April 2024
Geoff Huston

DNSSEC and .nz

I had the opportunity to participate in the New Zealand Network Operators Group meeting (NZNOG) in Nelson earlier this month. This article was prompted by a presentation from Josh Simpson on an .nz service outage incident in May 2023.

I guess we've become used to reading evasive and vague outage reports that talk about "operational anomalies" causing "service interruptions" that are "being rectified by our team of support engineers as we speak". When we see a report that details the issues and the remedial measures that have been taken in response, it sticks out as a welcome deviation from the mean. It's as if any admission of the details of a fault in the service exposes the provider to some form of ill-defined liability or reputation damage, and to minimise this exposure the reports of faults, root causes and mediation actions are all phrased in terms of vague and meaningless generalities.

Other industries have got over this overly defensive stance, albeit in some cases with a little outside assistance. The airline industry is a good case in point where the intent of such investigations is not to attribute blame and determine liability, but to determine the causes of the incident and understand how such circumstances can be avoided in the future, all because of the obvious overarching safety concerns. Other industries, including the automobile industry, the nuclear power industry, and the chemical industry have all been taught the sometimes-painful lesson that the path to a safer service and safer products necessarily involves an open, dispassionate, and honest investigation into incidents with the service. Incidents are an opportunity to learn why a system fails, and an honest and comprehensive post-event analysis can offer invaluable pointers as to what measures can be taken to avoid similar failure modes in the future. It allows all service providers to operate a safer service.

Yet despite the thorough disclosure practices that have been adopted in other industries, the information technology industry all too often regards itself as "special". For decades software vendors have been able to sell faulty and insecure product without even a hint of liability, and the effort to improve the robustness of their products was often seen as an avoidable cost to the software vendor. This attitude is still pervasive in this industry and manifests itself in outages on the Internet with depressing regularity. "Move fast and break things" became a pervasive mantra of the Internet, and not only did Meta's Mark Zuckerberg adopt this as the operating principle for their internal engineering efforts some years ago, but he went further to observe that "unless you are breaking stuff, you are not moving fast enough." Perhaps we should simply be grateful that Meta does not build aeroplanes, nuclear power plants or automobiles. But this mantra of rapid and at times somewhat careless innovation isn't unique to Meta, and has been applicable equally to many others, including Amazon, Apple, and Alphabet, who have all been moving quickly and doubtless they all have been breaking a few things along the way!

So, it's a welcome sight to see a careful and thoughtful analysis of a service outage. One such instance was a presentation by .nz's Josh Simpson at the recent NZNOG meeting, reporting on a service outage for .nz domains.

New Zealand's network infrastructure has been a leading adopter of DNSSEC validation starting from 2016, and currently some 84% of the country's users sit behind DNSSEC-validating DNS resolvers. They will not be able to resolve a DNS name if the name is signed with an invalid signature. That's a big result and well above the internet-wide average of 31% (Figure 1).

