OARC30

Software Report

Jerry Lundström

May 6, 2019

Table of Contents

1 Development platforms

The development server (sponsored by Netnod) runs a bunch of VMs to do continuous building and testing which is managed by Buildbot since September last year. Buildbot is really great(!) and it was a huge improvement from running Jenkins.

https://dev.dns-oarc.net/

Our platforms, which we keep to the latest stable/LTS release, are:

- Debian 9.9
- Ubuntu 18.04.2
- CentOS 7.6.1810
- FreeBSD 12.0-RELEASE-p3
- OpenBSD 6.4 (pending update to 6.5)

1.1 Continuous Integration

During the last half year I've had more and more problems with Coverity Scan which we have used for code analysis (the free for open source). In December the problems was just too much, the build queue just stopped or was moving at a pace that would take months.

So I started to look for alternatives and found two cloud based ones, LGTM and sonarcloud, which have been set up with all our repositories. But they still need a lot of work before being reliable, for example there are always false-positives at first and repository integration needs more work. I also found clang's `scan-build` tool which I incorporated into Buildbot to run on all builds.

1.2 Packaging

In the pipe-line is to look at automatic package building for all our distributions which would help immensely with things like keeping in-sync with distributions versions and some tricky dependency chains. We use the official build platforms (LaunchPad, COPR, openSUSE Build Service) for most packages, Debian is built using cowbuilder/pbuilder in a VM.

https://pkg.dns-oarc.net

Today we build packages for:

- Debian: Stretch (stable), Buster (testing), Sid (unstable)
- Ubuntu: Xenial (16.04), Bionic (18.04), Cosmic (18.10), Disco (19.04)
- CentOS/RHEL: EPEL 7
- Fedora: 28, 29, 30, rawhide
- SUSE Linux Enterprise: 12 SP3, 12 SP4, 15
- openSUSE: Leap 15.0, Leap 42.2, Leap 42.3, Tumbleweed

2 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

2.1 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.

dnsperf found a new home this year at DNS-OARC and we are excited to be continuing to maintain it. Our thanks to Nominum/Akamai for the many years of developing dnsperf.

First release by DNS-OARC (v2.2.0) came with a rework of the code to use autotools, semantic versioning and bug-fixes pulled from other's forks:

- Fix infinite loop in argument parsing
- Fix min/max latency summing for multi-threaded runs
- Fix calculation of per_thread socket counts
- Fixes to queryparse along with python3 support
- Fix various compilation issues
- Fix issues found by code analysis tools

Shortly after the release a few issues got reported (and fixed) by Vladimír Čunát (CZ.NIC) and release v2.2.1 was made. These issues had to do with the usage of ISC internal types.

I don't know the full story behind dnsperf but it looks like it was created with parts from Bind which now makes it depend on internal headers and libraries. These headers and libraries will change, or even be removed, in the future so I've started to look at how we can move away from this dependency.

TCP and TLS support is coming, hopefully soon, to dnsperf. This was initially pulled from Sinodun's repository but after looking over the code I found a better way to add additional transportation into dnsperf which makes future additions easier.

2.2 dnscap

dnscap is a network capture utility designed specifically for DNS traffic. It produces binary data in pcap(3) and other format. 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.

Thanks to funding from Verisign there are now 5 new plugins for dnscap (release v1.10.0) that do various IP anonymization/deanonymization which we hope will help our members comply with privacy requirements.

The methods to do (pseudo-)anonymization have been taken from the RSSAC040 "Major Proposals for Methods of Anonymizing IP Addresses" and include:

- **anonaes128**

  Anonymize IP addresses by encrypting them with AES128 (RSSAC040 4.1/4.3).

  Since AES128 works on 128 bit blocks the IPv4 addresses (32 bits) are padded by copying itself to fill the 128 bits (IPv4*4) and then the output is truncated to 32 bits which means that it can't be deanonymized. No modifications are needed for IPv6 since the output length is the same. Thanks to help from Jim Hague (Sinodun) we have successfully tested interoperability with anonymization features of compactor/inspector and this plugin.

- **anonmask**

  Pseudo-anonymize IP addresses by masking them as you do with netmasks (RSSAC040 4.4).

  The default is a /24 for IPv4 and /48 for IPv6 but it can be changed by command line options to the plugin.

- **cryptopan**

  Anonymize IP addresses using an extension to Crypto-PAn (College of Computing, Georgia Tech) made by David Stott (Lucent) (RSSAC040 4.2).

  The extension was picked instead of the reference implementation because it provided a deanonymization function, handled endian and hopefully gives better randomness in the resulting anonymized addresses.

- **cryptopant**

  Anonymize IP addresses using the library cryptopANT, a different implementation of Crypto-PAn, made by the ANT project at USC/ISI (RSSAC040 4.2).

- **ipcrypt**

  Anonymize IP addresses using ipcrypt create by Jean-Philippe Aumasson (RSSAC040 4.3).

