OARC

Annual General Meeting

Software Report

Jerry Lundström

Sep 23, 2020

Table of Contents

This report contains all major software happenings since OARC31. If you're a frequent reader of my development update blog posts then you'll probably recognize a lot of the information here.

1 Development platforms

Besides keeping the platforms up-to-date with the latest and greatest releases of all distributions, there have also been additions to the various platforms. We now also compile and test on CentOS 8 (x86_64) and a 32bit Debian 10 (i686).

Current distributions are:

- Debian 10 (x86_64, i686)
- Ubuntu 20.04.1 (x86_64)
- CentOS 7.8.2003 (x86_64), 8.2.2004 (x86_64)
- FreeBSD 12.1-RELEASE-p9 (amd64)
- OpenBSD 6.7 (amd64)

2 Funded Projects

2.1 Soteria

As announced at OARC31, project Soteria aims at greatly improving the quality and testing of DNS-OARC's software and tools for the community, by using code coverage and code analysis, and was wrapped up just before OARC33.

This project was funded by The Swedish Internet Foundation and some 200+ hours has been spent between January and September 2020.

Code Analysis

Three different code analysis tools are being used to analyze our code.

- scan-build, part of Clang, runs on every pull-request
- Looks Good to Me (LGTM) integrates via their GitHub app and runs on every pull-request
- SonarCloud runs and analyses the develop branch on regular basis

While SonarCloud has a GitHub app, C/C++ projects are not fully supported by it so I run a scheduled job on the main buildbot platform to analyze the code using sonar-scanner.

As code analysis has been enabled for each of our software repositories I've also handled all the issues that havebeen reported.

On LGTM I've fixed 27 alerts and removed about 110 false-positive or alerts on software that are not OARC's. On SonarCloud the numbers are a bit higher, 112 fixed issues and over 145 false-positives removed.

scan-build was running on pull-requests before this project started so I have no numbers on that, but there's been quite a few and what's positive about running this many different analysis tools are that they catch different things.

Code Coverage

For code coverage I first looked at what kind of reporting there was and was happily surprised at how good SonarCloud's free integration with GNU GCC's coverage (gcov) looked.

It didn't that long at all to get it working with C and C++ projects and after a bit of tinkering I had also added it to our Python projects.

Now adding coverage testing is one thing, having coverage is a whole other. Once gcov was added to the majority of our projects I could see that the (grim) overall coverage was 30%. That is indeed bad, but not unexpected since a lot of the projects never had any tests to being with.

Thanks to how gcov works, by adding code that reports function and branch usage, the overall coverage was actually higher then I thought it would be. And it also made it very easy to increase it!

As this was the last phase of the Soteria project, I decided to aim for 80% and use up all the remaining time in the project, but if I spent too long trying to get a few more percent I would skip that project and more to the next (the 80-20 or 70-30 rule).

Here is the initial gcov result vs the end of Soteria:

- tinyframe: 66% => 88%
- packetq: 64%
- dsc-datatool: 0% => 82%
- dsc: 30% => 70%
- dnswire 46% => 73%
- dnsperf 0% => 81%
- dnsjit 23%
- dnscap 41% => 71%
- dnsmeter 0% => 6%

As shown, some of the projects managed to get to 80%+ but for dsc, dnswire and dnscap, I ran into a wall when it came to testing code that captures traffic or in other ways uses the network. Time also ran out for doing anything with PacketQ, dnsjit or dnsmeter.

At the end of Soteria the overall coverage went from 30% to 62%!

Future Work

Now that the code analysis and coverage reporting is in place, it easy to follow how future development affects each project, and once SonarCloud can be integrated into pull-request then we can easily say that new code must have a certain coverage to be included. Our code is in a much better state now thanks to the Soteria project!

2.2 dsc-datatool rewrite

Thanks to funded development by EURid, the dsc-datatool has been rewritten from Perl to Python in order to be easier to maintain. The choice of Python instead of Perl was made because a lot more of our Members seem to use Python then Perl, and a few new projects (both internal and external) are in Python unless they are in C.

Read more about the rewrite and DSC history in my APNIC blog post that was recently published.

