JOHANNESBURG – DNSSEC Workshop
Monday, June 26, 2017 – 09:00 to 12:00 JNB
ICANN59 | Johannesburg, South Africa

JULIE HEDLUND: Welcome, everyone, to the DNSSEC Workshop. Please take your seats. Be sure you grab a program. On the other side of that is an all-important lunch ticket that you're going to want to keep. We are going to start just a little bit late to give more time for people to arrive. But anyway, welcome to the DNSSEC Workshop.

Welcome, everyone, to the DNSSEC Workshop. We'll be starting probably in about 10 minutes just to give people more time to arrive. Don't be afraid of the front seats. You can move up nice and close, we don't bite much, and be sure that you grab a program. The back side of the program is your lunch ticket. Lunch will be at the Level 4 foyer following the workshop and before the continuation of Tech Day. The opposite side of your program is that ticket. You should hold on to that ticket even if you leave the workshop so that you can attend lunch. Thank you.

Welcome, everyone, to the DNSSEC Workshop. We'll be starting momentarily. There are plenty of seats up front in case you'd like to come up a little bit closer, and be sure to grab a program. The back side of the program is your lunch ticket so if you want

Note: The following is the output resulting from transcribing an audio file into a word/text document. Although the transcription is largely accurate, in some cases may be incomplete or inaccurate due to inaudible passages and grammatical corrections. It is posted as an aid to the original audio file, but should not be treated as an authoritative record.

lunch you will need a ticket. If you don't see one near you, they are scattered all around and I can see that there are some on the front row here. Anyway, we'll get started momentarily. Thank you.

Welcome, everyone, to the DNSSEC Workshop and we are going to get started in less than a minute so thank you for joining. I'm Julie Hedlund from ICANN staff. I'll shortly turn things over to our Master of Ceremonies Jacques Latour from CIRA, but please do come up front if you'd like. We promise we won't bite.

There's also some programs around. Be sure you grab a program and be sure you keep that because the backside of the program is your ticket to lunch so if you want lunch you will need to have a ticket. With that, I will turn things over to Jacques. Thank you.

JACQUES LATOUR: Thank you and welcome to the ICANN59 DNSSEC Workshop. Today is the half day workshop so this morning is the DNSSEC Workshop, this afternoon is Tech Day, so we have a compressed program today.

Can the Program Committee member in the room raise their hand just so we know who's on the list here? All right, good.

We spend about an hour a week between every ICANN meeting to plan this workshop so it was quite a lot of work from the Program Committee on this.

Today the lunch is sponsored by CIRA, Afilias, and SIDN so thank you to the sponsors for free food. And if you want to sponsor, we're looking for more sponsors all the time so you can reach out to Dan York. His e-mail address is down there. We're looking for more sponsors. That means we get better food, the more people we have. It's also sponsoring for the Implementer Gathering event so we have a meeting. We'll cover the details on that later but so we need more sponsors.

The DNSSEC Workshop is organized by SSAC and also with the help of ISOC, the Deploy360 Programme with Dan York. Like I said, once a week, we have a meeting to plan the workshop and focus on the content.

So this is the agenda for today. It's our standard presentation we're doing. So 9:15 we have DNSSEC Deployment channels so that the regional panel and Mark is going to moderate this panel. The panel is going to last until 10:15. The Middle-Box presentation by Andrew is going to be done at 10:30, so we have a change of agenda there, and then the Root Key Rollover Tutorial is postponed. The people presenting this are not available for this. We all know what the date is.

Then in the afternoon we have a panel discussion. It's kind of an experiment. It's what's the policy impact of CDS and CNS implementation, and then we have DNSSEC How Can We Help or How Can I Help. Russ is not here today so I'll be doing that one, and then we have a quiz at 11:50 and then the lunch.

DNSSEC Deployment Around the World, so numbers, stats. ISOC just released a report recently on the state of DNSSEC deployment so you can download it at that link. It's a pretty detailed report on how the DNSSEC stats in general for the year 2016.

In terms of DNSSEC validation, it's pretty stable. It's around 14% global. Those are the APNIC Lab stats. That's a program that Jeff Houston runs so we've got a lot of stats on DNSSEC validation there.

Interesting. Oceania, Melanesia or Oceania, Africa are on the top to validate DNSSEC and that means the ISPs are validating DNSSEC for the African region more than the rest of the world. That's a good thing.

So DNSSEC validation is the ISP resolving DNSSEC and use Google public DNSs, the percentage of the ISPs that rely on Google recursive to validate DNSSEC. The numbers vary around the country or the region.

In terms of Africa specific region, we see that by the country the DNSSEC validation is some countries are pretty high and the ISPs also rely a lot on the Google DNS to do DNSSEC validation. That's an indicator if they run the own recursive or if they rely on Google. Overall these are good numbers. Much better than Canada, at least, so that's a good thing.

In terms of deployment, DNSSEC deployment worldwide, Rick runs this report. You can see the number of ccTLD that are assigned in the root. It's pretty high for TLDs to be signed in the root but the second level domain based on these stats is fairly low. It's like 5%. Less than 5% are signed at the second level so we need to do a lot of work at that level.

And then the validating users, that's based on the APNIC so around 15%, so we're making progress. A lot of TLDs are signed now. We need to focus on the second level, try to bring that line up from 3% up to a higher number.

So this is the number of signed domains second level by TLD. SIDN – .nl has the highest count of signed domains at 46%, Brazil at 24. This is sorted by total domain, sorry. You can see the stats in there.

New gTLD, the slide here shows that .bank has 100% of the domains. There's 15,000 of them, second level, and 100% of .bank are signed. I guess that's part of their valued proposition

that all the second level domains under their TLD are signed with DNSSEC. So that shows the stats.

Now we do the DNSSEC implementation around the world, so we track various stages of implementation status. Yellow is they plan to deploy DNSSEC partial. That means it's only assigned but it's not in operation.

There's a DS in the root. It's a flashy green and operational means the DS is in the root but they're also accepting signed delegation to the registry portal 3PP.

So we're making progress. So that's the latest stats. We've got a bit of work to do, still, in Africa but overall we're getting there. In terms of Africa specific stats, the latest TLD to be signed in Africa, the ccTLD, is Liberia.lr. They were signed in April this year so congratulations. Anybody from Liberia here?

In terms of APTLD stats so Asia Pacific, Saudi Arabia signed in June 2017. That's the latest one to be signed globally as well. Iraq is still work in progress. Italy announced—I think they're partial now, I'm not sure. I think they actually moved up a little bit.

We're still working on Greenland. Erwin has an action item to get Greenland information here. And then the LAC region, so Argentina is still DS in the root so we've got a bit of work to do in

this region also to get the DNSSEC signed. And then Greenland, that's part of North America, yes, and they have a DS in the root so they need to go to the final stage and go operational.

We published a map through ISOC. We have tools to monitor the status of TLDs and all that and then we publish a map every Monday morning.

Dan likes to collect stats and information and he's got a history project that he's working on so please go look at the project. If you have any information to contribute, that would be very welcome. He's looking for more information to populate his website, so pretty much history about the DNSSEC deployment. He's trying to track all the pertinent details around that.

And that's it. Any questions? Okay. Now we're going to go to the regional panel with Mark. The DNSSEC Deployment Challenges.

MARK ELKINS: Thank you very much, Jacques. Good morning, everybody. My name is Mark Elkins. I work for the local registry in South Africa and the backend provider for the registry, DNS, Domain Name Services, in this case.

I'm moderating four people including myself. Alain is in the audience somewhere as well, isn't he? Will he come sit at the front as well and join us, please? Alain Aina is actually the first

person to speak. He has been running around Africa trying to get people to deploy IPv6 and DNSSEC. Those are his hats so here we're going to be talking about, I guess, what he's been doing. There's no slides for this. It's off the cuff. Off you go.

ALAIN AINA: Okay, thank you, Mark. You can see that it's difficult for people to spell my name. My name is Alain Aina. I work for WACREN, West and Central Research and Education Network, but I am also a project lead for the ICANN DNSSEC Roadshow, and I think people may also be interested to know that I'm one of the crypto officers for the root zone for the East Coast key management facilities.

As Mark said, as part of the DNSSEC Roadshow project which is a project conducted by the African Bureau of ICANN, we try since 2013 to get the ccTLDs in Africa to deploy DNSSEC. And as you saw on the map and as Jacques was saying, we still have work to do, but if we compare the map from 2013 to what we have right now, you can see that we've made some progress but we still have a long way to go.

In 2013, we ran a survey. We asked some of the African ccTLDs and one of the questions at that time was what are the main reasons of non-adoption of DNSSEC. From the result we got, I think the top four reasons were lack of knowledge and local

expertise, lack of local community interest, lack of DNSSEC risk assessment skills and capability, and registry operation not ready. These were the top four reasons we got from the survey in 2013.

Based on these results, we engaged in three tracks to help the ccTLDs to close the gap and adopt DNSSEC. The first track was, okay, those who said through the survey, "We have no issue. We are ready to go."—at that time I think TZ was on that list because I see my friend from TZ in the room—first track was to just follow-up with these people. We say, "Okay, if you guys said you have no issue, please move on and let's share your timeline, etc."

The second track was those who said they deployed DNSSEC but there is no community interest. So we say, okay, we'll organize the event and we see how we increase the community interest.

