DNSSEC for Humans and BIND 10

Paul Vixie
Internet Systems Consortium
June 9, 2011

Agenda

- BIND and DNSSEC
- Why do I want DNSSEC?
- Why DNSSEC for Humans?
- BIND 9.7 Features
- More DNSSEC for Humans
- Why BIND 10?
- How We Do BIND 10
- State of BIND 10
- What Production Ready Will Mean
- Why You Want to Try It
- Supporting ISC and BIND

BIND and DNSSEC…

- DNSSEC as we know it today was finalized as a standard in 2004
- Introduced in BIND 9.3
- Operational and developmental moving target ever since
- Deployment is growing steadily

Why do I want DNSSEC?

- DNSSEC has come of age
- The root zone is signed
- It’s the right thing to do…
- Deployment obstacles…
- Enter: DNSSEC for Humans

Why DNSSEC for Humans?

An example of why we call recent BIND 9 work "DNSSEC for Humans"

Take the previous commands for creating a typical set of DNSSEC keys in BIND with NSEC:

ZSK: `dnssec-keygen -a RSASHA1 -b 1024 -n ZONE example.com`

KSK: `dnssec-keygen -a RSASHA1 -b 2048 -f KSK -n ZONE example.com`

And if you wanted NSEC 3….

- ZSK: `dnssec-keygen -a NSEC3RSASHA1 -b 1024 -n ZONE example.com`
- KSK: `dnssec-keygen -a NSEC3RSASHA1 -b 2048 -f KSK -n ZONE example.com`

Same required arguments, but now you have to remember how to spell NSEC3RSASHA1…

In BIND 9.7

- DNSSEC for Humans style:
- For NSEC:
  - `ZSK: dnssec-keygen example.com`
  - `KSK: dnssec-keygen -fk example.com`
- For NSEC3:
  - `ZSK: dnssec-keygen -3 example.com`
  - `KSK: dnssec-keygen -3 -fk example.com`

Smart signing

- The old way:
  - `cat *.key example.com > zone`
  - `dnssec-signzone -o example.com -k <ksk> -f example.com.signed zone <zsk>`
- The new way:
  - `dnssec-signzone -S example.com`
- Keys are imported into the zone automatically
- NSEC/NSEC3 parameters are retained when a zone is re-signed

Fully Automatic Signing of Zones

- In BIND 9.7, `named` can import keys from a key directory and start signing.

- The private key file format has been extended to contain key timing metadata, allowing the administrator to schedule when a key will be scheduled, published, and revoked.

Automated Trust Anchor Maintenance

- *RFC 5011, Automated Updates of DNS Security (DNSSEC) Trust Anchors*, documents a method for automated, authenticated, and authorized updating of DNSSEC "trust anchors”.

- The new managed-keys statement provides named with trusted keys which are automatically kept up to date using RFC 5011.

Simplified configuration of DLV

- A new configuration setting `auto` was added for the `dnssec-lookaside` option.
- This enables DLV by using the dlv.isc.org repository and provides a built-in key for it.
- This feature defaults to off but the key is included for ease of DLV administration.
- What is this DLV, you say?

DLV is…

- DLV (DNSSEC Look-aside Validation) is an extension to the DNSSECbis protocol. It is designed to assist in DNSSEC adoption by simplifying the configuration of recursive servers.
- DLV provides an additional entry point (besides the root zone) from which to obtain DNSSEC validation information.
- ISC recognizes that as the root zone is signed, DLV is nearing the end of its usefulness, however it will remain useful and available until the need for it no longer exists.

Simplified DDNS Configuration

- The update-policy zone option has been extended to add a `local` setting to enable Dynamic DNS for a zone. named will generate a `TSIG` session key at startup which will be used for these updates.
- The `nsupdate` tool now has a `-l` switch to tell it to sign updates using the generated session key and to send the update requests to the `localhost`.
- The new `ddns-confgen` tool may be manually used to create a local authentication key and generate an example configuration for `named.conf` and the `nsupdate` syntax.

Why is DDNS relevant to DNSSEC?

With these new dynamic DNS features, it is also now easier to configure automatic zone re-signing for DNSSEC.

Improved and extended libdns library

The BIND 9 DNS libraries are available for use with third-party (non-BIND) applications.

BIND 9.7.0 introduces new libdns DNSSEC features including:

- DNS client API with support for DNSSEC and dynamic updates
- DNSSEC-aware `getaddrinfo()` and `getnameinfo()`

Improved Ease of Use in PKCS#11

- Public Key Cryptography Standard #11 (PKCS#11) defines a platform- independent API for the control of hardware security modules (HSMs) and cryptographic support devices.

