ICANN71 | Virtual Policy Forum – Tech Day (2 of 2)
Monday, June 14, 2021 – 16:30 to 17:30 CEST

KIM CARLSON: Hello, and welcome back to Tech Day at ICANN71, part 2. My name is Kim Carlson along with Kathy Schnitt. We are the remote participation managers for this session. Please note that this session is being recorded and follows the ICANN expected standards of behavior. During this session questions or comments will be read aloud if submitted within the Q&A pod. The rest of the housekeeping reminders, Kathy will add to the chat and with that, I'll turn the call back over to you Eberhard. Thank you.

EBERHARD LISSE: Thank you very much. Welcome back from the break. Next on our list, Jothan Frakes will give us a public suffix list update. We have heard him before. We will hear him again and today he will do his usual update. You have the floor.

JOTHAN FRAKES: Okay. Thank you very much and hello everyone from a community that I certainly respect and endear, the registry operator. My name's Jothan Frakes. I'm the CEO of a small boutique-sized registrar called Plisk. I'm here, though, to speak to you in the context of a volunteer project that I've invested the last 15 or so years of my life in called the public suffix list.

The title of this slide is voluntary rainbow bridge between domains and developers and I'll explain a little bit more about this. It's going to vary slightly from past presentations I've given on the public suffix list and I do make some assumptions that many of you are already somewhat familiar with the public suffix list. Next slide, please.

I'm essentially going to say what the PSL is, how developers—the developer community—interact with the domain names and then explain how the public suffix list plays an important fundamental role as a magical rainbow bridge between the world of ICANN and the worlds of Internet developers. Next slide, please.

What is the public suffix list? Next slide. Essentially, this is a document which takes the IANA root, which lists individual entries that are root listed to top-level domains and then we go into further detail within those so that we would have a list that has awareness. For example, not just of .uk being a top-level domain but also that there's an existence of a co.uk, similar with .au that there's a calm .au or other subspaces within those top-level domains that effectively operate as if they are TLDs.

Initially, this was created to help ensure that cookies would not be shared outside of their appropriate segmentation, such that somebody could not create a cookie for co.uk that could read anything of subdomains of co.uk. That would be a potential problem. Since then, the types of uses have widely spread. It's free and publicly available. It costs nothing to download. It costs nothing to submit. It's maintained by volunteers like myself. There's a handful of really dedicated volunteers who help keep this operational.

They're vetted and validated changes so we make sure that we use version control. People have to have account and have validated their most used two-factor authentication. When we receive updates, we vet them through requesting a special text record. Gets placed into the respective and effected DNS zone. The file is widely used and incorporated and here's an important piece. It's not really maintained by the ICANN or the IANA. Next slide.

So essentially, it expands the IANA list ethat they provide. It respects and sticks to the ICP3 one route which basically has us rejecting any alt route requests. We, as I mentioned, dive into further elegance and detail into individual namespaces. We also have what's called a pseudo or private section. Inside of that section, outside of those that come from the IANA list specifically, inside of this private section are jump-in points where a company like Shopify or GitHub or other companies that offer subdomain directories or other places that you would want an effective TLD behavior to happen inside of those subspaces. Next slide.

So here's where I'm really raw and authentic. It's the least awful but best-maintained resource of it kind. It's entirely driven by volunteers and contributors. There's no service levels. We make no warranties or guarantees on how it works. We don't imply any security but some do assume we do and I'll get into that a little bit later. The static public list, if used and incorporated in software, perpetuates the host.text dilemma where you've got a growing individual static file and we went into that in the document SAC070 with the Security and Stability Advisory Group.

If you request a badly formatted entry or you don't think your entry through, you can actually harm your namespace and we see that more in the private section where somebody will request … They'll have a primary website but they'll request a wild card be added into it and we will end up breaking cookies for their primary name which is a bad experience. We end up having to spend time really vetting and making sure people really need what they're after.

We don't have any control over downstream propagation so the file is created and generated. At any given point, whatever type of user downstream that's incorporating the public suffix list, they do that at their pace. They may make adds or removals to the file. They will do it at the timing that they choose to. One place that we see that really frequently is during the new TLD rollout when iOS devices would receive their update but on the pace of when you get an iOS update. If you make a change in the PSL, it doesn't magically cascade out to all the places it would get used, it happens at the pace of propagation of whichever voluntary user. Next slide.

