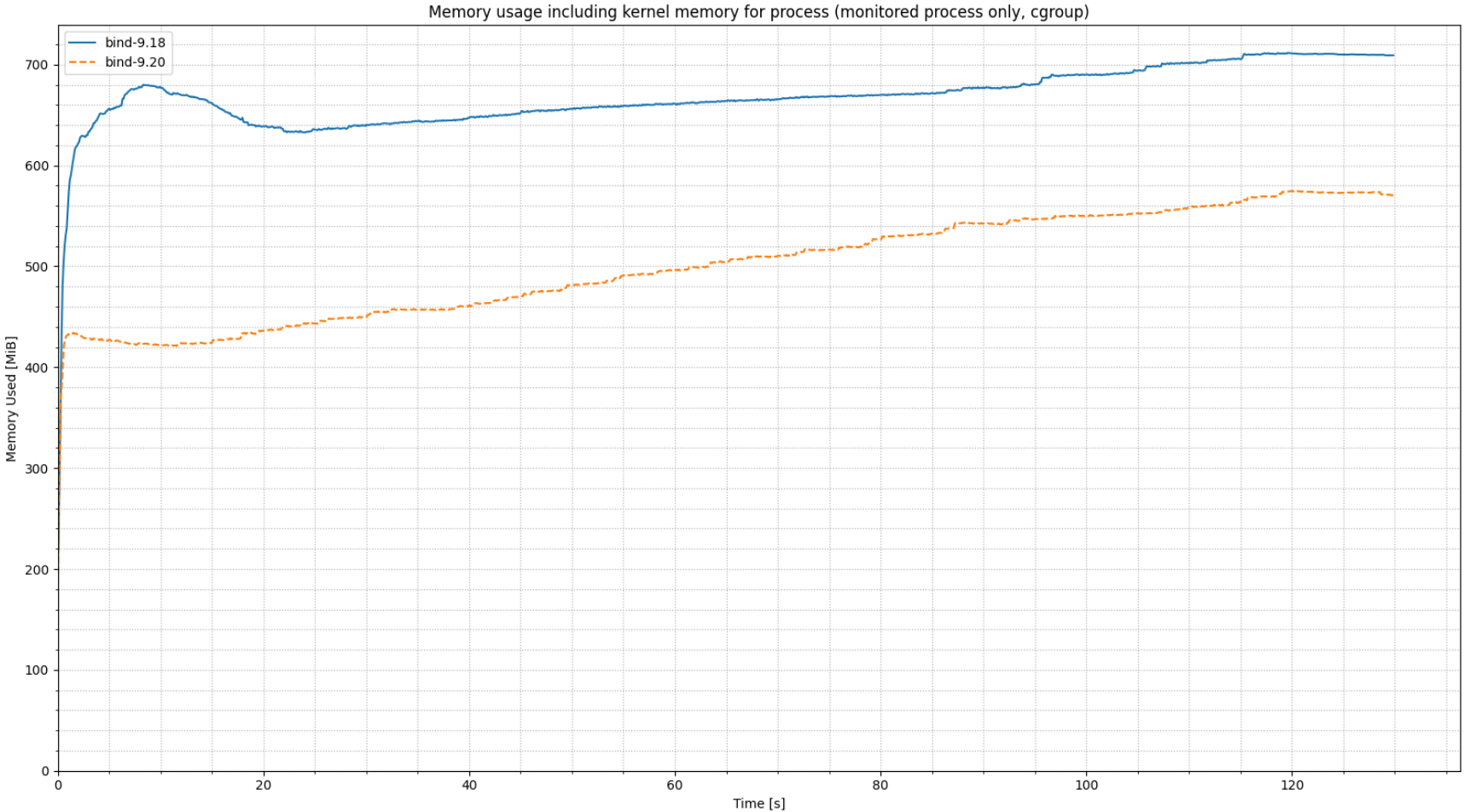
Quicker lookups, smaller memory footprint