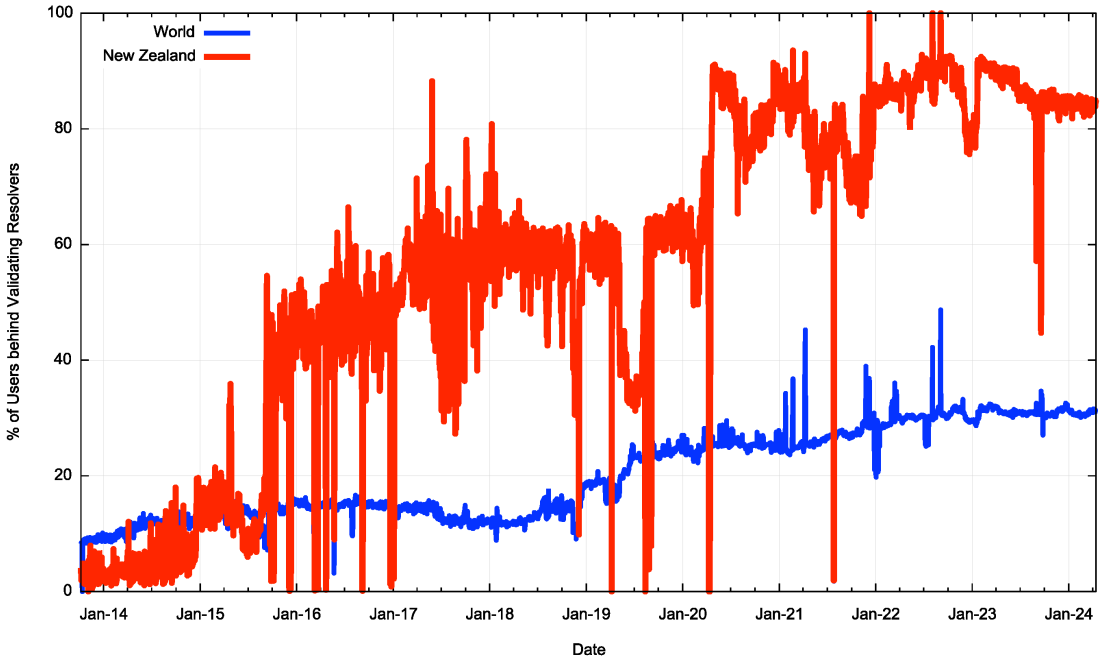

Figure 1 – New Zealand and World DNSSEC Validation rates (from https://stats.labs.apnic.net/dnssec)

However, this adoption of DNSSEC validation level is not without its risks, and while users who sit behind DNSSEC-validating recursive resolvers will not accept DNS responses that show evidence of tampering, they will also fail to resolve DNS names when the DNSSEC credentials on any label within a DNS name are incorrectly configured. DNSSEC is unforgiving in this respect.

DNSSEC is perhaps even more unforgiving that other security technologies. For example, when there is a routing mishap the problem can be rectified and service can be restored immediately. When there is a DNSSEC mishap the problem can persist in local caches across the Internet, and an operator simply needs to be patient to let the cached information expire before the issue can be corrected and the service restored. The longer the cache time to live (TTL) of the DNSSEC data, the more patience the operator needs to have! This was the reason why a DNSSEC error in Slack's DNS had a 24 hour impact for Slack's customers in September 2021 (https://slack.engineering/what-happened-during-slacks-dnssec-rollout/).

As I understand the issue for the **.nz** domain, the .**nz** registry operator is within a protracted process of transitioning from its in-house registry platform (SRS) to a new InternetNZ Registry System (IRS), based on software provided by the Canadian Internet Registry Authority (CIRA). It is an entirely new platform, with its own set of servers, operating system, racks, and networks. From the outside it looks a lot like the operational platform for .**nz** is in a transition to an entirely new DNS service provider.

Transitioning between DNS service providers is never easy, and more challenging by an order of magnitude if the zones are DNSSEC-signed. It's a case of juggling "old" and "new" data and being mindful of cached information in various resolvers. The general approach to transition is to introduce the "new" information alongside the "old" information, and then wait for at least a TTL interval to ensure that the new information has been loaded into resolvers who have actively cached the "old" data. At that point the authority point can be shifted from "old" to "new" and after waiting for another TTL interval, the "old" information can be flushed.

But there is a low more to it when the zone is DNSSEC-signed, and RFC 8901 has some advice about Multi-Signer DNSSEC models and the task of key management. There are two models described in this document, where "Model 1" uses a single Key-Signing Key (KSK), and each zone operator runs their own Zone-Signing Key (ZSK). When a zone operator wants to roll their local ZSK, then they pass the public part of the ZSK to this single zone administrator who adds it to the zone's DNSKEY record set, and signs it with the KSK. Both operators the incorporate this new DNSKEY record into their own signed zone (Figure 2).