Here's where this plays in a little bit more. Next slide, please. To a developer and from the developer's perspective. Domain names are "parsley," not the meal. And I specifically misspelled parsley because it's about parsing. Their primary care is about achieving what their objective is. The domain name is just like a phone to a business. It's just a resource utility, something that's going to help them do what they need to do. The business is the focus, not the domain, for the developers. Next slide.

So why is this important? Well, everything is online now. If you look at software such as Google's G-Suite, such as Microsoft 365, similar type of Suites are now on subscription models. They're very, very interdependent on domain names. There's just so many elements of our life that are online and utilize domains for email or for other identification.

So software needs this sophistication in dealing with domain names and they need a manner in which machines can just be more intelligent about processing domains. Next slide, please. There's three sagas in this. There was the early days and it's pretty much two or three characters would be a domain, some people would hard code some logic into creating elegance. I'm doing air quotes, "elegance" for what's a domain or not.

So they would just use regular expressions or other things. Then came, I'd say, in the 2000s because there was a couple rounds, 2000 and 2005, where more TLDs started to come out. A few more ccTLDs started to emerge. IDN ccTLDs came and programmers started to really look at how can I get into a bit more detail and accuracy here? And this is where we enter the PSL for creating a bit more elegance around that.

In 2012, there's the thundering herd, the .floodgates and all the IDN gTLDs. But there was a significant increase in the amount of top-level domains that were present and developers really looked for ways to help universal acceptance, which caused an incredible increase in the use of the public suffix list, either for validation of a name's existence or to just have your software be smart about what is a domain or not because to a developer's perspective, they were appearing every week and it would be hard to keep track of.

They don't follow like we in this group how closely the new TLDs are deploying. Next slide. App developers and ICANN. If you really think about this there's not really a place for developers inside of ICANN. Non-commercial business, those are made for developers. IETFs is really more of a standards body or a group that are putting forth standards. The At-Large really claims to represent users and not actual registrants all the time and so it's more wide. So it's not really a specific group inside of ICANN catering to the community of developers.

Now if there were, it's worth contemplating if the cost of participating would have as much value to them because working on standards type of work slows their pace. Governance means creating more friction to their objectives, typically. And the time and money spent on it, pragmatically, they would probably focus it more towards profit-delivering initiatives. Next slide.

So the problem to solve here is how do you machine parse domain transitions and do so in using some form of elegance? Regular expression, really bad. False positives. Doesn't solve third-level domains. String length, that's really bad. It causes a Y2K issue without the Y2K impetus. .info, even today, is still sometimes affected because it's four characters instead of three in some places. You know how many years since .info launched. So the thinking is, is there a list of TLDs, then? Instead of using regular expression or string length, is there a list? And somebody takes and looks in Google and they find the public suffix list. Next slide.

Here's where I talk about the rainbow bridge. Next slide. When the 2012 rounds initially started to deploy particularly in WebKit-based devices like Safari on IOS and Apple systems, there would be a confusion about whether or not to send the request of the user in the location bar to search or to the DNS system.

So there was quite a few registry operator executives that would go and show their shareholders a quick demo or, "Take a look at it on my iPad here," right after it was delegated and it wouldn't work. The presence of the TLD and PSLs had an absolute effect—a positive effect, once they would get listed—on making sure that they wouldn't get misrouted to search.

I had the privilege of working with a group within ICANN to help get the current contracting JSON file designed and we were able to rather than … I mentioned the propagation issue earlier. Because it would work at the pace of contracting, which would precede a delegation to the root, we were able to close the gap on that propagation challenge, such that you would get a good, hopefully 60 to 90 days lead time on having the domain name or having the TLD listed into the public suffix list.

It fixed the issue. The cost was that we went through an SSAC review but we did result in SAC070 and it gave some great advice on things to tighten up and tune. We had completely revamped our process of acceptance and submission for the PSL and it really gave us good advice on some things that we could do to make sure that this was a healthy and stable process. Next slide, please.