BIND 9.20: libuv

- Introduced in 9.16 with a new network manager, initially to handle network events
- In 9.20, it replaced the custom-made cooperative scheduling in the whole server.
- Enables to dispatch events in different categories:
  - Very fast, i.e. cached query response
  - Crypto (slow) operations on a dedicated thread-pool
  - Slow and blocking (i.e. IO operations) on another dedicated thread-pool

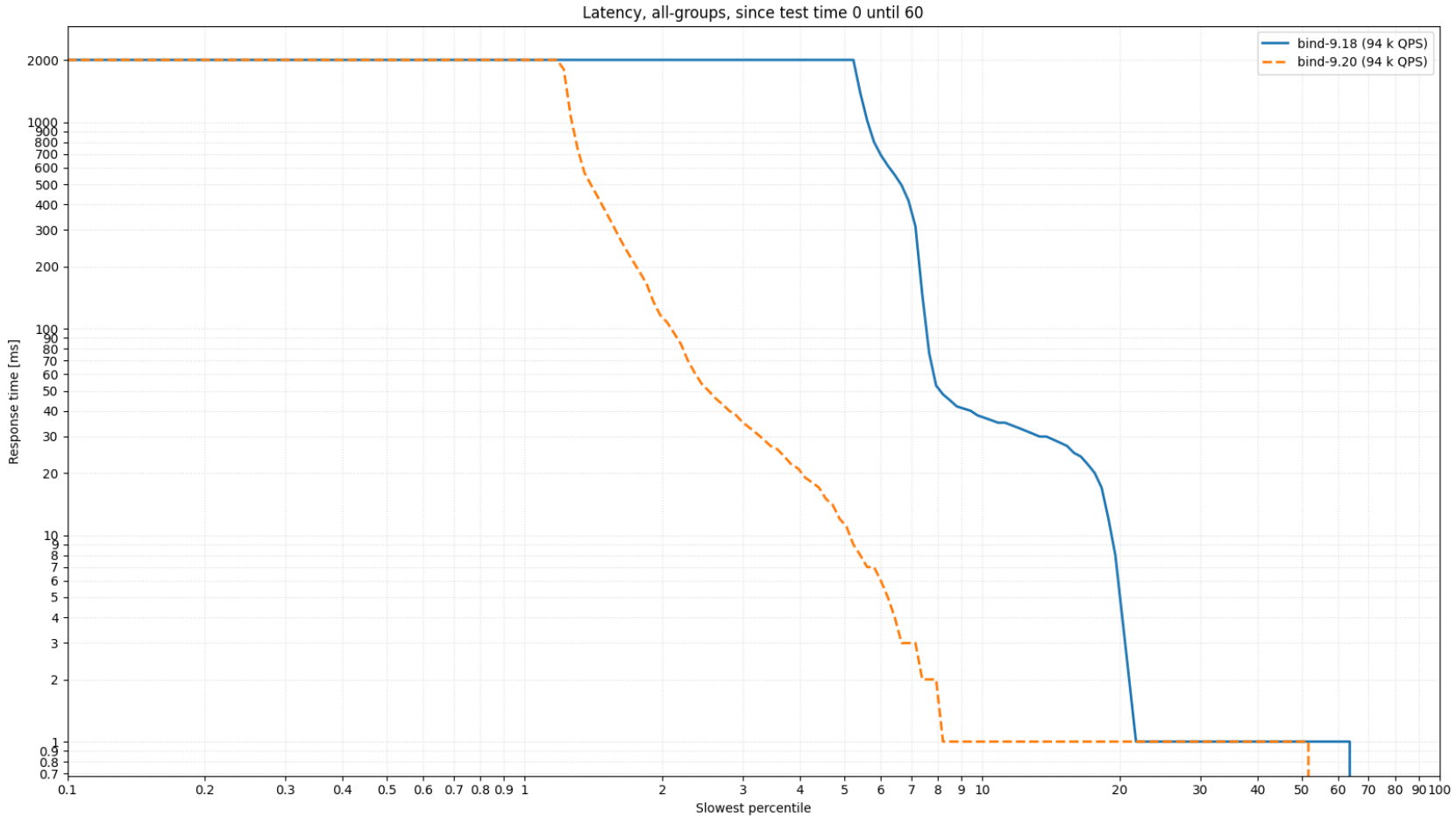
Simplifies the internal architecture,
helps us to focus on what's matter (we're doing DNS, not scheduling),
and reduces context-switching as events processing are pinned to threads.
(and the OS takes care of fair thread scheduling)