The third track was—and this is where we had a lot of work to do—was to help those who said, "We're not ready, registry operation not ready. We do not have expertise," etc, etc. We tried this since 2013 to engage some of the ccTLD so we organized DNSSEC Roadshow event, three days. One day was Global General Awareness, the second day, Technical Days, and the third day we spent time with the registry to do risk assessment.

You can see that any time we're trying to do the registry assessment and the risk assessment, this is when the thing stops then we find out that the registry is not at all ready – no process, people, skills, no process registry operations, people are not even confident of the current registry operation, etc, etc, so you'll just stop and say, "Oh, we can't talk DNSSEC here. We need to fix the current registry." then you end up having a plan. So this is the first thing you must do, second thing you must do, etc, etc.

The other thing we also noticed which is a challenge is most of the registry do not have a test environment. No test environment so it's difficult. You ask them, "Okay, when you finish cleaning this and having this process in place, the next thing you need to do, you need to create a test environment where we can try the DNSSEC deployment, etc.

So these are the challenges we see on the ground and the cc, but you can see things are improving. If you look at the statistics, we have now almost 18 cc who have DS in the route but from this 18, less than half are already operational as you saw.

So the TLD itself is signed but the second level is not signed and they're not accepting DS from the registrar or from the end user. But this typically explains that the first step for most of them is to get the TLDs signed but because the registry operation is not

ready, they have difficulties getting the second level zone sign and accepting DS. So that's okay, let's get the TLDs signed so we'll be seen as green, then we will continue preparing the registry, the registry operation and the relationship with the registrars and the registrars, then when we are ready then we can sign the second level domain and then we start accepting DS.

But this also has a risk. We have seen that the people – I won't mention TLDs but I think we all know – we have seen TLDs who sign the TLDs, they have not signed the second level, not accepting but because the DNSSEC and the TLDs broke, the whole TLD disappears.

So then the question is if you're not ready to be fully operational and you just keep the TLD signed and not maintain it correctly, you create a different problem. This is what we've seen but we encourage them to not reverse. Keep the TLD sign but put process in place to maintain the TLD to avoid—because we also used to tell them, "Before you start selling the services, if you keep having issues it will be difficult because DNSSEC is about trust. If you break the trust before, it will be difficult."

So this is the issue or the challenges we are dealing with. I think they explain some of the data, the statistics we see and I think I will stop there. Thank you.

MARK ELKINS: Thank you very much, Alain Aina. The next presentation is by myself, is by me. And if I can get my slides up, please? Okay, like I said before, my name is Mark Elkins. I'm also the DNSSEC trainer for the South African Central Registry and so what we have been doing for an awful long of time is a lot of training and some of the successes will come out in a moment.

Okay, great. Okay. So South Africa ZA did sign the .za a while ago. Like everything else, these things seem to take a lot, lot longer than an engineer can understand. The plan to do the whole approval was initially way back in August last year and we eventually got ZA signed around about the 12th of September.

It sat there without any delegation from root for awhile while we tested things, and just for fun we also brought in web.za—web.za is one of the smaller second levels—so that we could test relationships between the second level, the top level, and the root.

I understand that the ZA DS records were put into the root around about 9th of December so it was a great Christmas present for us here in South Africa. And everything was working very nicely. We then looked at the next couple of second levels being org.za and net.za. Org.za was relatively large-ish in that it

was about 25,000, 30,000 domains. Net.za was very similar in size to web.za. So those are the small ones.

A little bit of background, in South Africa, initially the person who got ZA assigned to himself was doing that basically so that he could get ac.za, the academic or the university network up and running. He then handed out other second levels to whoever would volunteer to run them.

The South African Central Registry has since collected web.za, net.za, org.za and co.za, as well as looking after the three cities and .africa as well. There are other second levels out there in the wild.

Everything was looking really good and org.za and net.za were added into the system and we have these tests that we run, etc, and then all of a sudden when co.za was added, it broke.

Co.za is over in million domain names and what would happen is the OpenDNSSEC would quietly go to sleep and not wake up. When we tried to get it working again, what happened is that it decided then to break everything in the OpenDNSSEC and all the others suddenly disappeared on us, so before anyone noticed we removed co.za from the system.

We have a new plan here now that after the—well, let me just state do we have any South Africans in the audience? Just you,

Cedrick, okay, and a few people at the back. Oh, sorry, Victor, okay.

The team doing the DNSSEC has also been doing .africa and because we spent a lot of resources and time making sure that things like .africa work properly, DNSSEC has taken a bit of a backburner. So now that .africa is well and truly on the way, time can be spent again on getting co.za up and run, so the current plan is to get it resigned around about the 28th of August and extensively tested so that OpenDNSSEC doesn't fall asleep again.

Hopefully it should be linked into ZA around about the 4th of September, which incidentally is around about the time that in Southern Africa we have an event called iWeek. It's the event run by the ISPA locally and that will also combine with another event SAFNOG, the Southern African Network Operator Group, and that will be held down in Durban, which everyone of course is welcome to come to if you are from the African region.

So co.za will be added. If we look at the other domains, ICANN policy stipulates that all gTLDs must be signed when going live. That's very true, that's what's happened so Durban, Joburg and Cape Town obviously were signed. There's actually very, very few signed domains in those three gTLDs.

Africa is also signed. We are in the last week of land rush now so people will not see anything particularly happening with .africa for now but of course it is signed.

So what else have we been doing with DNS and DNSSEC? We had an ICANN meeting in Cape Town in 2003 or 2004 back then. That's when we saw the need for having DNS and DNSSEC training and for every year after that until about two years ago we've been doing training sessions twice a year both in Cape Town and Johannesburg, and that's trained about 300 odd people.

Since then some people have been on multiple training sessions and I would like to think that the DNSSEC validation that's been happening in the South Africa region is because people like Telkom, our local incumbent, has been on the training courses and it's the Telkom people who have put validating servers on the network and I believe that's been one of the successes.

One of the other successes is actually sitting to the right of me and he will be presenting a little bit later on. Right from the beginning since we've been running EPP, EPP has allowed access to the managing of the DS records and that's been going on for some time. So although co.za is not signed, I certainly personally have DS records in the net.za, the web.za, and the

org.za domains so our customers do have access to the second level.

So the map that you were shown at the beginning in my mind is slightly up-to-date. We are operational. Because of the training, we do have people's skills already built up as well and I think that's an important thing of DNSSEC employment.

My last slide, Repercussions of what happens. Like I said, I personally sign domains and I allow other people working through my systems to also get their domains signed and DS records to be inserted.

One of my customers was running a .joburg and he simply moved it to a different provider and now it is broken if you try and use DNSSEC because the new supplier is not DNSSEC aware and I don't think they know it's broken and they haven't gone in to remove the DS records that I added.

And lastly there is one co.za domain and that will be in the same boat as well. It was signed. It was sitting with me and then was simply moved to a different service provider.

So my take on this is even if you don't do or you don't support DNSSEC, it can certainly still bite you. There is no ICANN policy that states that when a domain is transferred the DS records

should be removed, and perhaps that's something that needs to be looked at.

As it possibly needs to be looked at from a ZA regulatory point of view as well, I am aware that some ccTLDs do insist that DS records should be removed if the new provider cannot do DNSSEC but that's how it stands today.

Thank you very much for that and we will move on to one of my other successors. Sitting to the right of me I have Heinrich Strauss of Strauss Consulting. He's not going to be using slides. Over to you, Hendricks.

HEINRICH STRAUSS: Hi, everyone. This is my first ICANN meeting. I'm quite happy to see all of you, a couple of names and faces that I've really wanted to meet for a long time.

I took the DNSSEC training from Mark back in, I think, 2011. I was working at an Internet service provider in South Africa called Internet Solutions and theirs was just sort of there was this new thing security related. I definitely just wanted to see what this was about.

After doing the training, it became quite obvious that with the correct [prices] and the correct technology in place, this was going to be quite a simple thing to deploy. If things break, at

least I've had a bit of training and a bit of hands-on on seeing where the possible flaws are.

I'm running about 50 domains in my personal capacity split across a bunch of TLDs but probably about 30 to 40 of those are under .co .za, and I've chosen Mark as my registrar simply because when we speak about DNSSEC that's pretty much the only ISP that seems to understand what you're on about in South Africa or at least was at the time. I haven't rally investigated after that.

The only major issues that I've come across are sort of at the point of key rollover, when you've got the key signing, key rollover, that sort of—we kind of try and schedule once a year at least. If you run sequence up properly, and this is completely my fault, and you don't test it properly, you end up with having to go unsigned for awhile while you fix things up.

That is probably the most encountered thing I've seen so far. The only other time that there are issues is when there are zone cuts into like private zones of multiple possible zone files of record, I suppose. These are things under [inaudible] .arpa and things like that where they're not technically valid as the single source of [inaudible].

So those would have been the only two major issues where I've seen any issues with the signing side or with the resolution of the signing side over the last five, six years.

The other thing was in terms of DNS response sizes, for South Africa I know I haven't seen any specific networks where we have issues with small MTUs. So I haven't seen any issues with retrieving records from DNS on my side. I actually have had quite a good time with rolling out DNSSEC wherever, resolving servers I tend to put them on able validation as soon as I have deployed them and new zones are pretty much signed immediately.

One of the things that's been helping over the last while was the ISC's DLV, the DNSSEC look-aside validation source. It's been quite good for things such as [inaudible] back when things were – so, while things unsigned so it's sort of good for that. And also for reverse records, if you're doing things like IPv6 tunneling. Because in South Africa a lot of internet service providers also don't really provide native v6. I think Mark does on his side, but most other consumer ISPs don't provide it and I don't even think very many of the business-based ISPs provide IPv6. So it's just for things like trying to play with new infrastructure before it actually becomes publicly accessible in South Africa. It's definitely at least as easy as most of the other things out there.