3 Feature Highlights

3.1 Check My DNS + RPKI origin validation

Check My DNS can now check if RPKI origin validation is enabled for the resolvers that query it, but as there is no UI part for this yet you will need to download cmdns-cli and the specific check.

```
cmdns-cli -checks trans_rpkiv4
```

Hints; Use *trans_rpkiv6* for IPv6 check and *-res <IP>:<port>* to use a different resolver then the system default.

Read the full development story in my blog post about this.

3.2 DNSTAP support in DSC

As of version 2.9.0 of DSC, DNSTAP is now supported! This version uses the new libraries dnswire and tinyframe, and supports reading DNSTAP from a file or over an UNIX socket, UDP or TCP connection.

DNSTAP allows DSC to receive DNS messages directly from the DNS server / daemon / software itself instead of capturing them. This will make it possible to gather statistics on more and other kinds of transactions, for example encrypted DNS transmissions.

NOTE: the DNSTAP support is currently Work in Progress (WIP), but has been tested to work with bind9 and unbound. Please reach out to me if you have any issues and I would love to hear if anyone will be testing this!

Read more about DNSTAP support in my Development Update #2003.

4 Software Updates

A key part of DNS-OARC's mission is to develop, maintain and host various software tools for DNS data collection, measurement and analysis. OARC can develop new, or enhance features of existing, tools via a custom for-hire development contract. OARC Members will receive priority for such work, and at a discounted rate depending on their membership tier.

You can find a list of all our software and information about funded development here:

https://www.dns-oarc.net/oarc/software

4.1 *new* tinyframe

tinyframe is a minimalistic library for reading and writing the Frame Streams protocol which is used to encapsulate DNSTAP.

This library is currently a Work in Progress which means backwards incompatible changes can be made. Once version 1.0.0 is released it will follow Semantic versioning 2.0 as all other software does.

v0.1.0

This marks the first release of tinyframe.

4.2 *new* dnswire

A C library for encoding/decoding different DNS encapsulations and transporting them over different protocols. It currently supports DNSTAP using Protobuf sent over Frame Streams using tinyframe.

This library is currently a Work in Progress which means backwards incompatible changes can be made. Once version 1.0.0 is released it will follow Semantic versioning 2.0 as all other software does.

v0.1.0

This marks the first release of dnswire.

v0.1.1

Fixed RPM package dependencies.

4.3 dsc

DNS Statistics Collector (dsc) is a tool used for collecting and exploring statistics from busy DNS servers. It uses a distributed architecture with collectors running on or near name-servers sending their data to one or more central systems for display and archiving.

v2.9.0

Added support for receiving DNS messages over DNSTAP along with documentation updates and eliminated compiler warnings.

To enable DNSTAP support, install dependencies and run configure with --enable-dnstap or use our pre-built distribution packages.

New configuration options:

- dnstap_file: specify input from DNSTAP file
- dnstap_unixsock: specify DNSTAP input from UNIX socket
- dnstap_tcp: specify DNSTAP input from TCP connections (dsc listens)
- dnstap_udp: specify DNSTAP input from UDP connections (dsc listens)

- dnstap_network: specify network information in place of missing DNSTAP attributes

Other changes:

- Add documentation about extra configure options that might be needed for FreeBSD/OpenBSD
- Fix compile warnings on FreeBSD 11.2
- Fix compile warning snprintf() truncation
- Packaging updates

v2.9.1

Fixed a few bugs, removed a lot of the debug messages about DNSTAP and removed GeoIP from openSUSE/SLE packages as it has been deprecated on those platforms.

Changes:

- daemon: Fix bug with listening for SIGINT when in foreground mode
- dnstap:
  - Fix #217: Unlink UNIX socket on exit if successfully initiated
  - Fix startup bug, exit() if unable to initialize
  - Fix #220:
    - Remove/hide a lot of debug messages and the printing of the DNSTAP message
    - Clarify a lot of the info and error messages
    - Prefix all DNSTAP related messages with DNSTAP:
- Fix compile warnings and include headers when GeoIP is missing
- asn_indexer: Fix bug, said unknown IPv4 when it was IPv6

v2.10.0

Added new configuration options to dnstap_unixsock, to control ownership and permissions for the DNSTAP socket file.

Other fixes:

- Unlink the DNSTAP socket file if an error during initialization occur
- Do hard exit in forks to not run atexit() (which will unlink the DNSTAP socket file)

v2.11.0

Updated the built in known TLDs table and added the optional configuration option knowntlds_file instead of using the built in table, load the data from a file.