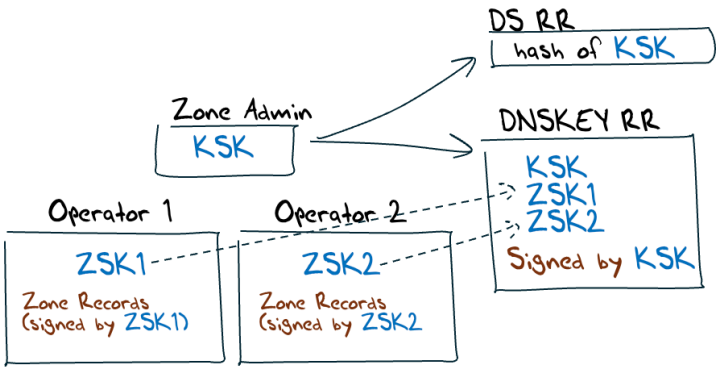

Figure 2 – RFC8901 Model 1

The RFC8901 "Model 2" approach uses distinct KSKs and ZSKs from each operator, which, in theory, can be managed separately. The distinct KSK values mean that at the parent zone there are now multiple DS records, one for each operator's KSK. This allows each operator to function more or less independently. The only residual interaction is for each operator to pull the KSK and ZSK value from the other (Figure 3).

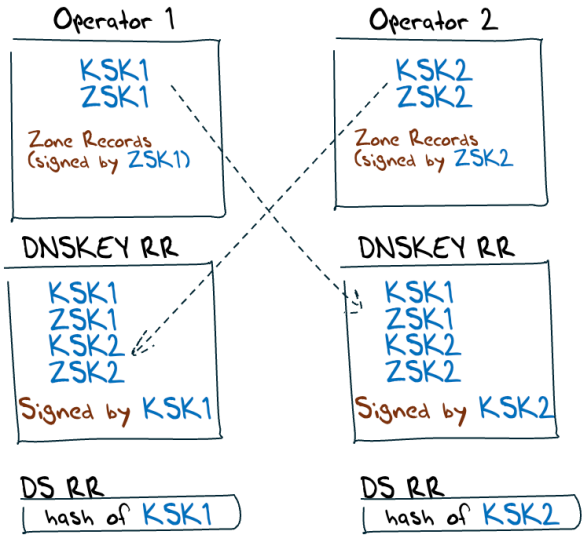

Figure 3 – RFC8901 Model 2

This separated model of multiple DNS zone operators (Model 2) is not a common observed approach in the root zone of the DNS. Of the 1,342 signed top-level domains (tlds), just 155 tlds have 2 or more DS records.

There are two tlds with 5 DS records, and 4 each with 4 and 3 DS records:

```
tld           DS records
xn--rhqv96g.     5
xn--nyqy26a.     5
icu.             4
beats.           4
bd.              4
apple.           4
xn--mgbx4cd0ab.  3
rest.            3
nz.              3
bar.             3
```

So, at this point the .**nz** registry is operating two independent registry systems, and doing so in a manner that each registry system has its own KSK and ZSK. The point of commonality is the zone's DNSKEY record, where each system's KSK and ZSK are listed:

DNSKEY Record

```
nz. 3475 IN   DNSKEY 256 3 8 AwEAAbu…
nz. 3475 IN   DNSKEY 256 3 8 AwEAAcr…
nz. 3475 IN   DNSKEY 257 3 8 AwEAAaX…
nz. 3475 IN   DNSKEY 257 3 8 AwEAAcH…
nz. 3475 IN   RRSIG  DNSKEY 8 1 3600 20240422091537 20240408051102 63529 nz. …
```

DS Record

```
nz. 86400 IN DS     49157 8 2 44628…
nz. 86400 IN DS     13646 8 2 569B1…
nz. 86400 IN DS     63529 8 2 65C96…
nz. 86400 IN RRSIG  DS    8 1 86400 20240429050000 20240416040000 5613 . LkI/Rw…
```

The Incident

The operational issue that occurred in May 2023 occurred due to the way the old system managed its DS record. The DS, or delegation signer record, is the hash of the delegated zone's KSK, and this record is published in the parent zone, signed by the parent zone's ZSK.

At the time of the incident this DS record was generated with a one hour TTL by the SRS platform (the "old" platform). The IRS platform (the "new" registry platform) used the zone's default TTL to generate the DS record, which was 1 day.

It's not clear from the presentation of this incident if the SRS platform aligned the TTLs of the DNSKEY and DS resource records, and set them both at one hour, or not. It would make a lot of sense if these two TTLs were the same, as the state you would really like to avoid occurs during a key rollover where a different cache expiration time causes one of these two resource records to expire and the refreshed value reflects a new key, while the other unexpired record in the cache still reflects the old key. At this point DNSSEC validation will fail.