So it's interesting that Mark's actually mentioned co.za and OpenDNSSEC. I hesitate to say this because I don't blame the authors of the software at all, it's a fairly complex process and I tend to skip over large chunks of the README files. So every once in a while things kind of go haywire. What I can say is at least that if you've taken backups of whatever databases on the config that you're currently using, rolling back to a state before uploading the new DS records to the parent zone generally works out reasonably well at least in the interim while you sort things out.

So I must say hats off to the guys doing OpenDNSSEC and even on the bind side, the manual side, I mean, I use that for one specific zone which is where I add my dynamic records. That's also been fairly lackluster in terms of actually having major issues after the initial set up of that.

So far I kind of struggle even within the security communities within South Africa. I don't think there are very many people who are interested in rolling out DNSSEC as such. That is something I'm trying to at least address with the guys. We have a couple of monthly meetups and we discuss things like that. There's an OS Group in Cape Town and we're trying to pitch the idea there.

I think as long as we have quality solutions like the automatic signing on BIND or OpenDNSSEC, it becomes quite easy to convince people that, apart from the initial kind of headache of getting it set up everything just flows nicely after that and if you do have any major issues there are resources, at least, locally that are able to help you resolve that.

I've been a student for about four years now and have been kind of missing from the community so if there is anybody in South African community that needs any help with this, I'm more than happy to help. And that is pretty much what I'd like to do for the next [world] is help improve connectivity at least in Southern Africa, but hopefully a bit further across the African continent. Thanks.

MARK ELKINS: Thank you very much, Heinrich. We'll take questions at the end of all of these sessions. Our last presentation, another student of mine. His name is – well, I call him Abdala. He runs the MASR gTLD, the Arabic gTLD from Egypt. It's a relatively small zone. He came to see me – he came on the course and within six months it was DNSSEC signed. So very successful from that point of view. We have his slides and then we're going to be playing a recording for each slide. Then I do believe he's also in the Adobe

Chat Room if there are any questions afterwards. Can I ask you to continue with the presentation?

ABDALMONEM GALILA: Good morning, I am Abdalmonem Galila, working for the National Telecom Regulatory Authority of Egypt as a Deputy Manager .masr IDN ccTLD. The Arabic domain name must mean [inaudible] as misr or masr. My presentation will be about the challenges we had through the deployment of DNSSEC. Actually, before and after the deployment. Next slide, please. Before the deployment of DNSSEC by six [masr] we didn't have an idea about DNSSEC, at least we should know what is DNSSEC and how DNSSEC works. We hear the word DNSSEC at Africa Internet Summit 2014 in Djibouti, from there we started thinking about the word DNSSEC and want to know some information about DNSSEC and [inaudible] through attend the DNSSEC Workshops.

We have two environments for .masr registry system. The first environment is a reduction environment which consists of a registry server, database, [CDNS] server Whois. And the second environment is for testing which is identical to the reduction environment. We still want to have a certain information about how DNSSEC works, so we need to build fake root server and the reason for was DNSSEC validation enabled at our testing environment to see how DNSSEC works. We did that and we

[did] have the information required to start the deployment of DNSSEC. At our present environment we signed as root of .masr and all [inaudible]. But our version of registry [inaudible] was an old [version] regardless to the ability to add DS Records for the [inaudible] domains through the registry interface. So upgraded the registry and it was working good.

One of our DNS servers time was delayed by 10 plus minutes. So we must make our system time synchronized so we build a time server and [make] all our server as clients to the server. We tried to be familiar with DNSSEC troubleshooting before we went online with DNSSEC. For your information we only take 15 minutes to reconfigure the reduction environment to have DNSSEC deployed. Next slide, please.

After the deployment of DNSSEC and before adding our [inaudible] we should know how to keep our system online all the time without signature examination. So we did a script for deciding the zone after – the news originated from the registry system with the capability to look in the area for the administrators for running [the keys] we didn't do any key rollover until now. But we will do during this year. Our mission now is to spread the word DNSSEC to our four local registrars. Next slide, please.

Also we noticed that one of our DNS server doesn't respond to TCP queries and it was an issue with the firewall and you already fixed it and it was working good now. The MTU, message transmission unit, most commonly found in the core of the internet is around 1,500 bytes and even that limit is routinely exceeded by DNSSEC signed responses and mainly took elements limit through the size of MTU may make the signed response to be dropped. So to avoid that, we signed our zone with the parameter minus X to only sign the DNSKEY [RSSAC] with key signing the keys and dummy signatures from zone signing keys. Most of our ISPs resolver don't have the DNSSEC validation enabled untill the moment although it is only one line of configuration. Next slide, please.

As a registrar, most of our registrars ISPs they think that it is difficult to sign their domains and not urgently need to do that. Also they have some [zones] like “Ah, their system is stable so why they have to change the current system? More time required to resolve domain names although the security [inaudible] will be better after the DNSSEC deployment. Domain [with] invalid signature will be blocked. They have already attacks but they can't mitigate it. The registrar don't have an idea about DNSSEC and the registrar interface doesn't have the ability to send DS Records to the registry system through [EPP]. They don’t have enough staff to monitor and troubleshoot DNSSEC –

JULIE HEDLUND: We are checking to see what has happened with our audio. Hold on, please. We will check with the speaker. Hold on. Yes. We've confirmed that that is the last line, so let me turn things back to you, Mark.

MARK ELKINS: Okay. That took me by surprise, I was expecting to say, thank you very much or something. Anyway, thank you very much for that [Abdala].

So that is the full local presentations. I suppose I should also add that there are other second levels certainly in South Africa, for example, I personally look after something called edu.za which is for other colleges or schools of further education needing an FET or further education or certificate. And I believe that has been put into the ZA zone as well looking at Cedric. Okay. Anyways almost, it's been signed for a while and it's almost in the ZA system.

So do we have any questions for any of the presentations, please? Yes, Eberhard, I recognize you.

[EBERHARD LISSE]: Thank you. Almost everybody does. I fully agree with what Alain said, putting it very nicely. Basically, it's laziness. We are just too lazy to sit down and actually do what needs to be done. A gynecologist can do it in six weeks sick leave, it cannot be too difficult.

But I actually have a question, OpenDNSSEC we hear it's been used widely and it crashes all the time in different way. Kenya had a problem, you noticed something. I use OpenDNSSEC with the card reader and to sign in the card. It's extremely reliable, I just have to kill the process everyday at least once because the underlying driver doesn't work properly. When I do that it's a hundred percent reliable. Now, I'm wondering could we not sort of engage some of your younglings to take OpenDNSSEC and take the tools apart and put it into smaller little pieces that you can then script? The software what it does is actually good, but my problem is that it's a big daemon organized things, you have to run the daemon, they die, you have to restart them and things. Would it not be possible that somebody commissions a few youngling programmers to have a look at the source code and then rip the pieces out and make it into small little programs that you can then control [inaudible] so you have much, much finer [inaudible] control. That would help a real, real, real, bit, bit. Because then we could have cheap hardware signing. We can sign up to about 3,000 keys per hour with one

card easily inside the card. And then we have a $50 hardware solution which takes away the expensive [HSM] of requirement.

MARK ELKINS: Yes, I would agree to that. But I don't work for or represent OpenDNSSEC.

[EBERHARD LISSE]: Its' open source. I'm saying could not perhaps some well-funded TLD or provider like you commission some students or some programmers to take the source code, rip it apart and take the bits out into shared – make available as open source. So sort of fork it. But make it available as open source using that as a base without necessarily changing much internally.

JACQUES LATOUR: Jacques Latour, CIRA. So OpenDNSSEC by Internet Labs, I guess that's what you're going to talk about. But if you do have issues with OpenDNSSEC you should open a ticket with these guys and look at it. But we use it extensively and we never add – I don't think it crashes every day. So I'm wondering if other people around here if they're using OpenDNSSEC they see the same behavior, so...

JAAP AKKERHUIS: This is Jaap Akkerhuis from NLnet Labs and we are the providers of OpenDNSSEC lately. It used to be a project between people in the Swedish Registry and some people over there and we were only doing lip service at that time. But we now have inherited the whole project due for political reasons and we're cleaning up a lot of stuff. I mean, very active development is going on in OpenDNSSEC, too. And we are really urging people to talk to us, put yourself on the mailing list and there's a very active discussions about all the problems. The nature of how Eberhard describes this, apparently there's something problematic [inaudible] to drive it to the specific card you are using there. And now, OpenDNSSEC is only talking the protocol. Also these drivers are closed so there is not a lot we can do there apart from trying to find workarounds.

But again, I'm not personally involved in the development, but very close to the people who are doing it. Come talk to us, let's see what happening and let's try to solve people's problems here.

UNIDENTIFIED MALE: Sorry. Just anecdotally from my side I remember the very buggy crashiness from the 1.x series. I must admit since I've gone to 2.x it's been pretty much fire and forget, but I might have some [inaudible] jobs running in the background. Long story short, I

remember it as a Legacy issue, I don't think it's as big an issue in the 2.x series, so thank you very much for development on that.

JAAP AKKERHUIS: Yeah, what we basically had to do is [inaudible] rewrite the internal software. I mean, the quality was not all that great as we ourselves expected of some of the files. So that took a long while to clean these up and we're very active trying to clean up even the current or long-term version of OpenDNSSEC and [active] working on the second version to be much more stable. Talk to me if you feel the need.

