1 Release Notes for BIND Version 9.11.0rc3

1.1 Introduction

BIND 9.11.0 is a new feature release of BIND, still under development. This document summarizes new features and functional changes that have been introduced on this branch. With each development release leading up to the final BIND 9.11.0 release, this document will be updated with additional features added and bugs fixed.

1.2 Download

The latest versions of BIND 9 software can always be found at http://www.isc.org/downloads/. There you will find additional information about each release, source code, and pre-compiled versions for Microsoft Windows operating systems.

1.3 License Change

With the release of BIND 9.11.0, ISC is changing the open source license for BIND from the ISC license to the Mozilla Public License (MPL 2.0). This change is effective from BIND 9.11.0b1 onwards.

The MPL-2.0 license requires that if you make changes to licensed software (e.g. BIND) and distribute them outside your organization, that you publish those changes under that same license. It does not require that you publish or disclose anything other than the changes you made to our software.

This new requirement will not affect anyone who is using BIND without redistributing it, nor anyone redistributing it without changes, therefore this change will be without consequence for most individuals and organizations who are using BIND.

Those unsure whether or not the license change affects their use of BIND, or who wish to discuss how to comply with the license may contact ISC at https://www.isc.org/mission/contact/.

1.4 Security Fixes

- It was possible to trigger a assertion when rendering a message using a specially crafted request. This flaw is disclosed in CVE-2016-2776. [RT #43139]

- getrrsetbyname with a non absolute name could trigger an infinite recursion bug in lwresd and named with lwres configured if when combined with a search list entry the resulting name is too long. This flaw is disclosed in CVE-2016-2775. [RT #42694]

1.5 New Features

- A new method of provisioning secondary servers called "Catalog Zones" has been added. This is an implementation of draft-muks-dnsop-dns-catalog-zones/ .

  A catalog zone is a regular DNS zone which contains a list of "member zones", along with the configuration options for each of those zones. When a server is configured to use a catalog zone, all the zones listed in the catalog zone are added to the local server as slave zones. When the catalog zone is updated (e.g., by adding or removing zones, or changing configuration options for existing zones) those changes will be put into effect. Since the catalog zone is itself a DNS zone, this means configuration changes can be propagated to slaves using the standard AXFR/IXFR update mechanism.

  This feature should be considered experimental. It currently supports only basic features; more advanced features such as ACLs and TSIG keys are not yet supported. Example catalog zone configurations can be found in the Chapter 9 of the BIND Administrator Reference Manual.

  Support for master entries with TSIG keys has been added to catalog zones, as well as support for allow-query and allow-transfer.

- Added an **isc.rndc** Python module, which allows **rndc** commands to be sent from Python programs.

- Added support for DynDB, a new interface for loading zone data from an external database, developed by Red Hat for the FreeIPA project. (Thanks in particular to Adam Tkac and Petr Spacek of Red Hat for the contribution.)

  Unlike the existing DLZ and SDB interfaces, which provide a limited subset of database functionality within BIND --- translating DNS queries into real-time database lookups with relatively poor performance and with no ability to handle DNSSEC-signed data --- DynDB is able to fully implement and extend the database API used natively by BIND.

  A DynDB module could pre-load data from an external data source, then serve it with the same performance and functionality as conventional BIND zones, and with the ability to take advantage of database features not available in BIND, such as multi-master replication.

- Fetch quotas are now compiled in by default: they no longer require BIND to be configured with **--enable-fetchlimit**, as was the case when the feature was introduced in BIND 9.10.3.

  These quotas limit the queries that are sent by recursive resolvers to authoritative servers experiencing denial-of-service attacks. They can both reduce the harm done to authoritative servers and also avoid the resource exhaustion that can be experienced by recursive servers when they are being used as a vehicle for such an attack.

  - `fetches-per-server` limits the number of simultaneous queries that can be sent to any single authoritative server. The configured value is a starting point; it is automatically adjusted downward if the server is partially or completely non-responsive. The algorithm used to adjust the quota can be configured via the `fetch-quota-params` option.
  - `fetches-per-zone` limits the number of simultaneous queries that can be sent for names within a single domain. (Note: Unlike "fetches-per-server", this value is not self-tuning.)