But I'll tell you one thing. And it shouldn't make you scared, but it should be known. This all leans together informally but very widely. Over the past 15 years, adoption and incorporation has really spread out because it's the only resource of its kind. Entries, as I mentioned in the case of iOS have a widespread affectation to help a domain operate in an expected way with respect to cookies. As I mentioned, that can be good or bad. If somebody makes a mistake, it takes a little while to roll back due to the propagation. Next slide, please.

So the volunteer piece. Next slide, please. Browsers, they use it because it exists. It's free, it's simple, and it's a casual relationship. The browser adoption is the widest use and it's the majority of the market of browsers that use it. But each do it their own way. How frequently they refresh it, what kind of delta, do they do any processing to add or remove entries and the behaviors related? Some may treat it differently with respect to how it may list a URL in the location tag. It may do different behaviors, how it treats a DOM or other aspects of it can be a little bit different in behavior within the browsers.

It's a choice, though, to incorporate the PSL. There is no compulsion to do anything on behalf of any of the browsers or other users for that matter. SAC070 made some recommendations attempting to compel developers to suggest that they do certain things and developers there’s really no dominion of the security and stability advisory committee over developers outside of ICANN's orbit. It’s like, “Okay. Thank you. But we’re going to keep doing what we’re doing.”

So the challenge is browsers may or will abandon the PSL for their own solutions, if too much changes happen to the status quo situation. When I press on this to try to figure out if there’s some more organization that could be placed into it, individual browser authors, software companies will suggest that they may just go their own path and proceed in that direction, which would balkanize and separate out and have a variety of different places rather than one place to affect domains’ behavior.

So that could be good, could be bad, but the status quo is that the majority of the market listens to this PSL and uses it for domain affectation. Next slide. Other groups do too. There’s a variety. Let’s Encrypt and other certificate authorities use the public suffix list so that they don’t generate a wildcard certificate for what is supposed to behave like a subdomain. Cloudflare, you can put a domain into an account in Cloudflare and you can add so many but you can’t have a domain in multiple accounts if you’re not in the public suffix list. You may run into size, scale, or rate limits there.

The Facebook Pixel. Recently, there was a challenge where Apple updated some of the behavior related to tracking where there was an opt-in to tracking that dramatically affected Facebook Pixel. Both Apple and Facebook had these support pages that you could ultimately find and they basically said that if you’re listed in the public suffix list, then things will work fine. So we received a big flood of requests but we declined them due to the need to actually go through and give a bit more review of those. I'll talk about that a little bit later.

Security solutions, there's a lot of things such as DMARC but also many of the reporting software that show reports, use the public suffix list to split and identify subdomains so that they don't group together or get aggregated incorrectly.

Apps and libraries are using it to parse a field input, process domain logic, measuring for valid or invalid as field entry or otherwise for processing. You also see this as linkification in your Skype or Telegram or Twitter. And there's loads of other purposes, a lot of which are covered in the hard work of the people in the universal acceptance group. Next slide.

Registries, this is my call to you. I have had a real privilege. I come and present to you at the Tech Days, I guess to keep a connection point so that you can talk to somebody if you need some help in reviewing it and understand that it exists and how to affect your zones. So if you would like, I have my contact information on the last slide.

But we really do watch for and take very seriously the requests that we get when they affect a top-level domain versus the private stuff, the subdomain. We want to make sure we can validate any kind of change. We use the DNS to do this. And in some cases, when we can use the DNS to do this, we'll either use the listed policies on the IANA-listed NIC or we'll reach out to the IANA contact, which I know you get a lot of spam to. But we'd ask that you maybe watch for something for the PSL if we're looking to validate something.

With the cascading benefit of correct listings, it's really a positive thing for your TLD. It can really help you ensure that your customers are having a good positive experience with their domain, which I know is important to all of you. And I really am glad to help and serve in this role of volunteer, to help make sure that your listings are accurately listed and I look forward to hearing from some of you afterward. Next slide.

I've been working to educate developers to understand what the public suffix list is or isn't. That can be good. It's wonderful for machines understanding TLDs or effective TLDs and it's great to have software be smarter about domains. It's not, as I mentioned, designed to be a sieve for limits. I mentioned that Facebook had essentially created a dead-end landing page in their support queue for people who were affected by this iOS change and they just listed the public suffix list as the magic solution. So we got flooded with requests and there were not very well vetted. So we had to work with Facebook to make sure that the requests weren't going to break stuff for the people who were requesting them.