BIND 9.20: Performance



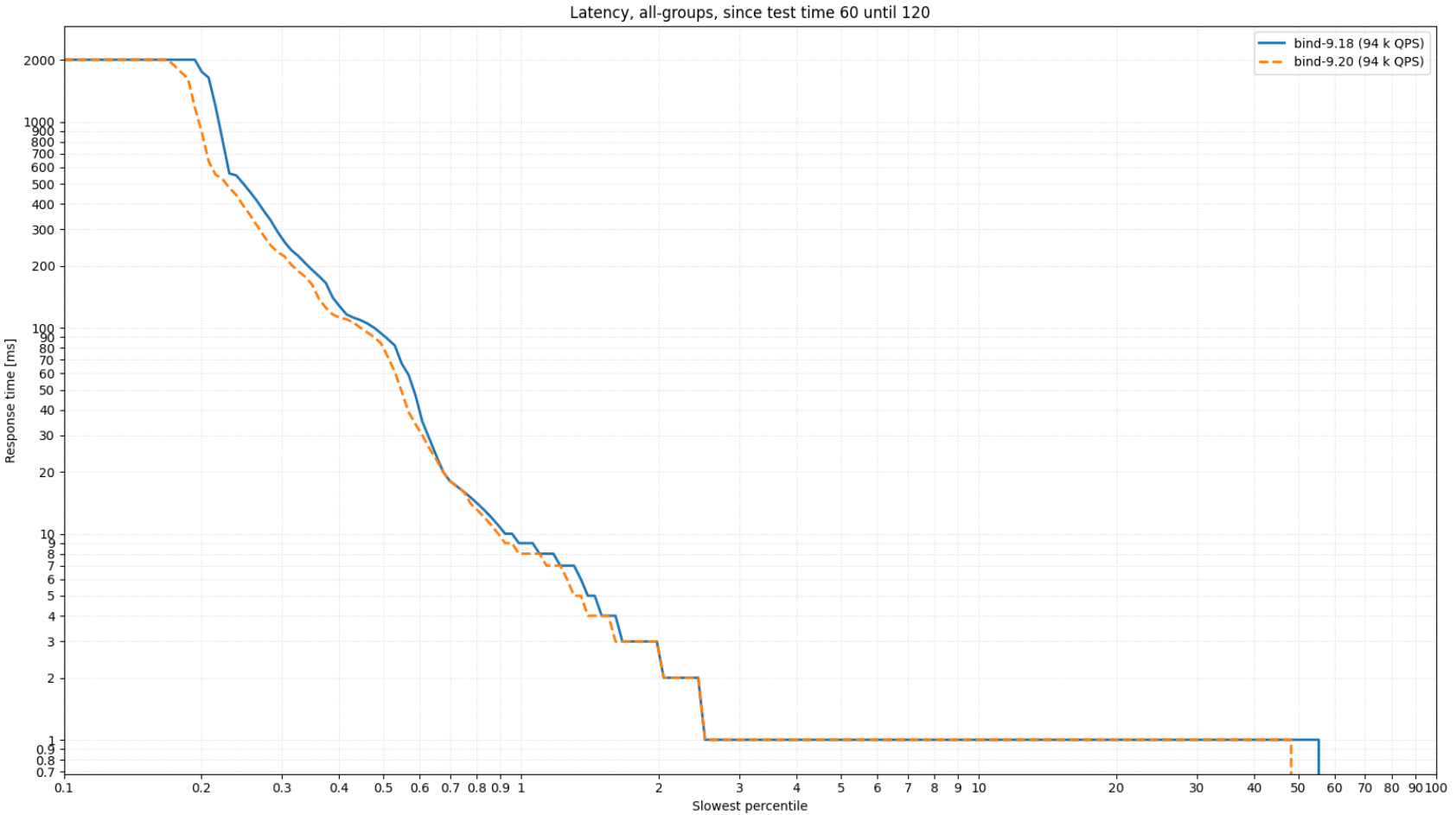
BIND 9.20: Performance

Lattency on a cold cache



BIND 9.20: Performance

Lattency on a hot cache

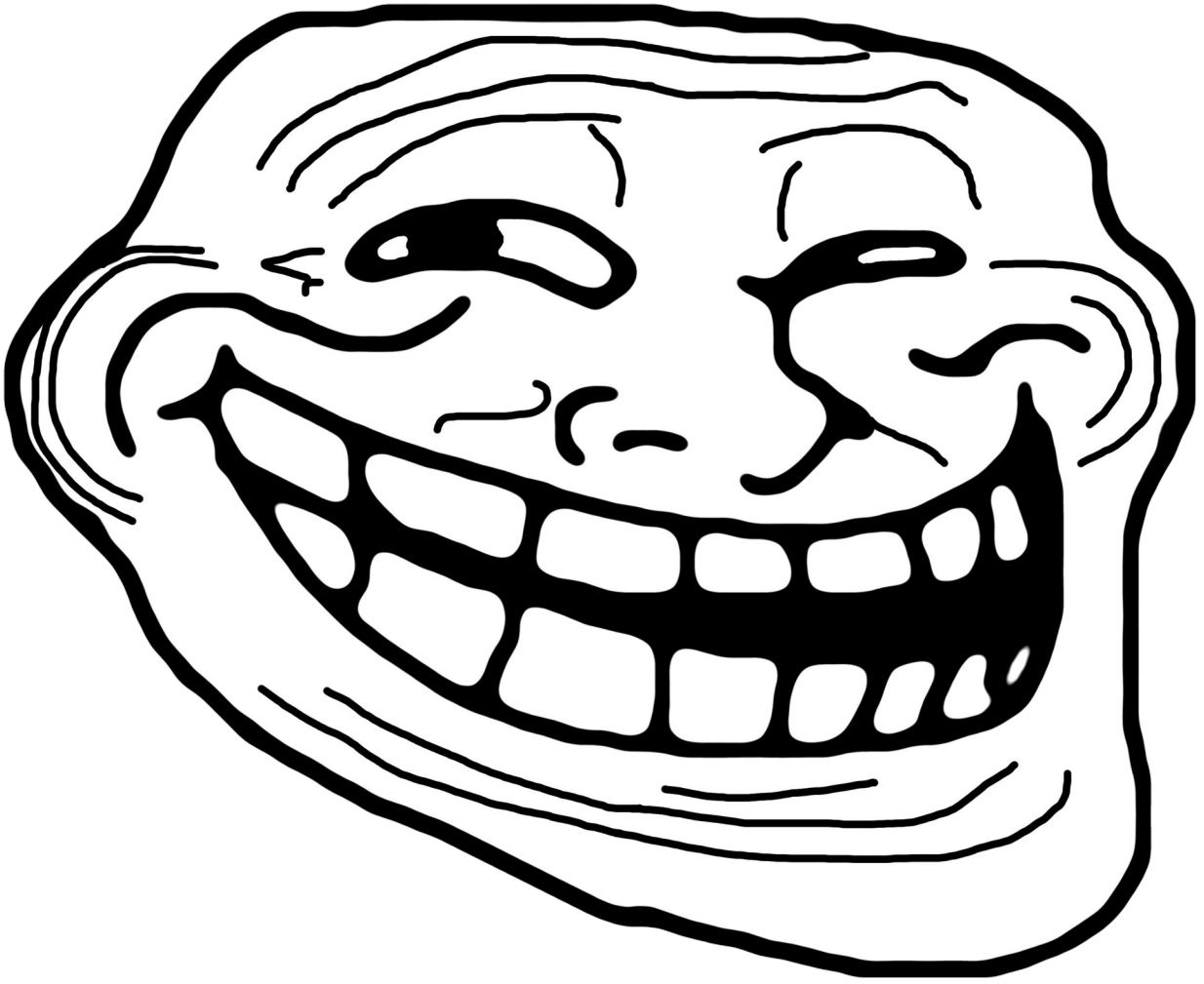


BIND 9.20: Performance

```
options {
     prefetch-llm-backend: llama | gpt4.0;
};
```

Using LLMs to predict most
common queries and prefetch them

→ 42% less time out on a cold cache



Future of BIND: new build system

`autotools → meson`

- *Way* shorter build time
  (using hyperfine, Intel Ultra 7 165U machine running Linux 6.14.2)

	`autotools`	`meson`
configure	6.39s	3.02s
build	29.46s	6.16s

- Build commands:

```
meson setup [--prefix=<prefix>] builddir
ninja -C builddir
meson install -C buildir
```

- Requires `python3` and `ninja`

Question for you: are you building BIND 9 from sources? Or using distro-packages?

Future of BIND: Zone templates

Simplify the configuration of multiple zones with similar properties

```
template foo {
    type primary;
    file “$name.db”;
};

zone “example1.org” {
    template foo;
};

zone “internal” {
    template foo;
    allow-query { 192.168.1.0/24; };
};
```

```