JULIE HEDLUND: This is Julie Hedlund from ICANN Staff. I will read two questions in the chat room from Abdalmonem Galila, who is also our last speaker. His first question is: “How many DNSSEC signed domain names are under .za?” And his second question is why ISPs do not use EPP to manage DS records?

UNIDENTIFIED MALE: I'll take that that's a question for myself. There's about 40 to 50 zones signed under ZA or rather should I say there's about 40 to 50 domains that have DS records associated with them, including CO.ZA which is not quite signed, so it's about that. Why do registrars not use the EPP? I have no idea.

JULIE HEDLUND: Actually, he was asking why do ISPs not use EPP to manage DS records?

UNIDENTIFIED MALE: I would reply on that in a standard 3-R Model Domain Name System where you've got registry at the top, registrars and then registrants at the bottom who are the customers. The registrars are generally speaking ISPs, so ISPs if you can interchange them between ISPs and registrars, most registrars in a 3R system would probably have, well, I assume to have a formal arrangement with the registry and they'll either use EPP or they use the one that looks like a web URL – RESTful, that's it. Yes, thank you, sorry. And I'm sure RESTful handles DS Records probably as well.

DAVID LAWRENCE: I just wanted to point that this is going to be a significant part of what we're talking about at 11:00 with this whole issue, so.

MARK ELKINS: Well, if there are no more questions, then technically it is relatively close to 10:15 and according to the schedule we have a coffee break between 10:15, so back to you.

JULIE HEDLUND: Thank you very much, Mark. And so, everyone, we do have time for a coffee break. Now, it looks to be 20 minutes. The breaks are I think if you go out to the right and then I think it's to the left there's a coffee break area. That runs from 10:15 to 10:30, but I imagine it's probably getting set up now. But we will start promptly at 10:30, so please do come back. And again, please be sure you keep your program with the lunch ticket so that you will have it for lunch after the workshop ends. Thank you very much.

[BREAK]

JULIE HEDLUND: Yes. So, everyone, we're going to go ahead and get started, I've got 10:31. So we do want to keep people on time here, so we're going to move on in the program and we're going to move to a presentation. And this is a presentation by Andrew McConachie from ICANN staff and he's going to talk about Danish, Middle-Box Dane validation for HTTPS. So please everyone come back in, obviously bring your coffee with you and we'll go ahead and get started. Thank you.

ANDREW MCCONACHIE: Okay. I'll just go ahead and get started. My name is Andrew McConachie. I'm going to be talking about a program I wrote for Open WRT and LEDE called Danish which does some DANE validation for HTTPS only. And I called it Danish partly so I could display a pastry here while you're drinking coffee. And also because it's not really DANE, it's kind of Dan-ish, so that explains the pastry. Next slide.

So what it is, is it's Daemon for validating HTTPS DANE from RFC 6698. It's written for Open WRT and LEDE, both of them because they're kind of the same thing, but kind of different. It's pretty experimental at this point but I've been running it for a few months and it seems to work. It uses local hosts as DNSSEC Validating Resolver. I developed it against DNSMASQ, but it should work against anything like Unbound or anything else over in Open WRT or LEDE.

And before I get into it too much, here's kind of a question I've been asking myself as I've been developing it with HTTPS which is really that most of the HTTPS connections I see originating in my little apartment are not coming from browsers and so that's – you know, we usually think about Dane and HTTPS being kind of in browsers, but does it really have to be. And that' s kind of a philosophical or interesting question I've been asking myself as I've been doing this.

So what does it actually do? So what Danish does is it snoops TLS ClientHello and ServerHello messages. And from the ClientHello, it grabs the SNI and from the ServerHello, it grabs the Cert., the X509 Certificate. And then it does a TLSA Lookup and then if everything's kosher and they match according to our RFC 6698, Danish does nothing. If they don't match and it looks like the certificate does not match what's in the TLSA record for whatever reason, Danish installs some ACLs. It does this because it's just basically running on Linux, so it uses some Linux kernel modules to install two ACLs which live for a short time which kill the TCP connection and then it installs one ACL for a very long period. A period equal to the TTL of the TLSA message to kill all egressing TLS set up messages that have a matching SNI. So you can actually program an ACL to do like a string match in a packet. And I just do a string match for an SNI and then drop those packets.

So here's a little bit of a diagram based on time of what's going on and the only thing here I kind of want to point out is because I've written this on a platform where there's local hosts as DNSSEC Validating Resolver, I want to fill that cache as quickly as possible because I want to be able to win the race condition of being able to block traffic before the TLS connection is set up. So I do a TLSA Query when I get the ClientHello, when I have the SNI. And then I do it again when the ServerHello comes back.

And on that second one hopefully the cache is already populated because I go to local host caching, DNSSEC Resolver. So it should be really fast and I should be able to win the race condition there. And then if they don't match, I install some ACLs and kill the connection. And again if they do match and 99% of the time I get an ex-domain when I search for a TLSA record. I mean, right. But when they don't match, ACLs are installed when they do match Danish does nothing.

See, here's a current support. The TLS supports I think pretty standard. It also supports IPv4 and IPv6. The actually DANE support is limited because I don't actually have a working Open SSL Package on Open WRT or LEDE that supports DANE. That version is, I guess, still too experimental for those platforms to support. So anything that requires Danish to parse ASN.1 records. It's just I'm not going to write an ASN.1 parser because I'll probably just screw it up so that's for the future when the DANE Open SSL is better supported on Open WRT and LEDE.

That's it, that's all I got. Any questions? Anybody curious more about this? I'm curious to hear people's responses about this thing. Can we get mics? Oh, cool.

UNIDENTIFIED MALE: So first off, that's kind of awesome. I really like that as an idea. Do you keep stats like how often you actually find TLSA records and what stuff and how often it hasn't?

ANDREW MCCONACHIE: So fortunately, I mean, this is something I just run in my apartment. The only broken TLSA record I found, where broken means it doesn't match the cert are the test ones that I've set up myself. And, unfortunately, I can't support a lot of the ones because they would require Open SSL because a lot of people are doing like DANE TA. I think freebsd.org does and the ITF does that. I mean, my stats would be really boring because right now this has a deployment of one.

UNIDENTIFIED MALE: And are you planning on contributing it back ever or something into upstream?

ANDREW MCCONACHIE: Yeah. I'm turning it into an Open WRT package. So then anybody can build an Open WRT setup and then just run this thing. It's dependent on a couple of other Open WRT packages which I'm also maintaining, which I'm waiting for the maintainers to actually like merge. So it will be merged eventually, but it's not merged yet.

JULIE HEDLUND: More questions for Andrew? Anyone? And I'm looking, there aren't any in the chat room right now, so everybody please join me in thanking Andrew for a very intriguing presentation.

JACQUES LATOUR: All right. So the next presentation is a panel discussion for CDS, CNS, CSYNC. So what's the policy in back of all this stuff? Can I ask the panelist to come up front? All right. And Jaromir from CZ also wanted to participate so he's going to sit in. He's a last minute addition.

Okay. So today it's kind of an experiment. What I'd like to focus is this is a policy meeting and what I've noticed or what some of us noticed is that there's a lot of technology being developed like CDS, CSYNC, CNS that there's RFCs for those protocol. But we haven't looked at what the policy impact, and everybody says, "Well, you can't do this or that, because you need to look at the policy and it's got to be in sync." And the purpose of this workshop is to figure what policy do we need to look at and who do we need to get involved to make this technology work.

So the panelists we have today is Erwin Lansing from .dk, myself, David Lawrence from Akamai, John Levine and Paul Wouters.

And so we're all involved and also Jaromir from .cz and he's going to talk quickly about new initiative he's done here.

So this is basically what I talked about. So CDS, CSYNC, their technology, I'll just do a quick overview. But, we're going to cover that in more detail. But, basically it's communication between the child and the parent. And it changes the DNS registration information potentially in the registry and it does that potentially without involving the registrar. So this is what we need to look at. And the DNS operator, the role that the entity – that entity initiates the changes but they're not recognizing the framework. We need to figure out where they fit. The goal of this is to figure out what policy we need to look at.

So today this is a standard model that we have, you know, the standard registrant, registrar, registry framework. So we're modifying, there's new people and this is the framework that we had a long time ago and it changed over time, but it looks like this. So we had the registrant, registrar, registry DNS, but then we have the hosting provider and then we have resellers and then we have CDN content delivery and there's DNS operator role in there. And they have DNS information that needs to be modified to the – no more channel, there's no way of getting this updated.

So in the past ICANN DNSSEC Workshop there's lot of presentation on details, issues relative to this framework. But what we're looking at is we need a new way for DNS operator to update the domain that they manage and so we'll cover some of that today.

So there are new flows. So there's CDS, there's a DS record that potentially can change or I should have added CDNSKEY also. And there's also CNS or CSYNC which a child tells a parent to update his – the child can instruct the parent using CSYNC records.

So these slides are available, I'm just going through them quickly. But what we need to look at is a DNS operator being able to tell a registrar that, “You know what, we need to change DS or nameservers.” So that's a flow that we need to look at and what are the implications of this.

The second one is a DNS operator that might go directly to a registry and tell the registry to update the nameserver and CDS. So that's another thing we need to look at, there's two ways of doing it.