- Updates in BIND 9.7:
  - README.pkcs11 updated
  - Added support for the AEP KeyPer HSM to existing support for the Sun SCA 6000 cryptographic acceleration board
  - Patch to OpenSSL provides two PKCS#11 engines `sign-only` and `crypto-accelerator`
  - New PKCS#11 tools for HSM operations:
    - `pkcs11-keygen` -- for generating RSA key pairs on the device
    - `pkcs11-list` -- for listing the PKCS#11 objects
    - `pkcs11-destroy` -- for destroying keys stored on the device

More DNSSEC for Humans

- BIND 9.8 added the GOST algorithm and a sample root key for ease of configuration for the signed root.
- BIND 9.8 is the most recent currently supported BIND version.

And More...

- BIND 9.9:
- Improvments to the key management and signing processes of previous BIND 9 versions. This is intended to make key management far easier and the behavior of BIND 9's internal signer more functional.
- The internal signer will also be able to function as a “bump in the wire” signer, where it may transfer data in using AXFR, sign it, and publish it as a signed zone file. The requirement that zones managed by the internal BIND signer be dynamic will also be removed at this point.
- BIND 9.9 will be available in fall 2011.

Why BIND 10?

Motivation for BIND 10

- BIND 9 is 10 years old
- The computing world has changed
- The networking world has changed
- DNS software “marketplace” evolved
- A new architecture for the next 10+ years

Open Collaborative Development

- BIND 10 development is public
  - https://bind10.isc.org
  - bind10-dev@lists.isc.org
  - Public Git repository
- BIND 10 team coding away
  - 8 ISC Developers and Support Staff
  - 3 JPRS Developers
  - 7 CNNIC Developers and Testers
  - 1 CZ.NIC Developer

Open Collaborative Development

- BIND 10 development is public
  - https://bind10.isc.org
  - bind10-dev@lists.isc.org
  - Public Git repository
- BIND 10 team coding away
  - 8 ISC Developers and Support Staff
  - 3 JPRS Developers
  - 7 CNNIC Developers and Testers
  - 1 CZ.NIC Developer

State of BIND 10

Current Status: Overview

- Functioning DNS server
  - Authoritative
  - Recursive
  - Forwarder
- DNS libraries, C++ and Python

Current Status: Authoritative

- SQLite & in-memory data sources
- DNSSEC support (NSEC & NSEC3)
- IPv4 and IPv6
- Master, Slave support via AXFR