  Statistics counters have also been added to track the number of queries affected by these quotas.

- Added support for **dnstap**, a fast, flexible method for capturing and logging DNS traffic, developed by Robert Edmonds at Farsight Security, Inc., whose assistance is gratefully acknowledged.

  To enable **dnstap** at compile time, the **fstrm** and **protobuf-c** libraries must be available, and BIND must be configured with `--enable-dnstap`.

  A new utility **dnstap-read** has been added to allow **dnstap** data to be presented in a human-readable format.

  rndc dnstap -roll causes **dnstap** output files to be rolled like log files -- the most recent output file is renamed with a `.0` suffix, the next most recent with `.1`, etc. (Note that this only works when **dnstap** output is being written to a file, not to a UNIX domain socket.) An optional numerical argument specifies how many backup log files to retain; if not specified or set to 0, there is no limit.

  rndc dnstap -reopen simply closes and reopens the **dnstap** output channel without renaming the output file.

  For more information on **dnstap**, see http://dnstap.info.

- New statistics counters have been added to track traffic sizes, as specified in RSSAC002. Query and response message sizes are broken up into ranges of histogram buckets: TCP and UDP queries of size 0-15, 16-31, ..., 272-288, and 288+, and TCP and UDP responses of size 0-15, 16-31, ..., 4080-4095, and 4096+. These values can be accessed via the XML and JSON statistics channels at, for example, http://localhost:8888/xml/v3/traffic or http://localhost:8888/json/v1/traffic.

  Statistics for RSSAC02v3 traffic-volume, traffic-sizes and rcode-volume reporting are now collected.

- A new DNSSEC key management utility, **dnssec-keymgr**, has been added. This tool is meant to run unattended (e.g., under **cron**). It reads a policy definition file (default `/etc/dnssec-policy.conf`) and creates or updates DNSSEC keys as necessary to ensure that a zone's keys match the defined policy for that zone. New keys are created whenever necessary to ensure rollovers occur correctly. Existing keys' timing metadata is adjusted as needed to set the correct rollover period, prepublication interval, etc. If the configured policy changes, keys are corrected automatically. See the **dnssec-keymgr** man page for full details.

Note: **dnssec-keymgr** depends on Python and on the Python lex/yacc module, PLY. The other Python-based tools, **dnssec-coverage** and **dnssec-checkds**, have been refactored and updated as part of this work.

dnssec-keymgr now takes a -r *randomfile* option.

(Many thanks to Sebastián Castro for his assistance in developing this tool at the IETF 95 Hackathon in Buenos Aires, April 2016.)

- The serial number of a dynamically updatable zone can now be set using **rndc signing -serial *number zonename***. This is particularly useful with inline-signing zones that have been reset. Setting the serial number to a value larger than that on the slaves will trigger an AXFR-style transfer.
- When answering recursive queries, SERVFAIL responses can now be cached by the server for a limited time; subsequent queries for the same query name and type will return another SERVFAIL until the cache times out. This reduces the frequency of retries when a query is persistently failing, which can be a burden on recursive servers. The SERVFAIL cache timeout is controlled by servfail-ttl, which defaults to 1 second and has an upper limit of 30.
- The new **rndc nta** command can now be used to set a "negative trust anchor" (NTA), disabling DNSSEC validation for a specific domain; this can be used when responses from a domain are known to be failing validation due to administrative error rather than because of a spoofing attack. NTAs are strictly temporary; by default they expire after one hour, but can be configured to last up to one week. The default NTA lifetime can be changed by setting the nta-lifetime in named.conf. When added, NTAs are stored in a file (*viewname*.nta) in order to persist across restarts of the **named** server.
- The EDNS Client Subnet (ECS) option is now supported for authoritative servers; if a query contains an ECS option then ACLs containing geoip or ecs elements can match against the address encoded in the option. This can be used to select a view for a query, so that different answers can be provided depending on the client network.
- The EDNS EXPIRE option has been implemented on the client side, allowing a slave server to set the expiration timer correctly when transferring zone data from another slave server.
- A new masterfile-style zone option controls the formatting of text zone files: When set to full, the zone file will dumped in single-line-per-record format.
- **dig +ednsopt** can now be used to set arbitrary EDNS options in DNS requests.
- **dig +ednsflags** can now be used to set yet-to-be-defined EDNS flags in DNS requests.
- **dig +[no]ednsnegotiation** can now be used enable / disable EDNS version negotiation.
- **dig +header-only** can now be used to send queries without a question section.
- **dig +ttlunits** causes **dig** to print TTL values with time-unit suffixes: w, d, h, m, s for weeks, days, hours, minutes, and seconds.
- **dig +zflag** can be used to set the last unassigned DNS header flag bit. This bit is normally zero.
- **dig +dscp=*value*** can now be used to set the DSCP code point in outgoing query packets.
- **dig +mapped** can now be used to determine if mapped IPv4 addresses can be used.
- **nslookup** will now look up IPv6 as well as IPv4 addresses by default. [RT #40420]
- serial-update-method can now be set to date. On update, the serial number will be set to the current date in YYYYMMDDNN format.
- **dnssec-signzone -N date** also sets the serial number to YYYYMMDDNN.
- **named -L *filename*** causes **named** to send log messages to the specified file by default instead of to the system log.