This is an observation in that large registrars today they think of themselves as a registrar, but they also have within the same business a hosting role and a DNS operator role. And what I see is that they merge, the DNS operator, the hosting and the

registrar function, they bundle that into the registrar. But today, their DNS operator role needs does what we're talking. So, part of their business, the DNS operator role for large registrar, they update the registrar portal or system to update nameservers and DNS. It's not the registrar on its own the registrar function that updates the registry. It's their DNS operator business that updates the registrar. So, a large hosting company, they got this model built in.

So, the discussion is – so Patrick provided this feedback. Today, EPP model is that the registrar and the registry, they need to be mirrored. And if the registry is updated by DNS operator, then we need to make sure that everything is in sync. So that's the data flow.

And then if we have information that's out of sync, how do we fix it? So, it says we have ton of policy development work to do. So, I'd like to know what we need to do and then maybe we can work on that in a structured way and make sure this is done, so that we do this in a proactive way of implementing this technology instead of being reactive and telling people we're doing stuff that we're not supposed to or it hasn't been taught of properly.

And so, we need to make sure, very important, that we don't deploy this by accident but that we actually figure out how to do this properly.

First panelist would be –

ERWIN LANSING: I can go.

JACQUES LATOUR: You want to go? The order? Erwin? Let's start with Erwin.

ERWIN LANSING: Okay. So, I'm going to talk a bit about something we have been doing for quite a long time without needing all these new fancy DNS workers. We're a ccTLD for Denmark and as many ccTLDs, we do allow things in our own way. There lots of opinions of what's good and bad, but here, we got one quirk that we can actually use for good especially with the operation outpost, the nameservers and DNSSEC.

We don't operate classic RRR model. All new domains have to be registered through a registrar. But after that, it's based on the role a given person has to the domain. So, there are different roles. There's of course the registrant, there's an administrator,

there's a billing contact, and there's the nameserver operator role. The registrar can of course put himself on all these roles except registrant. And it looks a little bit like the classical RRR.

So, what does it mean for this panel? All nameservers have to be registered at the registry level. We do not delegate the domain unless we know the nameserver already in our system. This way, we also can determine if we need to spool a glue record into the zone if the domain is in the .dk.

This also means we have direct contact with the nameserver operator. This means we could do fancy stuff like send notifications if the domain is suspended. Or if moved to another nameserver, then we can send an e-mail, “It's now time to remove the zone because you're no longer an operator for this zone.”

So, here's one of my domains. There's the different roles, I'm the registrant, I'm the administrator. I want my nameservers. And on the right, you can see there's also a contact for those nameservers so you can actually find out who is wanting the nameservers and get in touch with those.

This also means that nameserver operator has a login to the registry and can talk directly to the registry through all the different channels we have. There's of course our self-server

portal. We have little servers that was there before there was rest, called the DSU. I'll show a little bit how that works.

And we also want to open up EPP for nameserver operators to do those things they can do from their role as being a nameserver operator. So, it won't be a full EPP, they can't register the domains but there are things they can do through the self-server portal [inaudible] to also be able to do automatically.

So, they can, of course, manage their own nameservers, they can give them new names, they can delete them, they can move their IP addresses and manage the glue records. They can also do what we call the internal redelegation. That is, they can move a domain through different nameservers they own themselves. They don't have to ask the registrant. The registrant probably doesn't even want to know, they just want the domain to work and they don't care if it's nameserver 2 and 25 or whatever nameserver it is. They just want the operator to make sure it works. And one of the things we added a year and a half ago, I think it is, that they can manage the DS on behalf of the registrant.

DSU is, like I said, before there was a rest so it's a [inaudible] post. Just give the login information of the nameserver operator,

to domain name and to metadata for the key to add to the domain.

So how does this work? Of course, the good is the nameserver operator can do all these technical things on behalf of the registrant without explaining to the registrant what DNS or DNSSEC is. It also works if the operator is not a registrant himself. And it also works if the registrar for the domain name does not support DNSSEC for a given TLD.

We also use its two-letter DS records follow the domain operator or the owner of the DNSKEY. The one that sends the key to us is the owner of that key. If that owner no longer has a role to that domain, we remove the key. But it also means if the nameservers change but the operator stays the same, we leave the key the zone.

Of course, the bad here is that all nameservers have to be registered. If you're a big CDN, that's a lot of work. It's also a problem if a registrant goes to a registrar and says, “I want a new domain name and he points to a nameserver that is not registered yet. The name principal, the registrar has to tell the registrant, “I'm sorry, I can't register domain right now. You have to go to your nameserver operator and he has to register domain at DK Hostmaster before I can give you the domain name.”

And, of course, like all quicks in how ccTLD is run, it's not standard and it makes everybody has to implement different ways both technical and policy-wise.

JACQUES LATOUR: Thanks, Erwin. So, the takeaway is that .dk support DNS operator directly in a registry with nameservers and DNSSEC operation. Right? Okay. So, the next panelist is Paul Wouters. We'll do questions at the end. Please.

UNIDENTIFIED MALE: No, it doesn’t.

JACQUES LATOUR: No, it doesn’t. We'll do questions at the end because I think we're going to have quite a few. Okay, Paul.

PAUL WOUTERS: Can I reach? Okay. Hi. I'm Paul Wouters from Redhat. I've been involved in some of the automatic DNS updating RFCs at IETF although I was not involved with the CSYNC record.

So, the problem with updating the NS, the A and the AAAA record is that it has to go through the registrar. And that's a problem if you're a DNS operator or if you're a registrant, then you don’t

really know how any of this works. You've set up your domain once and you don't really know anything technical about it and you're just hoping that the company that you're paying is dealing with all of this.

And these operators, as was explained before, they don't really have an official role so they don't really have any access to update any of this information. And so, the DNS operator has to go back to the registrant to tell them how to do it. And that goes back to the registrar before that all happens.

So, this is a cumbersome process. It involves humans that are really upset that is not automated. And so, do we automate this in a nice way?

So first of all, it assumes that you have a DNSSEC security in place already. So, once you've a secure bootstrap, you can use a DNS itself to then update these records from the child to the parent. But since you want to have some sort of authentication in there, you have to use DNSSEC to authenticate this.

So, what is being done is there's a new record type called CSYNC. And you specify the serial number of your zone and some flags and then the DNS record types that you want to update. So usually, that is the DNS record and the glue records if you need and the A and the AAAA records.

So, you see an example there is redhat.com IN CSYNC serial number, 3 is the flag so it basically say do it immediately when you see this. And then it tells you to grab the NS, the A and the AAAA records.

So once the parent nameserver sees this at the child nameserver because the CSYNC record appears only in the child zone. So, when the parent sees that, it can go and query the NS records, the A and the AAAA records and put that in the parent zone.

And then the child can detect when the parent has done this and then remove that CSYNC record because otherwise, the parent will keep parsing it and will keep seeing, “Oh, it's nothing. Nothing has changed”. So, it's nice if the child will then remove the record afterwards.

And so that results in this little updated diagram where the DNS operators and the registrants are now cheering people. They're very happy, everything is automated. The information flows from the DNS operator using the CSYNC record directly to the registry.

The registry can then, via some recent EPP extensions, notify the registrar that an update has happened outside of their information channels. So now the registrar is also aware that these records have changed and they can update their own data

so that if the registrant would use the web portal or whatever the registrar has, then they can also present the up-to-date information.

So, I think that is it. I don't even have a question slide apparently.

The question was, "Has this been implemented yet?" I'm not sure there's an implementation of it yet, but I don't think so. It's an IETF document. When the document has been moved to Internet standard, there has to be two interoperable implementations so we're not there yet.

JACQUES LATOUR: David.

DAVID LAWRENCE: My talk is going to be a little bit redundant because we did do a lot of coordination on this in advance but hopefully it'll add a little bit new insight but it's pretty similar to what Jacques pointed out so far.

Basically, the usability problem that exists is the domain name holders need to get their delegation related information into their parent zone, but a lot of domain holders don't really understand the domain system. And even though they do, they

don’t want to be involved in regular operations so third-party operators are the ones that step in to handle all that.

The current policy environment though is that the shared registry system recognizes basically three actors called the RRR model. As we've talked about before, the registry, registrar and registrant. The formal dataflow relationships only exist between any pairs of those.

And so, there's a missing element there, the Regoperator, another R for the system. Third party operators are really second classes citizens in this entire process. They're not formally acknowledged as ICANN constituents.

So, the policy problem that exists is that operators really need to be able to maintain delegation information. And so, the nameserver and DNSSEC records that normally come from the operator would be the NS record, the policy authoritative nameservers, the DS record that establishes the chain of trust along with the DNSKEY record for DNSSEC. And sometimes also if their nameserver records are hosted within the zone itself, you also need glue address record, both the IPv4 and IPv6.

They require registrant action to update the registrar to push that finally to the registry, and getting registrants to do this can sometimes be a really huge obstacle. Beyond that, getting

registrars to report some records is also problematic. There are many registrars out there that still don't actually really support DNSSEC very well.

So, some technical parts of the problem for a partial solution do exists has been mentioned. RFC 7477 defines the CSYNC record that Paul was just talking about to update names and addresses. And RFC 8078 defines the CDS and CDNSKEY records to maintain DNSSEC information.

Each allows the data published in the child to update the parent. The comment on the slide Paul has actually pointed out to me is wrong and I was confused between an older version of the draft and what finally got the final published RFC. But CSYNC doesn't support any kind of bootstrapping which there's more thoughts on that but probably a little too much for right now, but the CDS, the DNSSEC information actually can be bootstrap. So, it's not that neither supports bootstrapping, it's only CSYNC that doesn't.