But this was not the problem here, as far as I can see. Here the issue was that the new IRS system applied a one day TTL to all records, including the DS record, which was a hash of the new system's KSK. The old, and still operational, SRS system attached a one-hour TTL to the DS record of the old system's KSK.

If that was all there was to this, namely that there was different TTLs associated with the two distinct instances of the DS record, one record with a one-hour TTL and the other record with a one-day TTL, again this would normally not generate a validation failure scenario. RFC 2181, Section 5.2 says: "Should an authoritative source send such a malformed RRSet [where the component records have different

TTLs], the client should treat the RRs for all purposes as if all TTLs in the RRSet had been set to the value of the lowest TTL in the RRSet. In no case may a server send an RRSet with TTLs not all equal." So, even if the servers for **.nz** are faithfully (but incorrectly in terms of this RFC) reflecting the different TTLs for these DS records, recursive resolver clients should behave as if all the DS records had a one-hour TTL and act accordingly.

So, this too is not the problem here, as far as I can tell.

The issue appears to be a little more insidious than that and appears to be the result of an interaction with their DNSSEC key management tool, OpenDNSSEC, and this somewhat unique situation of differing TTLs for the two DS records.

To quote from a subsequent report on the incident:

> "The OpenDNSSEC key rollover tool recognised that multiple hours had past since the updated "DS" record for "ac.nz" had been seen, much longer than its configured time to wait after the record update had been seen, and it proceeded on to the next phase of the "ac.nz" KSK key rollover, which was to automatically stop using the old KSK key for signing the ZSK (zone signing key). From the next zone publication run, only the new KSK was used to sign the ZSK, which was in turn signing "ac.nz" records. This meant that the DNSSEC "trust chain" to reach the signed records in "ac.nz" now exclusively relied on trusting the new "ac.nz" KSK."
> (https://internetnz.nz/assets/Archives/External-report-on-nz-DNSSEC-chain-validation-incident-on-May-2023.pdf)

In essence it was OpenDNSSEC that removed the old SRS system's KSK signature from the zone's DNSKEY record, and also apparently removed the old system's DS record from the parent zone. The result was that any recursive resolver that was using a cached value of the old system's ZSK as part of its chain of trust was unable to validate it as the short (one hour) TTL meant that the resolver needed to reload the old KSK's DS record, but this record was no longer in the published zone.

Oops.

So, this appears to be an outcome of OpenDNSSEC behaviour, where OpenDNSSEC made an incorrect assumption about the intended key state for the zone and stopped using the old KSK to sign the common DNSKEY record, thereby breaking the chain of trust for all DNSSEC clients who are using the old KSK/ZSK keys.

Observations

The choice of TTL values is always challenging in the DNS. Short values have the advantage of allowing timely changes to the zone, which can mitigate the effects of various misconfigurations in the zone, including DNSSEC. Longer values reduce the dependency levels on the availability of the authoritative server set and can improve the performance of name resolution. However, never ever use different TTLs for different instances of the same resource record type in a zone!

Multiple zone operators of a single zone is always going to be messy. It's not an easy decision as to which model to use. Multiple independent operators each publishing a complete zone including their own KSK (Model 2 in RFC8901) requires very close coordination of the DNSKEY and DS resource records, despite the supposed independence of each operator. When issues happen in this scenario the zone administrator is left in the same position, namely that any form of rectification of the issue requires all the cached data to be flushed, so the TTL choice is important.

Automating DNSSEC is still a long way from where it should be, and in this particular case it is unfortunate that a DNSSEC management tool managed to get itself confused about the underlying key state and strip out essential key information from the zone. OpenDNSSEC is a relatively old tool and

appears to pre-date the interest in using multiple zone operators as a means of securing greater robustness for the service. Its therefore unsurprising that the tool does not appear to accommodate the scenario of multiple independent zone operators in a robust manner.

A report of the .**nz** incident of May 2023 can be found at: https://internetnz.nz/assets/Archives/External-report-on-nz-DNSSEC-chain-validation-incident-on-May-2023.pdf.

Disclaimer

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre.

Author

Geoff Huston AM, B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region.

www.potaroo.net