We don't want to be the support queue or some form of a workaround for rate limits. It's a trust system. There's validation and verification but there's no implied security or trust in what we do. It's not intended to be a workaround for any kind of rate limits and it's not designed to receive customer service queue. I call it punts. But we want to make sure that we’re not doing support calls for other people if they can’t solve it themselves. Next slide.

Some question and answers. We probably only have a moment or two here but I’m glad to answer any questions. Also, I’m probably pretty well known so if they are any questions, glad to answer them offline, in the sake of time. Otherwise, I really appreciate your time today and next slide. Thank you very much for your time today and you can contact me. My email address is on the slide here. You can find out more about the public suffix list at either of those two URLs. And recently, I had the opportunity and privilege to work with a great team at ICANN's OCTO to help create some documentation around helping IANA top-level domains interact with the public suffix list where needed.

So thank you all very much. Appreciate the privilege to speaking to you today.

EBERHARD LISSE: Thank you very much. There is the question and answer pod so please open that so you can see it.

JOTHAN FRAKES: Yeah, I almost got away without … Okay.

EBERHARD LISSE: Yeah. One question was Mats Dufberg asked in the chat where the list is available and that question has been answered. You can see it right here. The other question is Calvin Brown says, "Thank you. That's fine." Then Shubham Saran, NIXI says, "What's your recommendation for the IDN TLD registries?"

JOTHAN FRAKES: Calvin, you're welcome. I'll answer it live. Shubham, that's hard to answer. The IDN TLDs are in the public suffix list. They come from the JSON so they're listed and added. I will just say NIXI, in the case of IN, you have quite a number of different language variants because of all the diversity of languages and character sets used there. They would be certainly worth taking a look at your entries to make sure they're appropriate.

We do, from time to time, track the IDN ccTLD announcements and we are always looking to make sure that we're up-to-date on that but we could always use help so if you would take a look at the public suffix list and ensure that NIXI's entries are accurate, that would be a good starting point.

EBERHARD LISSE: Brett has asked one and a half questions, which is the last one we're going to take. Go ahead.

JOTHAN FRAKES: Brett, all the time. People are coming up with all kinds of ideas to throw this into DNS. We get a variety of different ways this is presented to us. There is, in fact, one or two projects that have done this, that have it in the DNS so you can call it on demand. That is there. A .psl, we would have to go through the process to apply for it and we're volunteer-driven so we would most likely not go down that path. I could see it potentially at some point being a registry. But we have to be mindful of the voluntary nature in which our users would take and all of the status quo purposes that are out there. So if we make too many changes we would have to really make sure that they supported the legacy use cases. You're welcome.

EBERHARD LISSE: Okay, thank you very much. Of course, any other questions take it offline via email. Thank you again, Jothan and I hope you're doing well. I'll sort you out later today or tomorrow.

JOTHAN FRAKES: Thank you, sir, and thank you all for your time.

EBERHARD LISSE: You are welcome. Next one is, and last but not least, is Howard Eland from Donuts who will speak how to sort out algorithms on large scale. You have the floor.

HOWARD ELAND: Thank you. Can everyone hear me fine? I hope.

EBERHAD LISSE: Yes, we can.

HOWARD ELAND: Yup. Very good. This is Howard Eland, previously working with Afilias. Now we've moved on to Donuts. A bit about that at the end. But for those of the folks who don't know me, I've been running the DNS side for the Afilias folks about 15 years now. I just wanted to talk a little bit today about some of the recent projects we did with getting rid of SHA-1 across our TLDs and hopefully help folks understand what they can do if they have yet to perform this function and are interested in knowing the trials and tribulations and what to watch out for. Next slide, please.

Okay. So just a little bit of background about what we were looking at and some decisions that we were making here. When we started out, just without DNSSEC we were at about 50 million resource records in our various and sundry registries. We had started our DNSSEC operations way back in 2008 and then we signed .org the next year, as well as some of our others—.info and quite a few of the others.