But as he also just mentioned, there aren't implementations of this yet. In order to do this, they can be a part of the puzzle for getting the additional support from registrars and registries. But even when they implement these, it's not the full solution as Paul mentioned. For example, a registry might want to use EPP then to push data back down to the registrar to say that a

change has been made. And neither of these documents really address that.

The real situation out there though is it's even more complicated than the RRR model. It's like the RRRRRR model because we have a registry that is informed by, in the case of .com, hundreds of the credited registrars and then beyond that, an untold number. Maybe somebody knows the number but I don't, of independent uncredited resellers that are all constituents and part of the process.

And so, getting the multitude of these middlemen to support new DNS features has historically shown itself to be very hard. Many still, as I mentioned, don't even really adequately support basic DNSSEC functions.

So, the consequences here are that there are a number of unnecessary delays introduced in the system. Manual intervention by the registrant to make the updates can be as quick as minutes but it has been seen to often be as long as days or weeks or sometimes never happen. And this can read to a broken resolution because of several forms of human error that can get in to the process when updating things. And so, this can result in a domain becoming unresolvable for some of our clients or at least very poorly performing.

There's diminished resilience of DNS overall because customers have been known to monkey. Even when the operator has told them, “Use this set of six nameservers,” for whatever reason, they will then go ahead and make their own custom modifications to that set that can be a real problem for resiliency.

This increases workload for everybody because not only do the registrants have to actually do these manual interventions that they might not even want to be doing, but when things go awry, then there is more overhead on users and the other DNS operators and customer service trying to figure this out.

So, we could have a few options here. We could tell as Jacques mentioned, there are some registrars who do provide operator services. And so, they'd be pretty happy with saying, “Yes, you have to do this too.”

One downside of that though is that there are a lot of operators really not interested in being part of the registrar business. If you look at the Alexa top 500 domains, a huge number of them are actually being run by non-registrar operators.

We could have operators interface directly with registrars. One problem with that, I think, is that, as I mentioned, a lot of registrars have not really shown that much interest in

supporting the DNS itself. They really just want to be in the domain name business but not the domain name system business.

Even in this system, it can be complicated depending on which model you use, understanding which registrar is supposed to be properly updated with the data. It also changes, of course, then their model around both of the supported record types of the RFC came up with the CDS model, CSYNC model and the CDS model require pulling from the parents. And right now, neither registrars nor registries are really involved at looking at child side data in order to inform what's supposed to be in the parent or looking at it for any reason whatsoever. And so, for both registrars and registries, this would be a change of their current model with interacting with the DNS.

My personal preference is for option number three, would be to be able to have operators update the registry parent directly, preferably through CSYNC and CDS and so on. And this clearly though needs some policy work from ICANN in order to essentially recognize that this is a function that registries would be providing.

And then our last option is do nothing, just leave it the festering swamp that it currently is. Enjoy.

So just in summary for mind, operators do need a way to maintain this data. There is more technical work that we need. Even though we do have these records, they only go so far, as far as what has already been established. But ultimately, ICANN policy changes really need to happen to get this understood as a normal function of the DNS.

JACQUES LATOUR: Yes. Thanks. John.

JOHN LEVINE: Yes. I didn't bring slides so I got to rewrite my talk on the fly. I saw what everybody else said. I guess I should give a disclaimer although I am a trustee of the Internet society and the liaison from Taugh, I'm not speaking for either of those. This rant is just from me.

So, I'm speaking as a smallish DNS operator. I have about 300 zones, all of which were signed but only about 100 of which you can actually use DNSSEC because those are the only ones for which I actually could install the DS records.

CDS and CSYNC certainly can be useful tools but I think people underestimate both the amount of technical work and the other policy work needed to make them useful. In particular, although

the RFC claims that there is a bootstrap process for CDS, when I talked to the guy who wrote it, there's clouds of smoke.

And I think we need to start by figuring out how secure does the bootstrap process really need to be keeping in mind that the current alternative is typically just the username and a password, logging into a registrar. And if people really cool registrar, it might be two-factor on your phone.

It's what needs to be no worse than a not terribly secure password, and I think we can do that. But we need to decide what we're going to do, and we need to figure out how the bootstrap is going to work. I don't think you can bootstrap CSYNC because if you don't have a nameserver, you don't know where to start from.

And having done that, I think we also need to figure out operationally how to make sure it works even in some of the more unpleasant exception cases. For example, I occasionally have taken over zones from other people where the previous operators disappeared which means that I had a signed zone which works but I don't have the signing key. So, I have no way to rotate to a new signing key.

So, in this case, I would need a re-bootstrap process for a zone that is already signed and somehow say, "Yeah, yeah. I know

you've got this rotation process which is very secure but in this case, we need to break the chain because the link is broken at our end."

That's the most obvious situation but I think there's other ones like that. And given how key this is to what ICANN does, I think we also need to do some fairly serious prototyping. We can't just speck it out and say this will work because if we speck it out without experience, it won't work.

So, with that in mind, I agree with everything that David said about the policy process. Attempting to do this through registrars can't work just because historically, there's too many registrars and their skill range is too broad.

But we need to figure out where the technical holes are and the related policy holes particularly in bootstrapping and re-bootstrapping. Define them, prototype them. And then with any luck, we can make it work and solve this problem, finally. And that's the end of the rant.

JACQUES LATOUR: Thank you. I guess having Jaromir here talk about the .cz experience is going to put things in perspective a little bit. Jaromir?

JAROMIR TALIR: Thanks, Jacques. I'm Jaromir Talir from CZ NIC, .cz registry. I'm not on the program because the changes that we implemented was really on the last minute. We launched the project last Wednesday so I believe that we will have a chance to talk more in depth about that in Abu Dhabi next DNSSEC workshop.

But just a quick update without slides, the project that we launched is actually answer for all these topics, questions and issues that we were talking about during the panel.

What we started to do last Wednesday is that we started to scan the whole .cz zone file looking for CDNSKEY records. The domain model .cz is based not on a publishing DS records but on publishing DNSKEY records. So, what we have to do is to look for CDNSKEY records.

So, when we find the CDNSKEY record for the domain in the .cz zone file, actually the scanning is done via TCP to all the nameservers that we have registered in the registry for all the IPs and all the answers must be the same.

When we found out there is the new record for fulfilling these conditions, we notified the technical contact and we started to scan the same domain for next seven days. And if after seven days, every day, the answer is the same. The last day, we

updated the registry with DNSKEY of the domain and this domain will be secured since the time. And we again notify the technical contact and the registrant.

So, when the domain is already secured, it's much easier because we only check that the proper chain of trust is okay, that the answer will validate so we can easily update the current DNSKEY record for the domain that we have in the registry.

And, of course, I forgot to mention that we notify registrar as well by [inaudible] message. We actually talked about this project with the registrars on our February meeting. We are asking them if they want to support that procedure because they actually can do that themselves.

We need not to interfere if they would have implemented all but they said they don't want to do that and they have nothing against the registry doing that. So that's why we implemented this procedure.

So, we started this Wednesday. We are only scanning right now. The first update will be probably this Wednesday this week so we will see what will happen. There is currently about 150 domains. All of these domains come from Cloudflare because Cloudflare is supporting this almost all there are a few testing. Sorry. I see [Andrei's]. His domain is also signed.

So, the Cloudflare is actually publishing CDS records for all their domain for two or three months already. But as I said, we need CDNSKEY records, not CDS. So, they implemented the change in Cloudflare perform and they started to publish CDNSKEY records for .cz zone particularly.

So, we have these domains. And that will be probably updated with the proper DS records in the [.c] zone. But otherwise, we see that the support for CDS and CDNSKEY records in the software is quite bad that OpenDNSSEC actually doesn't support it right now. I was told that maybe this year, it's on the roadmap that it should be published this year but we will see. And the bind as well, it has some basic support for CDS and CDNSKEY but it's not easy to turn it on.

So, we actually implemented how we think it should be implemented in the new version of not DNS. Not DNS already has automated signing. And from the version that was released in the beginning of June, it's 2.5. It already provides the proper KSK rollover process based on publishing CDS and CDNSKEY records.

So, we encourage all owners of .cz domains to install not DNS and just to switch DNSSEC on and we will take care of the rest. So that's just quick update. I think that more detailed presentation will also on the next meeting probably.

JACQUES LATOUR: Thank you. Just to summarize, so you scan your entire zone on a daily basis. If you find CDNSKEY record, then you create an action to notify the registrant that you're going to add the DS in your zone, basically. And then you wait seven days and then there's no objection from the registrant, then you create that. You bootstrap the domain right there and then. And you do some validation over TCP to make sure all the nameservers be called the same language, good DNS IG. Okay.

I guess we can open up to questions now. I guess the goal is to figure out which policy that we need to look at to address this potential future mess that we're heading toward if it's not done properly.

PATRIK FALSTROM: Hello. Thank you very much for very good presentations. Is it possible to relatively get one of the slides up again? There was a slide that showed the multiple number of registrars and resellers and that was communicating with the registry from your presentation. Yes. The RRRRRRRR thing.

One of the basis why, specifically as a registrar, why I had been talking to various people about the need for actually talking about the policy to probably support this session a lot is that it's

even worse because each registrar is trying to sell domain names to multiple registries.