  Although the method was designed for IPv4 addresses, the plugin can handle IPv6 addresses too. It does this with a command line option, treating IPv6 addresses as four IPv4 addresses, encrypting/decrypting them separately.

2.3 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 nameservers sending their data to one or more central presenters for display and archiving.

The v2.8.0 release brings a new indexer `response_time` (funded by NIC.AT) which can track queries and report the time it took to receive the response. Support for MaxMind DB (GeoIP2) was also added since their old GeoIP library and data files are being deprecated.

The new indexer `response_time` can report the response time in a couple of different ways, first as logarithmic scales (log2 and log10) and as configurable buckets of microseconds. It will also report timeouts, missing queries (received a response but have never seen the query), dropped queries (due to memory limitations) and internal errors.

Example of log10 output:

```
    <array name="response_time" dimensions="2"
start_time="1478727151"
        stop_time="1478727180">
      <dimension number="1" type="All"/>
      <dimension number="2" type="ResponseTime"/>
      <data>
        <All val="ALL">
          <ResponseTime val="100000-1000000" count="77"/>
          <ResponseTime val="10000-100000" count="42"/>
          <ResponseTime val="1000-10000" count="3"/>
          <ResponseTime val="missing_queries" count="1"/>
        </All>
      </data>
    </array>
```

Sadly I somehow totally missed to add the glue for the configuration of the response time indexer in v2.8.0, this was fixed in v2.8.1.

New configuration options in v2.8.0/v2.8.1:

- asn_indexer_backend: Control what backend to use for the ASN indexer
- country_indexer_backend: Control what backend to use for the country indexer
- maxminddb_asn: Specify database for ASN lookups using MaxMind DB
- maxminddb_country: Specify database for country lookups using MaxMind DB
- dns_port: Control the DNS port
- response_time_mode: Set the output mode of the response time indexer
- response_time_bucket_size: The size of bucket (microseconds)
- Following options exists to control internal aspects of response_time indexer, see man-page for more information:
    - response_time_max_queries
    - response_time_full_mode
    - response_time_max_seconds
    - response_time_max_sec_mode

2.4 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.

The v0.04 release fixed a package dependency problem and has updated Grafana example dashboards which should now show correct numbers when zooming in and out.

We have a test instance of dsc → dsc-datatool → Grafana that has been running since 2016 with data going back to early 2017:

https://dev.dns-oarc.net/dsc-grafana2/d/000000012/dsc

If you want to test this yourself we have a wiki about how to set it up:

https://github.com/DNS-OARC/dsc-datatool/wiki/Setting-up-a-test-Grafana

And if you're using `dsc-datatool` and Grafana for dsc data I would love to hear about it, even if you don't have anything to report.

2.5 dnsjit + drool

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.

The develop branch of drool (DNS Replay Tool) is now written in Lua and can replay DNS traffic from packet capture (PCAP) files and send it to a specified server, with options such as to manipulate the timing between packets, as well as loop packets infinitely or for a set number of iterations.

Release v0.9.7 of dnsjit brought fixes to the TCP output `dnsjit.output.tcpcli` and a new TLS output `dnsjit.output.tlscli`. In the develop branch there is a new DNS output `dnsjit.output.dnscli` which can talk DNS over UDP, TCP and TLS. There is also a new example program `replay_multicli.lua` that uses this new output to open up multiple connections to the target, simulating multiple clients.

Release v1.99.4 of drool fixes the `--timing` option which previously only worked with `keep`.

Feel free to check out all the example programs in the dnsjit repository to learn more about how this "script engine" works:

https://github.com/DNS-OARC/dnsjit/tree/develop/examples

3 OARC portal v2

The new portal beta instance is live! And we encourage everyone to try it out and give us feedback on it. It's been completely rebuilt from scratch and is now using the microframework Flask for the server side logic and bootstrap for the graphical web interface.

https://dev.dns-oarc.net/portal2-beta

The data model of the new portal is a bit different compared to its previous version, to future proof and support non-member "members" we have reworked "members" into organizations. Each organization has a tier, this was previously the member level. Tiers belong to a category and to begin with we will have one category, "Member".

For example in the previous portal a member with the level of Silver would now be an organization in the Member category and with the Silver tier (specific to the Member category).

There have also been changes made to the data model within an organization, instead of users there are now contacts and the strong connection between portal accounts and other system accounts has been removed.

This is because we are introducing a new thing; services!

There is now a modular layer that allows us to create services and for our first release we will have a member mailing list service and a jabber service, since these features already existed. There is also two new services, an OARC Board service to list all board members and a Contact Directory to list all OARC member organizations and their contacts!

Another highlight is the new way to reset your password, gone is the requirement to have a PGP key uploaded. You'll now request a password reset and will get an email with instructions how to proceed.

Target date for launching the new portal is the end of May.