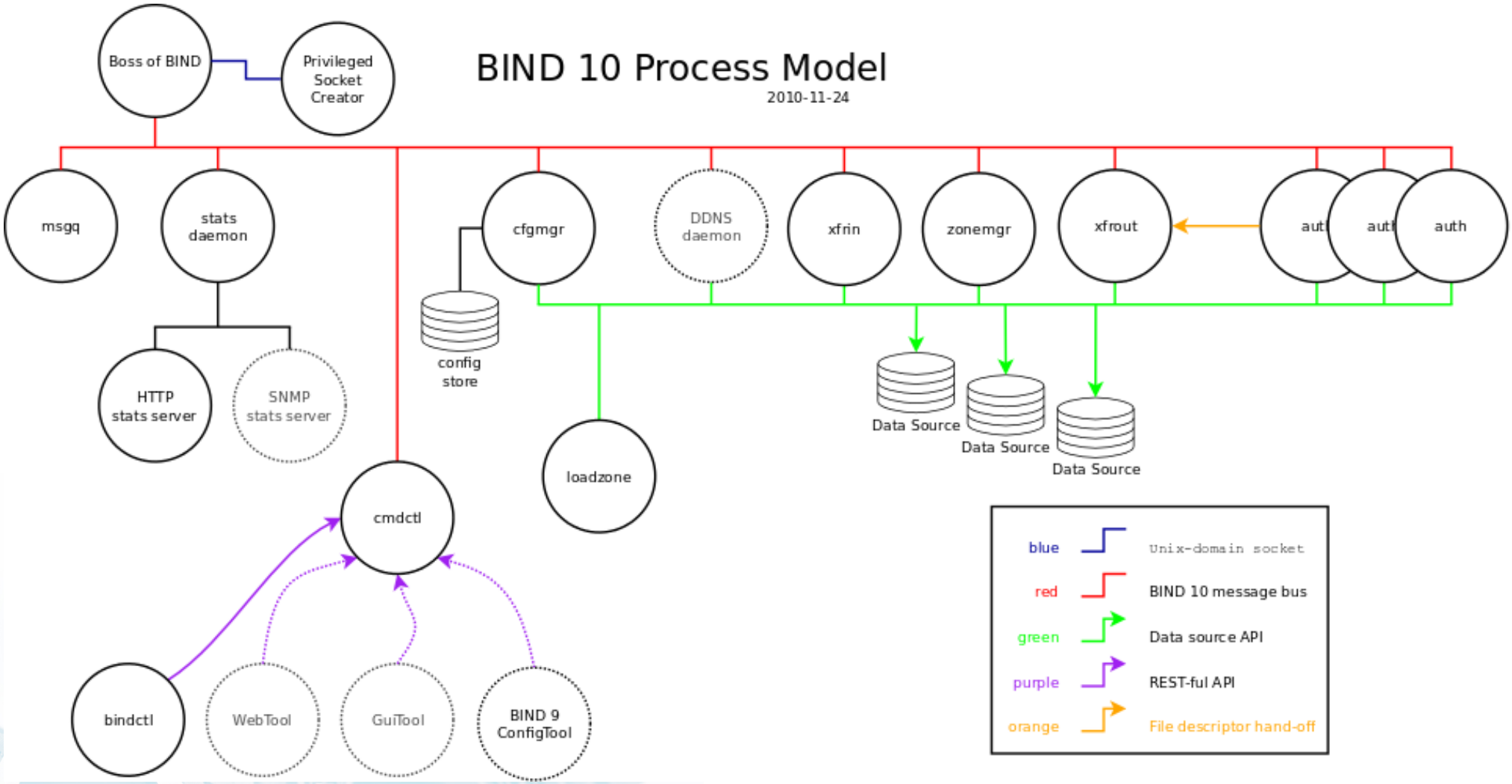
Data Sources

- Authoritative queries can be answered by various data sources
- SQLite
  - An embedded SQL
  - Instant startup, slower answers
- In-memory
  - Based on BIND 9 red/black trees
  - Currently reads zone files on startup

Master & Slave Setup

- Model: cooperating processes
  - Similar to Postfix
- xfrout: act as a master
  - Works with authoritative server
- xfrin & zonemgr: act as a slave
  - Also works with authoritative server

BIND 10 Process Model



https://bind10.isc.org/wiki/DesignDiagrams

Why Multi-Process?

- It allows customization
  - Don't need a feature? Don't run it!
  - Not the only way to customize
- Scaling comes naturally
- Fault isolation
  - Errors in one part do not break the entire system
  - Broken components can be restarted

Configuration

- Goal: simple and sophisticated
  - No restarts necessary
  - Immediate feedback
- New model: router configuration
  - Allows checks at runtime, rollback
  - Old config file still possible
- REST-ful interface available
  - Simple HTTP + XML for custom UI

Current Status: Resolver & Forwarder

- Performs basic DNS resolution
  - Working cache
  - Sophisticated NS selection
  - IPv4 and IPv6

Current Status: DNS Libraries

- C++ DNS library
- Python DNS library
  - Wrapper to C++ library
  - Same semantics, high speed

Production Readiness

What Is Production Readiness?

- Answer depends on your environment
- Features & Performance
  - What you need, you need
  - Nothing else matters!
- Stability
  - Should not fail
  - Failures must be manageable

Production Strategy: Features

- Pending feature list
  - Sorted based on survey feedback, sponsor needs, and intuition
- Snapshot releases every 6 weeks
  - At least one user-visible feature per release

Pending Features

- TSIG
- Logging
- ACL
- Views
- IXFR
- DDNS
- Hooks

- Command tool
- Support tools
- DNSSEC validation

Production Strategy: Stability

- Tests
- Security Audit
- Production Experience

Stability: Tests

- Build tests (done)
- Unit tests (done)
- Functional tests (pending)
  - Including recursive test framework
- Interoperability testing (pending)
  - Comparison with other DNS software

Stability: Production Experience

- Nothing beats running code
- Eating our own (Dog)food
  - `bind10.isc.org`
  - AS112
  - Public test resolver
  - ISC in-house resolver
  - SNS
  - F root name server
- External test program

Why and How You Should Try Out BIND 10

Why BIND 10?
Operational Advantages

- Modularity
- Well-defined APIs and libraries
- Full runtime control
- Flexible, robust, intuitive command line tool
- Customization
- Resilient to failures
- Clustering support

How BIND 10?

- Not a replacement for BIND 9 yet - "drop in" BIND 9 replacement is a Y4 goal

- Run in a separate instance
- Choose a specific use

How You Can Support

- BIND 10 Needs...
- Beta Testers
- (join bind10-beta@lists.isc.org)
- Kibbitzers
- (join bind10-announce@lists.isc.org)
- Developers
- (join bind10-dev@lists.isc.org)
- Sponsors
- (contact us directly)

- ... you!

Thank You to our Generous Sponsors

cira .ca

Afilias

jPRS
JAPAN REGISTRY SERVICES

CNNIC
中国互联网络信息中心
China Internet Network Information Center

.fr AFNIC

.de DENIC

nominet

RIPE NCC

SIDN

Google

.co.za
....Registry Services
A division of UniForum SA

registro.br

cz nic | CZ DOMAIN REGISTRY

ISC

How to Get BIND

BIND is available at:

https://www.isc.org/download/software/bind

If you are interested in participating in BIND beta programs, please register at:

BIND 9:
https://lists.isc.org/mailman/listinfo/bind-beta-response

BIND 10: http://bind10.isc.org/

BIND 9.9 will be released in Q3 2011

Developmental BIND 10 releases are made every 6 weeks.

How to Support ISC

Companies and individuals can learn more about ISC and BIND, our other software, operational programs, support, consulting, and training services, at

http://www.isc.org

As a non-profit open source software company, we rely upon donations and membership in our organization, forums and services to thrive.

Thank you for your support.