And as the registries don't have the same EPP implementation, with Denmark as typical example, it's a pain also from the registrar's perspective to actually implement this. So, I think given the pushback and this sort of miscommunication on whatnot between registries and registrars already regarding EPP and specifically on the notification side which is really a pain in the old EPP model to implement because notifications, the whole design of EPP was to have the dataflow in only one direction. And that's why synchronization is hard.

I think that for this to fly anything, I completely support what everyone is saying specifically John regarding finding what is easy is really, really important that all registries implement the same thing. Don't underestimate the bad letter E in EPP.

And I can really say that because I was the Area Director in IETF when we approved this and that was a mistake. So multiple registries as well to show the complete RRR model. Thank you.

ROBERT MARTIN-LEGENE: Can I ask a question? It's regarding the nameserver role and the .dk registry. From what I can see, it has some security issues that you decide to go a little bit easy over some of the security

aspects in favor of getting DNSSEC deployed at customer domains at all.

I don't know what the right solution is but from my .dk domains, I have one of the three DNS services you can find on the Internet and it didn't exist so I had to create it in the DK Hostmaster registry system which everybody does because, of course, the registry operator doesn't want to know about DK Hostmaster.

And after a few months, somebody else with the .dk domain name found the same DNS host and registered that domain names and suddenly I can control the DNSSEC for that domain. That's a little bit of a problem.

So, I think it's admirable that you try to do something to ease the adaptation of DS records. I'm just not sure that it's the right way to do it. And I hope that if anybody else tries to cover the example to at least look very seriously into the security complications of that.

JULIE HEDLUND: I'd just ask if people are asking questions to give their names and their affiliation for the record.

ROBERT MARTIN-LEGENE: Sorry.

JULIE HEDLUND: Thanks.

ROBERT MARTIN-LEGENE: I'm Robert Martin-Legene from DK – no that was before. Packet Clearing House (PCH).

JACQUES LATOUR: Wow. Mark. You want to reply?

MARK ELKINS: Okay. Hi. So, to reply to the DS record update, so there's two distinct cases, right? One is the case where you already have DNSSEC deployed and you have a secured relationship and now you can use that DNS relationship between the client and the parent to update the DS record. And that's as secure as the DNSSEC so that it's assumed to be secured.

Now, the bigger issue is when you have to bootstrap but you're not secure and now you're trying to become secure. So, if you're using something like the CDS record to push that security, then the registry should really make sure it's not a spoof DNS record.

So, it should check over and, like for instance, the .cz people do, you check a number of times over different days to make sure

that whoever is able to send that record out that is not signed can do so consistently for a week. And if they can do this for a week, then, for all effective purposes, they are that domain even if they're really not in their attack or spoofing it, if they can spoof the registry for a week, then they basically are the domain to the world view.

So, I don't think it's a security issue if you go from unsigned to signed using a CDS record. So, some registries, I think .ca has an implementation where they do want to do some additional checks to make sure that it's coming from a nonparty.

And maybe we can standardize those kinds of checks, that would be good. But there's been quite some resistance actually from the registrars that doing anything outside of the RR model it is considered evil. And so to get something standardized is actually very hard.

UNIDENTIFIED MALE: Sort of rather than beating around the bush, it looks like parents ignore children. And that's probably a good thing most of the time unless spoken to by EPP. So it looks like something simple needs to be done so that children can tickle the parents. And it sounds like an unauthenticated system needs to be put into place, a bit like a restful query to have the children talk to the

parents and then the parents can turn around and go and check the children.

That would sound a lot easier than scanning a whole zone to see if anything has changed. And I don't think it would be open to too much abuse.

MARK ELKINS: So I'll say that's only one way to do it. Although even with the existing system, you don't have to keep scanning the whole zone. Of course, it's either one query you could track by SOA. But if I – so I have a question for the room, for anybody who might know the answer. We've been talking about this desire for policy changes to recognize operators as a constituency and be able to support something like this.

I don't actually know what the formal process is. How do we start trying to get a policy change at ICANN? Should we be bringing this to a particular group? What's the process?

JACQUE LATOUR: That's why we're here right now. Because I'd like – so we're not leaving this table until we know where to start. So which policy do we need to look at? Where's the beginning of this? So we have a bunch of technology we're implementing in different ways. We have a huge problem with multiple registries, multiple

registrars, resellers, DNS operator. There's multiple issues we need to address. How do we get DNS operator recognized in this model? How do we make this? Everybody says tons of policy to be worked on. Patrick, where –

PATRICK FELDSTROM: Okay. And then I hand the microphone back to Warren. I think from my perspective that I think the DNS-based solutions with DNSSEC without the bootstrap, I think starting there and trying to find a solution that technically works both regardless of whether the communication is to the registry or the registrar which means both sort of both flows.

Second piece is to resolve the issue with the notification in EPP and I have one way of doing that. And then as sort of an extra additional sugar on top, using John's methodology to think about easy enough bootstrap for each one of the systems. But maybe by dividing the problem in the – sort of in the straps, it might be a way of moving forward. But this is just as a strawman thing and to get the discussion starting. Thank you.

JACQUES LATOUR: Warren?

WARREN KUMARI: So kind of going back to an earlier discussion. So myself and Oliver and George Barwood was the original authors of the CDS, CDNSKEY stuff. And while we were writing that, which is back in 2014, there was a fair bit of discussion on doing automatic enrollment using CDS and similar. At the time, and I think it's still the right sort of consensus, people were really scared of doing the automatic enrollment using this for bootstrapping.

The fact that a domain is owned by somebody for a week doesn't really prove that they actually own it. There are a huge number of domains that people forget about the nameservers go walk about. They then get used for something for a while, and then suddenly someone realizes, "Whoops, that was my domain. I should probably recover it."

The fact that now, you can use the fact that you own a domain to do something like DANE or ACME to get certificates for it, kind of makes this a bunch worse. Going to a website and seeing "I own this domain and I have some content here" has very different implications than going to a domain, getting a lock symbol, and then feeling happy to input your credit card info and similar.

So I think that there needs to be a lot more discussion around the security policies of sort of using CDS for a bootstrapping

method. And I suspect that Ted has something to say about the ACME site.

TED HARDY: Wow, you must have been speaking very softly. It's a big stick. Ted Hardy in my guise as one of the Chairs of ACME – Ted Hardy, in my guise as one of the Chairs of ACME. And we're currently in the final stages of standardizing the automated certificate management and protocol. So please review the document if you haven't done so recently. But it's a challenge-based system where the person who's attempting to enroll using ACME to get a certificate must meet a challenge of the choosing of the CA. And so the CA browser forum and others have worked with us to make sure that those challenges are not something where the proposer can actually select which of the challenges are going to be used in cases like this.

So there actually are pieces of ACME that are built to be resistant to the problem that somebody may control certain aspects of the domain name system for a period of time. That said, if you keep going to new CAs, you may will eventually find one who says, well it turns out that what I really want you to do is to populate this DNS record, right.

Then you're reliant on other people noticing that maybe that wasn't the right CA to match up to that domain, which puts it off

into the realm of strict transport security and other things. And I think to that extent, Paul's point was much the same as the working groups. Which is if you control the DNS name for a certain period of time, you are already capable of becoming a man in the middle to so much that the extent to which somebody else owns the domain name is somewhat notional unless they have already gone out and secured in other ways.

And so one of the things we are trying to do with ACME was to make it significantly easier to secure it in one of those ways so that that becomes a stop to random choices like this. But I caution you that the challenge response mechanism in ACME is not quite the same as the CDS model and you shouldn't assume that there's a one-to-one mapping between the two.

JACQUES LATOUR: And let me just address Warren's point as well. Because Warren said because of the new ACME now, we can do this and these things. But before there was ACME, there was also if you had the [AMX] record, you control the e-mail, you can do anything to the DNS that you ever wanted. So it's not really a change. It's a little easier now, but it's not really a big change.

WARREN KUMARI: So yes, that's true. Whenever you've had control of a domain, you could go and get a certificate. However, for a long time, we've been saying, that's kind of sucky, right? We've been saying it's really bad that the ability to control the domain means you can automatically get a certificate. So saying we now meet the same low bar as CAs for doing validation doesn't really put you in a good boat. It puts you in a bad boat.

JACQUES LATOUR: We've had this discussion over and over about the bootstrap part. The Internet draft for the DNS operator, RRR model. It's a restful EPI that DNS operator can trigger to tell the parent to go look at the CDS or the CDNSKEY to bootstrap. And in there, there is option for token and secret tokens and different kind of challenge to ensure that the DNS operator actually controls the zone in question. So that's one aspect of it.

The second aspect is once the request has been made, the parent. It's the parent should have a policy to check the [IGin] of the domain to make sure that it's actually the right place. So you need to have two different nameservers and two different AS and you pool everything over TCP. And you make sure all the nameservers listed, all respond the same, and they're all saying same with the same DNSKEY.

And then if you do that for a week, potentially it means somebody has to hijack two different prefix and two different AS for the same amount of time for a domain. That's a lot of work to compromise a domain. It's a lot easier just to [steal] the registrant credential, to log in the registrar and change it there. So there's need to weigh it in, to balance the pro and con for that part. I think we have addressed some of that in that Internet draft.

But the question I have is, which policy do we need to look at to get the DNS operator authorized to talk to TLD registries to perform activities? So still don't. I don't think they have the answer yet. We don't, right? Or nobody knows or…?

I got another observation question. After doing all of this, it came – it occurred to me that the DS record, there's been lots of discussion around that. Is it possible that – I'm questioning, who's the owner of the DS record in the registry? Is it the registry or the DNS operator that owns the DS record that goes in the zone file for the TLD?