- The rate limiter configured by the `serial-query-rate` option no longer covers NOTIFY messages; those are now separately controlled by `notify-rate` and `startup-notify-rate` (the latter of which controls the rate of NOTIFY messages sent when the server is first started up or reconfigured).
- The default number of tasks and client objects available for serving lightweight resolver queries have been increased, and are now configurable via the new `lwres-tasks` and `lwres-clients` options in `named.conf`. [RT #35857]
- Log output to files can now be buffered by specifying **buffered yes;** when creating a channel.
- **delv +tcp** will exclusively use TCP when sending queries.
- **named** will now check to see whether other name server processes are running before starting up. This is implemented in two ways: 1) by refusing to start if the configured network interfaces all return "address in use", and 2) by attempting to acquire a lock on a file specified by the `lock-file` option or the **-X** command line option. The default lock file is `/var/run/named/named.lock`. Specifying `none` will disable the lock file check.
- **rndc delzone** can now be applied to zones which were configured in `named.conf`; it is no longer restricted to zones which were added by **rndc addzone**. (Note, however, that this does not edit `named.conf`; the zone must be removed from the configuration or it will return when **named** is restarted or reloaded.)
- **rndc modzone** can be used to reconfigure a zone, using similar syntax to **rndc addzone**.
- **rndc showzone** displays the current configuration for a specified zone.
- When BIND is built with the **lmdb** library (Lightning Memory-Mapped Database), **named** will store the configuration information for zones that are added via **rndc addzone** in a database, rather than in a flat "NZF" file. This dramatically improves performance for **rndc delzone** and **rndc modzone**: deleting or changing the contents of a database is much faster than rewriting a text file.

  On startup, if **named** finds an existing NZF file, it will automatically convert it to the new NZD database format.

  To view the contents of an NZD, or to convert an NZD back to an NZF file (for example, to revert back to an earlier version of BIND which did not support the NZD format), use the new command **named-nzd2nzf** [RT #39837]
- Added server-side support for pipelined TCP queries. Clients may continue sending queries via TCP while previous queries are processed in parallel. Responses are sent when they are ready, not necessarily in the order in which the queries were received.