When we started, we went with NSEC3 with opt-out, which was at the time was provided by Algorithm 7 and we chose this for a couple of reasons. At the time we were looking at the advantages of A, not having to greatly expand the size of ours zones with NSEC and also the perceived increase in security by stopping things like tree walking—or at least I should say by averting things like tree walking. That's why we chose NSEC3 at the time and we also had both type 1 and type 2 DS record in there.

Then just a little bit on some of our DNSSEC parameters, we also have our NSEC3 hash iterations set to 1 at the time, a static [inaudible] that we almost never changed. And we also were pre-publishing the next KSK like well into the future. When we'd roll one, we'd effectively start pre-publishing the next one. So, the reasoning behind that at the time was to be able to effect a KSK roll in a hurry should we need to. Keep in mind a lot of those decisions were about 10 years ago or so. Next slide.

Okay, so a little bit of scope and driving factors on this change. Certainly, the impetus was to get rid of SHA-1 because of the ability for that to be compromised. But we also took this opportunity to say what other things can we do? It was well past time for us to look at some of the original choices to be made for DNSSEC implementation. And specifically, we wanted to say the things that we thought made sense then, do those make sense now? Because times have changed and we've learned a lot over the decade of DNSSEC implementation so it's well past time for us to take a look and say, "Can we better choices there? "

Then lastly, the other thing we had to do is make sure for any change that we wanted to do, of course, that we had the capability with the designers and our infrastructure to facilitate those changes. That is a very important point that I'm going to come to again here in the future.

To give you an idea of our scope, we had 209 top-level domains we were looking to make changes for. We also had another 208 zones in which we had … Either the name server records themselves were under the nic zones, or we had other second level or subzones, like for example in Australia and [South Korea], [inaudible] and what have you.

So the scope of this involves a total of about 417 zones and then we have some extra challenges that we needed to address as well, the biggest of which was almost all of them—I would say 70% or so, ballpark—were also already mid-KSK roll, which meant we had to complete that KSK roll before we could start the algorithm roll. That added both time and a little bit of finishing up that procedure to the project.

In many cases, the customers, the registry operators themselves, were the IANA administrative contact, which just meant that the added extra coordination and also some time to allow folks to make changes and such. Of course, we had our good friends at ICANN knocking on our door saying, "You guys are going to get this done, right?" Then we had the other parts of the industry giving us wonderful little hints, like DNS, [inaudible], with warnings saying, "Hey, SHA-1's bad. You should not be using them." So we had a lot of impetus to get this moving and get her underway. Next slide, please.

So some of the research that we did here, the first thing we did was look to see have folks had any problems or encountered any things before us? And I specifically want to call out the good folks at RIPE and at SIDN labs that went through a lot of this pain first out of the gate and did a really good job in documenting those things. I would strongly suggest if you're going to look at this, if you haven't done it yet, that you start there with some of their research so you can see some of the gotchas that they found. So that was a huge help to us. We didn't want to repeat any mistakes or any issues that the other folks have found so those are great resources.

Then off to the lab we went. So the first thing we did was start testing algorithms. So we looked first at SHA-256. That didn't have any issues at all. When started testing for the elliptical curve algorithms, we started seeing issues in our infrastructure were taking a long time to find. I know a lot of folks who said, "Wow, it should be faster," or what-have-you. But this was unfortunately where we saw a bit of challenge within our particular infrastructure. So between that and the fact that some of these algorithms were relatively fresh from an industry adoption at the TLD level, we decided that let's just stick with the devil we know here and we decided to stay on with SHA-256, Algorithm 8 for the roll here.

The other testing we did in addition to this, we set our NSEC3 hash iterations to 100. This was based on some of the guidance that has been out there for a while and is now a bit aged as well. Before anybody grabs a tomato, please just hold that thought because I'll have more on those hash iterations in just a moment. But that was one of the parameters we looked at. The next list is to say do we still need to pre-publish this KSK right at the flip of the other one? And we've decided against that so we'll only now pre-publish KSK's in advance of the roll—30 – 60 days or so before the roll. That, of course, shrinks our response size pretty significantly.

We also are getting rid of DS type 1. If we're going to get rid of SHA1, let's go ahead and do it. So we did that then we made sure that we followed each and every step for RFC 6781. And I really want to emphasize that because there are things that we found in the lab within our particular signing infrastructure where things worked but not exactly as we had expected them so we did find some surprises there.