Fixed an issue where if compiled with only MaxMindDB support then ASN and Country indexer would complain (and exit) that no database has been specified.

Other changes:

- Fix compile warnings
- COPR packaging fixes
- country_indexer: Fixed typos in log messages (was copied from ASN)
- Fix issues and false-positives reported by newer version of scan-build

v2.11.1

Fixed a 17-year old code cut&paste mistake in the classification indexer, until now it's been classifying funny query types based on the query class. Thanks to Jim Hague (Sinodun) for sending in this patch.

Other changes are based on code analysis reports and setup for code coverage.

4.4 dsc-datatool

dsc-datatool is a tool for converting, exporting, merging and transforming dsc data using a plugin architecture. It can be used to convert dsc XML data into InfluxDB which can be used by Grafana to display DNS statistics.

v1.0.0

This release brings a complete rewrite of the tool, from Perl to Python. This rewrite was made possible thanks to funding from EURid, and will help with maintainability and packaging.

Core design and command line syntax is kept the same but as the libraries the generators use have been changed additional command line options must be used.

client_subnet_authority (generator)

> This generator now uses IANA's IP address space registry CSVs to look up the network authority, therefor it needs either to fetch the CSV files or be given them on command line. See man dsc-datatool-generator client_subnet_authority for more information.

client_subnet_country (generator)

> This generator now uses MaxMind databases to look up country based on subnet. See man dsc-datatool generator client_subnet_country for more information and setup guide of the MaxMind databases.

v1.0.1

Added compatibility with Python v3.5 which allows packages to be built for Ubuntu Xenial.

4.5 dnsperf

dnsperf and resperf (part of dnsperf) are tools that makes it simple to gather accurate latency and throughput metrics for DNS services. These tools are easy-to-use and can simulate typical Internet usage, so network operators can benchmark their naming and addressing infrastructure and plan for upgrades.

v2.3.3

Changed the behavior of dnsperf and resperf when it comes to TCP and TLS connection resets or connection closed. They are now treated as "try again" so that the run is finished and not aborted.

As SIGPIPE might be received on usage of closed connections, it's now blocked in dnsperf and handled as a fatal action in resperf.

Updated package building using COPR, patch from Petr Menšík (Red Hat).

v2.3.4

Added a workaround, thanks to patch from Petr Menšík, for building on systems with BIND 9.16.

Improved error handling by using thread-safe strerror_r() instead of strerror().

4.6 dnscap

dnscap is a network capture utility designed specifically for DNS traffic. It produces binary data in pcap(3) and other formats. This utility is similar to tcpdump(1), but has a number of features tailored to DNS transactions and protocol options. DNS-OARC uses dnscap for DITL data collections.

v1.10.4

Fixed a bug that would not drop privileges when not specifying any interface (which is equal to capturing on all interfaces).

Added functionality to set the supplemental groups when dropping privileges and changing user, or clear them if that is not supported.

Other changes includes corrected man-page about `-w` and update to documentation.

v1.11.0

Added a new plugin called eventlog, contributed by Byron Darrah, to output DNS activity as log events, including answers to A and AAAA queries.

Other changes includes compile warning and code analysis fixes.

v1.11.1

Fixed a lot of issues found by code analysis, added an explicit memory zeroing function to remove account information (when dropping privileges) and added code coverage reporting.

The dnscap_memzero() will use explicit_bzero() on FreeBSD and OpenBSD, or memset_s() (if supported), otherwise it will manually set the memory to zero. This will hopefully ensure that the memory is zeroed as compilers can optimize out memset()'s that are just before free().

The plugins exit code for the help option -? has been changed to 0 to have the same as dnscap -?.

4.7 dnsjit

dnsjit is a combination of parts taken from dsc, dnscap, drool, and put together around Lua to create a script-based engine for easy capturing, parsing and statistics gathering of DNS messages while also providing facilities for replaying DNS traffic.

v1.0.0

dnsjit has finally crawled out of the alpha/beta sewers with the v1.0.0 release!

Most of the changes have been to the output dnssim which is the heart of DNS shotgun – a realistic DNS benchmarking tool that supports multiple transport protocols and can simulate hundreds of thousands of clients.