  To revert to the former behavior for a particular client address or range of addresses, specify the address prefix in the "keep-response-order" option. To revert to the former behavior for all clients, use "keep-response-order { any; };".
- The new **mdig** command is a version of **dig** that sends multiple pipelined queries and then waits for responses, instead of sending one query and waiting the response before sending the next. [RT #38261]
- To enable better monitoring and troubleshooting of RFC 5011 trust anchor management, the new **rndc managed-keys** can be used to check status of trust anchors or to force keys to be refreshed. Also, the managed-keys data file now has easier-to-read comments. [RT #38458]
- An **--enable-querytrace** configure switch is now available to enable very verbose query trace logging. This option can only be set at compile time. This option has a negative performance impact and should be used only for debugging. [RT #37520]
- A new **tcp-only** option can be specified in **server** statements to force **named** to connect to the specified server via TCP. [RT #37800]

- The **nxdomain-redirect** option specifies a DNS namespace to use for NXDOMAIN redirection. When a recursive lookup returns NXDOMAIN, a second lookup is initiated with the specified name appended to the query name. This allows NXDOMAIN redirection data to be supplied by multiple zones configured on the server, or by recursive queries to other servers. (The older method, using a single **type redirect** zone, has better average performance but is less flexible.) [RT #37989]

- The following types have been implemented: CSYNC, NINFO, RKEY, SINK, TA, TALINK.

- A new **message-compression** option can be used to specify whether or not to use name compression when answering queries. Setting this to **no** results in larger responses, but reduces CPU consumption and may improve throughput. The default is **yes**.

- A **read-only** option is now available in the **controls** statement to grant non-destructive control channel access. In such cases, a restricted set of **rndc** commands are allowed, which can report information from **named**, but cannot reconfigure or stop the server. By default, the control channel access is *not* restricted to these read-only operations. [RT #40498]

- When loading a signed zone, **named** will now check whether an RRSIG's inception time is in the future, and if so, it will regenerate the RRSIG immediately. This helps when a system's clock needs to be reset backwards.

- The new **minimal-any** option reduces the size of answers to UDP queries for type ANY by implementing one of the strategies in "draft-ietf-dnsop-refuse-any": returning a single arbitrarily-selected RRset that matches the query name rather than returning all of the matching RRsets. Thanks to Tony Finch for the contribution. [RT #41615]

- **named** now provides feedback to the owners of zones which have trust anchors configured (**trusted-keys**, **managed-keys**, **dnssec-validation auto;** and **dnssec-lookaside auto;**) by sending a daily query which encodes the keyids of the configured trust anchors for the zone. This is controlled by **trust-anchor-telemetry** and defaults to yes.

1.6 Feature Changes

- The logging format used for **querylog** has been altered. It now includes an additional field indicating the address in memory of the client object processing the query.

  The ISC DNSSEC Lookaside Validation (DLV) service is scheduled to be disabled in 2017. A warning is now logged when **named** is configured to use this service, either explicitly or via `dnssec-lookaside auto;`. [RT #42207]

- The timers returned by the statistics channel (indicating current time, server boot time, and most recent reconfiguration time) are now reported with millisecond accuracy. [RT #40082]

- Updated the compiled-in addresses for H.ROOT-SERVERS.NET and L.ROOT-SERVERS.NET.

- ACLs containing **geoip asnum** elements were not correctly matched unless the full organization name was specified in the ACL (as in **geoip asnum "AS1234 Example, Inc.";**). They can now match against the AS number alone (as in **geoip asnum "AS1234";**).

- When using native PKCS#11 cryptography (i.e., **configure --enable-native-pkcs11**) HSM PINs of up to 256 characters can now be used.

- NXDOMAIN responses to queries of type DS are now cached separately from those for other types. This helps when using "grafted" zones of type forward, for which the parent zone does not contain a delegation, such as local top-level domains. Previously a query of type DS for such a zone could cause the zone apex to be cached as NXDOMAIN, blocking all subsequent queries. (Note: This change is only helpful when DNSSEC validation is not enabled. "Grafted" zones without a delegation in the parent are not a recommended configuration.)

- Update forwarding performance has been improved by allowing a single TCP connection to be shared between multiple updates.