It was very important for us in the lab to go through each and every step of the algorithm roll and not just assume that anything works. So I really want to emphasize if you're going to do this or if you haven't done it yet, please make sure you go all the way through and get your timing set and things like that. We'll come to that in a bit. That's the piece to the signer operations—not just even revolving around the algorithm roll but how things move to a steady state as well as the actual function. So those were all critical pieces that we went to the lab and took a look at. Next slide, please.

Okay, so our approach. We wanted to focus on two specific areas in addition to the technical piece. That was education and notification. We wanted to make sure that the registry operators that we work with understood what was going on and knew, timing-wise, how things would happen and also to give them a heads-up when we were about to make any change to the root or what-have-you so they could participate when they're the admin contacts.

They wanted to make sure they knew what we were doing, why they were doing that. But more importantly, on the internal side, we spent a lot of time with our account managers within our organization to explain this process and why it's different from a simple KSK. You can imagine trying to show folks, A, how a key roll works in general but in addition why an algorithm roll is a special case. There is a lot of education that went on there. Then lastly, we wanted to make the poor folks at IANA know—give them as much heads up as possible that the flood was coming with 209 TLDs getting multiple requests for root zone changes. We tried to keep them well abreast of any time we made any changes. The reach out that we did was pretty extensive.

Then, one other piece. This is somewhat of a shameless plug. No, I'm not invested in Notion at all but I'll tell you for our internal tracking purposes we use Notion for this. It was really a great tool for communicating to other non-technical forces of the organization so they got a good feel for what the status was. Any time the status changed, we would make that update in Notion. And when you do it and with one view, you flip to another view and now you've got Kanban cards you can flip around or you've got a nice little Gantt chart there ready to go. So it was definitely … Our tracking Notion was very helpful. Highly recommend it.

On the backside of things, what we did was we wanted to break this up because to go all at once was just too big of a project. So we bumped up our zones into several different batches of changes. The first one was a proof of concept, where we had just—I think there was four TLDs and their associated NIC zones—just to make sure we had our processes tweaked exactly how we wanted them. Then we went with the Afilias zones. This is the colors red, blue, what have you. The PIR zones for .org and some of their accompanying zones, NGO, ONG and such.

Then we had a set of new TLDs and we kept the ccTLDs into a separate group. And then specifically Australia, they had just some minimal changes as most of those were ... We don't run the TLD itself and these second levels were mostly all already set up with Algorithm 8. So they just required things like a DS record removal for that type 1 DS record. So that was how we broke it down. Next slide, please.

Away we go. So we started this process here. We started the planning in January or so. The first one kicked off on the 31st of August of last year. Most of these zones were done in less than two months and that's an average with a wide spread. Some of them were done pretty quickly. Others took quite a bit of time. But one of the things that I want to bring out with this was we made a conscious effort to not mess with TTLs during this process. So in many cases, that incurred a longer hold-down time for us. So it was measured in days because a lot of these TTLs were set to a day.

But we wanted to not change too many production parameters in flux and we thought it was worthwhile, even if that meant that the time it took for an individual set of keys to roll would be increased. I think we were pretty happy with our decision there. We didn't change our resign schedule either. We signed on the 1st, 9th, and 17th, and 25th of each month and we kept to that schedule, even if that would mean there were a few days that went above and beyond what was needed for a hold-down.

So that all worked out pretty well. Right now we are at 416 out of 417. We've got one last TLD. Actually, this is now updated and that last TLD is now on. It's over the roll. It's waiting for that DS record to get published so you're still done. Not sure if we're going to hit that June 30th. I really wanted to hit Q2 but it may take a week or two into Q3 here, depending on that last one finishes up so we'll see. Okay. Next slide.

Okay, so where did we end up? So, now all of our zones are sitting on SHA-256, which is good. There's only a single DS record that is type 2 in the parents. We changed the salt. We did not get rid of it. We simply changed it. And we ended up with hash iterations at 10. So this is because of some of the work that has been going on for a while about discussing the attack vector that can hit at some [IR] resolvers when the hash iterations are too high.