zone “example1.org” {
    type primary;
    file “example1.org.db”;
};

zone “internal” {
    type primary;
    file “internal.db”;
    allow-query { 192.168.1.0/24; };
};
```

Future of BIND: Zone templates

A template can be built from a template too

```
template foo {
    type primary;
    file “$name.db”;
};

template bar {
    template foo;
    allow-update { any; };
};

zone “internal” {
    template bar;
    allow-query { 192.168.1.0/24; };
};
```

```
zone “internal” {
    type primary;
    file “internal.db”;
    allow-update { any; };
    allow-query { 192.168.1.0/24; };
};
```

Future of BIND: Admin API

Need to change a setting from a running server, or add/remove a zone?

Update `named.conf` → `rndc reconfig`

Manual process, it stops the server for a little while

Need to update an existing zone?

Update the zone DB → `rndc reload`

Manual process, it stops the server potentially for a *long* while

... BTW, don't forget to increment the SOA number!

`rndc {add,del,update}zone`

Watchout the manpages!

Some changes might be persistent (but not applied in `named.conf`).

Some changes might require manual `named.conf` changes to persist.

Future of BIND: Admin API

Update setting “X” while running, without traffic disruptions

X = { ACL,
zone,
view,
random-config-value-you-name-it!}

Automatic generation of the “running” named.conf

JSON and REST

Don’t put the server in an invalid state if a live configuration update is wrong (zone error, wrong ACL, etc.)

...Something goes wrong? Rollback to the previous working state!

Make deployment of new configs/numerous named servers easier

Question for you: are you building automation tools on top of RNDC (and scripts)? Would you be happy to use a “standard” HTTP-based interface instead?

Future of BIND: also...

- More EDEs
  Next implemented will likely be DNSSEC related-ones (Indeterminate, Bogus, DNSKEY Missing and RRSIG Missing)
  https://gitlab.isc.org/isc-projects/bind9/-/issues/1836

- More work on DBs
  Work towards an entirely lock-free database

- Focusing on better scale on modern CPUs
  see *How to make BIND 9 fast(er)*, Ondřej Surý, FOSDEM 2025
  https://fosdem.org/2025/schedule/event/fosdem-2025-4626-how-to-make-bind-9-fast-er-/

Thank you!

Colin Vidal, ISC, May 23 2025