- By default, **nsupdate** will now check the correctness of hostnames when adding records of type A, AAAA, MX, SOA, NS, SRV or PTR. This behavior can be disabled with **check-names no**.
- Added support for OPENPGPKEY type.
- The names of the files used to store managed keys and added zones for each view are no longer based on the SHA256 hash of the view name, except when this is necessary because the view name contains characters that would be incompatible with use as a file name. For views whose names do not contain forward slashes ('/'), backslashes ('\'), or capital letters - which could potentially cause namespace collision problems on case-insensitive filesystems - files will now be named after the view (for example, `internal.mkeys` or `external.nzf`). However, to ensure consistent behavior when upgrading, if a file using the old name format is found to exist, it will continue to be used.
- "rndc" can now return text output of arbitrary size to the caller. (Prior to this, certain commands such as "rndc tsig-list" and "rndc zonestatus" could return truncated output.)
- Errors reported when running **rndc addzone** (e.g., when a zone file cannot be loaded) have been clarified to make it easier to diagnose problems.
- When encountering an authoritative name server whose name is an alias pointing to another name, the resolver treats this as an error and skips to the next server. Previously this happened silently; now the error will be logged to the newly-created "cname" log category.
- If **named** is not configured to validate answers, then allow fallback to plain DNS on timeout even when we know the server supports EDNS. This will allow the server to potentially resolve signed queries when TCP is being blocked.
- Large inline-signing changes should be less disruptive. Signature generation is now done incrementally; the number of signatures to be generated in each quantum is controlled by "sig-signing-signatures *number*;". [RT #37927]
- The experimental SIT option (code point 65001) of BIND 9.10.0 through BIND 9.10.2 has been replaced with the COOKIE option (code point 10). It is no longer experimental, and is sent by default, by both **named** and **dig**.

  The SIT-related named.conf options have been marked as obsolete, and are otherwise ignored.
- When **dig** receives a truncated (TC=1) response or a BADCOOKIE response code from a server, it will automatically retry the query using the server COOKIE that was returned by the server in its initial response. [RT #39047]
- Retrieving the local port range from net.ipv4.ip_local_port_range on Linux is now supported.
- A new `nsip-wait-recurse` directive has been added to RPZ, specifying whether to look up unknown name server IP addresses and wait for a response before applying RPZ-NSIP rules. The default is **yes**. If set to **no**, **named** will only apply RPZ-NSIP rules to servers whose addresses are already cached. The addresses will be looked up in the background so the rule can be applied on subsequent queries. This improves performance when the cache is cold, at the cost of temporary imprecision in applying policy directives. [RT #35009]
- Within the `response-policy` option, it is now possible to configure RPZ rewrite logging on a per-zone basis using the `log` clause.
- The default preferred glue is now the address type of the transport the query was received over.
- On machines with 2 or more processors (CPU), the default value for the number of UDP listeners has been changed to the number of detected processors minus one.
- Zone transfers now use smaller message sizes to improve message compression. This results in reduced network usage.
- Added support for the AVC resource record type (Application Visibility and Control).

  Changed **rndc reconfig** behavior so that newly added zones are loaded asynchronously and the loading does not block the server.

- **minimal-responses** now takes two new arguments: `no-auth` suppresses populating the authority section but not the additional section; `no-auth-recursive` does the same but only when answering recursive queries.
- At server startup time, the queues for processing notify and zone refresh queries are now processed in LIFO rather than FIFO order, to speed up loading of newly added zones. [RT #42825]
- When answering queries of type MX or SRV, TLSA records for the target name are now included in the additional section to speed up DANE processing. [RT #42894]
- **named** can now use the TCP Fast Open mechanism on the server side, if supported by the local operating system. [RT #42866]

1.7 Bug Fixes

- Fixed a crash when calling **rndc stats** on some Windows builds: some Visual Studio compilers generate code that crashes when the "%z" printf() format specifier is used. [RT #42380]
- Windows installs were failing due to triggering UAC without the installation binary being signed.
- A change in the internal binary representation of the RBT database node structure enabled a race condition to occur (especially when BIND was built with certain compilers or optimizer settings), leading to inconsistent database state which caused random assertion failures. [RT #42380]

1.8 End of Life

The end of life for BIND 9.11 is yet to be determined but will not be before BIND 9.13.0 has been released for 6 months. https://www.isc.org/downloads/software-support-policy/

1.9 Thank You

Thank you to everyone who assisted us in making this release possible. If you would like to contribute to ISC to assist us in continuing to make quality open source software, please visit our donations page at http://www.isc.org/donate/.