So after many discussions with lots of folks that are OARC and such, we said, "Well, let's drop her down to 10." We didn't go all the way back to one because there was governmental issues with going all the way to the far end of the scale there so we settled on 10. That KSK pre-publish is now only during the KSK roll. It is not otherwise pre-published. And we also had this strange oddity where the ZSK, for some reason, was signing the DNSKEY RRSet which all I did add to response size. So we got rid of that silliness that was going on there and I think all those changes seemed to really improve our DNSSEC posture, I think.

So if we go to the next slide you'll see this is the red TLD zone and you'll see here on the left what things we're looking at. Like before, you see our extra DS and DNSKEY pre-published, along with all these little warning signs saying, "Hey, you're using SHA1. What are you doing?" So if you look on the right-hand side, now we're all nice and clean and svelte, if you will. So I think things look a lot better. Next slide.

So what are some of the challenges that we had in getting to where we are today? We did have one zone where a signer bug got triggered with just some precise configuration changes that happened at the wrong time and it chopped up some of the signatures on a zone for a few minutes so that was one that caught us by surprise. The other challenge—the bigger challenges, though—were with some of the RZM changes just because there were a ton of them and we could automate some things but we needed to keep some manual checks in place and also perform some manual functions.

So we had scripts that would generate the data to make the changes for each step and we also had scripts written to … They would basically scrape the confirmation page , if you will, before we actually launched it and those emails that went out to the contacts just to triple check. But after each and every step, we did a manual QC as well. So we checked things to make sure that each and every step performed as we expected to, just with different personnel looking at each of the options.

We also did that for the Notion update. Unfortunately, their API feature just came out recently and at the time we did not have that so those were manual updates that we wanted to cross-check to make sure everything was right there as well.

Then we had a couple of weird odd process states when we were doing RZM stuff. Almost all of those self-cleared though. We would get things like a zone would go into the exception state and that's all we knew. If you checked back a little bit a little bit later, it was no longer in the exception state. That was a little odd but for the most part, those just cleared up by themselves.

By far and away, the hardest process hurdle were delays waiting for registry operators to confirm the RZM requests. Despite our reach out and such, it just took some time for some folks to make those changes. Part of that reasoning was the IANA contacts had become either inaccurate or stale and either that email address was no longer valid or that person was no longer with the organization and it was not set up as a roll account. It was set up as a personal account.

So there were some challenges there, where we had to fall back to the letterhead confirmation procedure in order to get the context updated so we could then move forward with our changes for the algorithm roll. We had this whole pandemic thing going on in the meantime as well. Next slide, please.

So just some recommendations, lessons learned. If you haven't done it and you're getting ready to do it, the first and foremost, as just mentioned, keep those IANA contacts current. If you're just starting to research how to do this, that would be the first thing I would do is check to make sure this is done so you can go through that process now so you'll be ready when you finish. You'll be ready to go to your algorithm. Also, if you can set them up as a roll account with a group email address so that you don't have a single point of failure should someone go away, I think that's very helpful.

The second thing is to watch your timing. I know there's a lot of literature out there about having a hold-down for twice the TTL. We found that four times TTL was a better bet, just because the way some folks tend to ignore the TTLs that you set. So out of an abundance of caution, in many places, we waited for at least four times the TTL and that seemed to help a lot, even at the expense of increasing the time for this project to complete.

Then also, make sure you have a good understanding of which TTLs affect which steps. We didn't have an issue with this particular piece but the reason we didn't have an issue is because of the research that I had mentioned before, because we saw how other people said, "Hey, be careful with this," and we took note of that and that worked really well for us. So that was a big help.

Communication is a key. Make sure everyone is informed and current, including whoever is your parent. Whether that's ICANN or anybody else, make sure folks stay informed and there's no surprises that are avoidable. That's a key factor.

And then the last one, make sure you're in that lab testing each and every single step just because there can be some special oddities with the algorithm roll that you may not have even seen with the standard KSK roll. Next slide. Just a couple of thanks to some folks. Carl Clements was the guy doing the real work, I just flipped around RZMs and did some stuff on the root sign. But he was the one cranking through the signers and getting the real work done, so kudos to Carl.