Future Releases

Now that v1.0.0 is released, dnsjit will follow strict Semantic Versioning 2.0 as all other projects.

This will mean that whatever script or tool that is written for major version 1 will work on all major version 1's. New features and capabilities may be added through minor version releases and if there is dependency on that then the script or tool needs to check the minor version of dnsjit.

To make it easier for modules, such as dnssim, to operate under its own versioning, I've been wanting to implement support for easy integration of dynamic libraries into dnsjit but have been lacking time.

I've also been thinking about trying out other forks or variants of LuaJIT, such as RaptorJIT and moonjit, as the development of LuaJIT is a bit stale.

Hopefully some of this can be planned for in 2021.

4.8 PacketQ

packetq is a command line tool to run SQL queries directly on PCAP files, the results can be outputted as JSON (default), formatted/compact CSV and XML.

v1.4.2

Updated the built in list of DNS resource types and made changes to testing and packaging.

5 OARC Portal Updates

Our self-service portal has seen some updates over the last year, mainly bug-fixes but also a couple of UI improvements.

In the Contact Directory service you can now get a list of all contacts in Portal and filter/search them by name or email. For the list of organization the filter/search now also applies on the shorter form of the organization name.

We have hopefully clarified the organizational role of managing the contacts by changing the role's title from *Organization Administrator* to *Organization Manager*. We've also fixed a bug in the calculations of number of used contacts with regards to contacts marked as *Administrative Only Contact*.

Reminder; An *Administrative Only Contact* is a contact that does not count towards the contact limit for your organization, but has no access to services. It can be used to free up contact slots with service access for those that needs them. You can request that contacts be changed to an *Administrative Only Contact* by emailing us.

6 Member software updates

6.1 PROXYv2 support in PowerDNS software

To facilitate exchanging of network and other information between a DNS client and a DNS server PowerDNS has implemented the ProxyV2 protocol, starting with dnsdist 1.5 for the outgoing communication direction and PowerDNS Recursor 4.4 for the incoming channel. We encourage other vendors to do likewise.

- Peter van Dijk (PowerDNS)

6.2 DNS shotgun

DNS Shotgun is a realistic DNS benchmarking toolchain utilizing dnsjit. It supports UDP, TCP, DNS-over-TLS and DNS-over-HTTPS over IPv6. DNS Shotgun is capable of simulating hundreds of thousands of clients. Every client establishes its own connection when communicating over TCP-based protocol. This makes the tool uniquely suited for realistic benchmarking since its traffic patterns are very similar to real clients.

The tool produces no synthetic queries. Instead, real captured DNS packets are used along with their original timing to ensure the most accurate simulation of client behavior. However, this means a large amount of captured DNS traffic is necessary in order to properly use the tool.

The first released version (v20200914) has a basic UI that allows sending 100% of traffic over UDP, TCP, DoT or DoH. Future versions will support more complex configurations that will allow mixed protocol usage (e.g. sending 50 % of traffic over UDP, 25 % over DoT and 25 % over DoH).

- Tomas Krizek (CZ.NIC)

6.3 Flamethrower

Flamethrower is a small, fast, configurable tool for functional testing, benchmarking, and stress testing DNS servers and networks. It supports IPv4, IPv6, UDP, TCP, DoT, and DoH and has a modular system for generating queries used in the tests.

The recent v0.11.0 release includes new support for DNS over HTTPS (DoH). DNS over TLS was renamed from "tcptls" to "dot" in the command line options and documentation. There were also several bug fixes covering the rate limiter, the build, and the docker image.

There is now an official docker image located on Docker Hub.

- Shannon Weyrick (NS1)

6.4 pktvisor

NS1 announces a newly open sourced visibility tool: pktvisor!

It summarizes data streams in real time and provides a clean, time-windowed HTTP interface (for centralized collection) and command line UI to the results. Currently DNS focused, it is designed to eventually be used in broader contexts.

Summarized information includes, for example:

- Packet rates: 50th, 90th, 95th, 99th percentiles
- Packet counts by protocol and IP version
- Top 10 heavy hitters: IPs, ASNs, Geo, DNS qnames, slow transactions, result codes, query types, and more
- Cardinality of set of source IPs and DNS qnames seen in time window

- *Shannon Weyrick (NS1)*