Then I wanted to specifically mention Joe and Suzanne over at PIR. They always provide sage advice. As one of our largest customers, we took in as many of their recommendations as we could and they were a huge help. Then lastly I have to call out George and Selina over at ICANN, working through the IANA function. Boy, we hit them with a flood and they were troopers through the whole thing. They did a fantastic job, extremely professional, very timely. And we were extremely impressed with all of the hard work and help that they worked with us through on this project because I know it was a lot of changes. Okay. Last slide, then.

So you can reach me here at either one of these email addresses right now. We're trying to figure out if I'm an Afilinut or Dolius person. But we're in transition right now so either email address works but I'm happy to take any questions that you might have.

EBERHARD LISSE: Okay. Thank you very much. That was an interesting presentation. Many of us have been through key rolls. Some of us have been through algorithm rolls but doing this at scale is also very interesting. Please open your Q&A pod on the bottom. I'm going to read the question but it's better that you answer it from reading. Hugo Salgado asks, "Do you have any numbers if the size of DNSKEY plus RRSIG response decreased after the change? Although RSA SHA-256 is larger, I understand that not pre-publishing KSK or signing this as ZSK may have resulted in a lower net result."

HOWARD ELAND: Yeah. No. That's a great question. We absolutely saw a huge drop in response size because we had … Specifically, if you look at the DNSKEY RRSets, someone would send a query for that and the response that would come back were two KSKs, two ZSKs, and then a signature from the first KSK and a signature from the ZSK. So as you can imagine, that response size was pretty good-sized. We saw a drop, I want to say we dropped a good 20% or so. I don't have the actual number in front of me but we saw a big drop when we stopped doing that.

I don't think you necessarily expect, as you say. So yes. We did see a pretty big drop in packets. I can even do one of those right now. Yeah. So I just did a looking for the DNSKEY set for red with signatures on here and now I'm getting a message size of 895 so much better than what it was before. It was well over 1,200. So, yeah. Anything else?

EBERHARD LISSE: Thank you very much. That leaves us with the closing words, which is done by Jacques, who is one the video already. Please, you have the floor.

JACQUES LATOUR: I'm a little bit tired but it's all good. So today the first presentation, I think what we have is a comprehensive system for abuse, detection, and mitigation. I like the concept or the idea of deferred delegation when something is suspicious. It's a really good model for ccTLDs and gTLDs to look at and to think of supporting. Anything that makes it harder for bad actor to register a domain name abuse is good and there is a little bit of everything in there with machine learning and reaching out to intelligent partners to assist in the abuse mitigation so I really like that.

Cristian from SIDN did a presentation on DDOS [inaudible]. So it's about fingerprinting DDOS attacks, storing them in a clearinghouse, and sharing with other parties. So curious to see how the anti-DDOS Coalition Group is going to work in the pilot phase and so on. Looking forward to see what's the next steps on that are. I think Cristian did also a really good presentation on .NL, on where they're at from an architecture and operation and what their labs are up to. So that was a good host presentation.

Ray did a presentation on root system [inaudible]. So to look at the root servers' response time latency using the RIPE Atlas measurement. And what I understand is Brett is going to share the result for .UK and from this so that would be next Tech Day meeting. That's interesting.

So Jothan from PSL provided an update on the public suffix list and the challenges of using the list and how developers—what are the do's and don’ts so it’s very challenging to move this forward so that’s a big presentation.

Very interesting. Last session, Howard talked to us just now about [coordinating] a huge project to migrate away from SHA-1 at scale. It’s not something a whole lot of people can do so I think there’s good lessons learned in there and a good opportunity for a lot of people to clean up the IANA contract registration. So I think, in the end, we have a much better infrastructure. That’s a wrap for Tech Day 71.

EBERHARD LISSE: Thank you very much and that leaves me to thank our ICANN staff and, in particular, Kim and Kathy for their Zoom magic, as Stephen Deerhake usually says and the technical support. I have not listened to the live transcript or watched the live transcript because I find it distracting. But the technical thing worked very well so thank you very much. And I am in favor of doing a hybrid thing. And if it happens in Seattle, I’m definitely traveling if they let me, being vaccinated now and all. But if not we’ll do a virtual one again. Feel free to get in touch with presentations. In particular, Brett from the UK, I would like to see what you do with your stuff there. Okay. Thank you very much.

KIM CARLSON: Thank you, everyone. Please stop the recording.

[END OF TRANSCRIPTION]