If that record is owned 100% by the registry, then there's no need for exchange of information between registrars, DNS operator, and registries. Like in the case of CZ, they control their DS. They go – they poll their zone. They grab their CDNSKEY. They create a DS record. They put it in their zone after all the

test. And they're 100% in control of the DS record. It's theirs to manage.

So what would be the policy to make sure the registry, the TLD operator, the TLD themselves, are the owner of the DS and they're in control of that. How do you process that information that it doesn't belong to the registrant? So that's another question I had.

UNIDENTIFIED MALE: So, in our model, it depends on who sent the key to us. So the registrant can upload his own keys, or the DNS operator can upload keys. So they own that specific key. The registrant is of course still king of the domain so he can override the names of the operator. He could disable his keys or he can completely opt out of DNS, so we never spool the DS records even though we have the keys in our registry.

I would say if we would go for the scanning like CZ, it would be the registry of those keys, but I have to think about that if you would go that way.

JACQUES LATOUR: So again, policy. Every time we talk about this, people say there's tons of policies we need to change. Where, what, how? Maybe it'll be the topic for next, so we need to fix the technology

first and then build policies to support the technology. But what if the policies that get built don't support, don't accept all the work we've done? Then we've wasted a lot of time and effort. So before we go too far, we need to figure out –

UNIDENTIFIED MALE: Right. I don't think we have to solve all the technology part of it right now until we get some forward progress on whether policy changes can happen. And so some of these questions, do we say that the registry is going to generate knowing the DS? Or do we expect it to come from the operator?

That seems to me the finer point of the policy discussion, that either one can be implemented easily technologically. I think the focus really has to shift now that we have kind of the basic building blocks of what could happen is already established. We look at how the policy can make it happen. But I don't know who to take that to.

JACQUE LATOUR: So in Paul's slide, you had a picture of the CSYNC record going directly in the registry. And so that means we need to go to GNSO or something like that and ask them to change RAA to add more specific content. And a new policy to accept DNS operator supplying DNSSEC or nameservers information directly to

registries. There's lots of experienced ICANN people and [that know] how to do policy stuff. All right.

UNIDENTIFIED MALE: So for us, as ccs, we usually set our own policies, which is quite nice because we're quite independent but can it get done in a standard way that works for all TLDs. It's going to be awkward.

JACQUES LATOUR: Okay. Any other questions? All right, thank you. Thank you.

JULIE HEDLUND: Please join me in thanking Jacques and his panel. And hey, it's Jacques up next again. Correction, actually Mark Elkins is volunteering. Thank you.

MARK ELKINS: Mark is not volunteering. He's just feeling very, very sorry for Jacques who also has the next presentation afterwards. So if I get word stuck, I'm going to turn to him anyway. So DNSSEC. How can we help you? So the obvious starting point then is looking at signing your TLD. And as I've seen certainly, planning is crucial to success. Taking your time also seems to make a lot of sense rather than rushing into it.

And there are an awful lot of tools and services available. And a lot of other people available to help automate that particular process. Once you signed your TLD, it doesn't stop there. You need to be able to accept records from the children. The DS records and potentially DNSKEY records.

Personally, COZ the registration or EPP system accepted DNSKEY records. And I know there's a religious war about those two. I think preferably just accepting DS records makes a lot of sense. But make it easy and automatable to accept the records presumably by something like EPP or a restful interface seems to be the way to do things. Until we have policy on how to do it using the CSYNC system.

Work with your registrars. Unless you outreached to your registrars, it's not going to work. Your registrars are the people that are actually probably looking after the DNS for the clients, for the registrants. So I believe that one of the successors that we've had, for example, in the Southern Africa is doing a lot of outreach training programs, teaching people how to do DNSSEC, and getting them doing it hands on etc.

Once you've got DNSSEC up and running, then statistics. That makes things more fun so other people can see how great you are. There's no point doing all this perhaps unless you can blow your own trumpet.

So zone operator? Okay. Zone operators. Yes. So zone operators. The people running the DNS. Sign your zones as well, I guess. And obviously verify that your registrars and I presume the ccTLD, the registry supports DNSSEC. Make sure they know you want DNSSEC.

DNSSEC to me is very, very similar to IPv6. Unless you tell people that you want it, you won't get it. And again, help stats. And then we have enterprises, what can enterprises do. I presume by this we mean the banks and the big operators and the financial houses and the people that are selling goods, etc. on the Internet. It's not just things like SSL certificates. It's bringing DNSSEC, bringing DANE certificates, make the whole thing a lot more secure. Talk obviously with your DNS operators and your consumers. Teaching your consumers that there could be some additional [inaudible] icons that make things like web browsing a lot more safe.

And then we've got this KSK key rollover thingy. Well, there's two things for that. There's obviously it does make sense to potentially change the sign-in key. The default is once a year. But ICANN has managed to leave it for the length of a lithium batteries in their HSM machines. So they've left it for over five years. So one year is not necessarily too important.

On the other hand, if you don't do it regularly, you will forget how to do it and people move from one job to the next. So actually doing it quite frequently and having it properly written down probably makes a lot of sense.

ISPs. Of course, validating. Getting the ISPs to validate. So deploying the DNSSEC validators, it's really easy. On our DNS courses, we actually have an advanced – an intro course and an advanced course. We actually teach people to do this in the intro courses, that simple. It's a couple of lines into on a BIND system onto a resolve.com file. So really, really easy. And signing your own zones. Sign your forward zones, why not also sign your reverse zones as well? And again, the whole key sign-in key. Don't forget that. And do it regularly enough so that you don't forget how.

Steps everyone can take. Use DNSSEC yourself, that's certainly something that I've done. Whether you're an end user, doesn't matter. Whoever you are. By playing with DNSSEC, that's how you've become comfortable with it. And obviously sharing the lessons that you learned from doing these processes, which is what half of the workshop today has been about.

And obviously participate in workshops, meetings like this, and other occasions. And I guess this is a special thanks for all of

today's presenters and participants. Thank you very much for coming and sharing.

And yes, lunch. Lunch – you're sponsoring. You're sponsoring DNSSEC secured. So support the DNS workshop and associated activities at ICANN are an organized activity of the ICANN Security and Stability Advisory Committee. And with the additional assistance of the Internet society with their Deploy360 Programme. Thank you very much.

JACQUES LATOUR: Thank you, Mark. I appreciate it. So now we're almost done. The next step is the – with the tradition that we have now with the great DNS quiz. So if we can load that.

JULIE HEDLUND: So you should, for the quiz, for the great quiz, you will need to have an answer sheet and these are spread around the room. It's the page that has ten numbered lines on it. And so if you don't have one right by you, then I suggest you kind of walk around and see if you can find one, if you'd like to participate. Thank you.

JACQUES LATOUR: The DNS quiz. We've done enough DNSSEC stuff so it's all over. So it's a DNS and DNSSEC, it's part of DNS, right? No? Yes? And somehow I got voluntold to do the quiz this time. So voluntold yeah. So next slide.

So as the tradition is, I'm always right. It's one point per good answer. And it's one answer per question. And it's a maximum of 10. And the same question. And there's an actual chance of people getting a 10 here and not a minus score, huh, Mr. Wouter.

So let's go for question one. So Question #1, what is the date of the KSK rollover? The root KSK rollover. A) October 11, that's probably the right date, 2014. I'm right. So whatever I'm thinking, so you got to guess that. So pick the right date.

Question #2. So which ccTLD was most recently signed with the DS and the root worldwide? A) Argentina, Liberia, Saudi Arabia, or Greenland. If you weren't sleeping this morning.

Question #3. So which of the following is not a DNSSEC related RFC? So RFC 4986 requirement related to: 4509, the use of blah, blah, blah. 4025, a method for… 2325 definition of.

Next. Next question. You're good, yes? Which African ccTLD was the first to be signed with the DS in the root? Reunion Island, .re. .se, Seychelles. .yt Mayotte Island. .na Namibia. Namibia. Yeah? Yeah.

Question #5. Which one best describes DNSSEC algorithm number 14? So we keep talking about 13 but – so RSA/SHA-512. GOST. ECDSA Curve P-384. Or D) Unassigned. Everybody should know that.

Next one. We have enough time, yeah? Which African ccTLD was the most recently signed with DS in the root? So we talked about that this morning. So Senegal. .za, South Africa. .lr, Liberia. .ma, Morocco. So we'll see who was sleeping this morning.

Question #7. What percentage of all TLDs in the root are signed, secure delegation, DS in the root? So A) 95, B) 90, C) 85, D) 80, E) 75. So this morning, we had a slide just with that.

Question #8. Which African ccTLD is signed with no DS in the root? So Liberia. Sierra Leone. Madagascar, and Somalia.

Question #9. In RFC 7477, a record type specify how a child zone in the DNS can publish a record to indicate to the parental agent that a parental agent may copy and process certain records from the child zone. So is it HINFO, DNAME, CSYNC, DS record, or DNSKEY record? I think we've –

Question #10. So what key component of your DNSSEC validating resolver needs to be updated by October 11, 2017? So that's the answer to the previous one. To not affect your ability

to… So the root zone KSK, root zone DNSKEY, root zone trust anchor, or the root zone ZSK.

Next one, so now it's time to pass your sheet to your neighbor. And then we'll do the correction. So it's only one point per good answer. And now we're going to see if I'm wrong. Probably. We'll see.

Okay, so next one. I'll grab that clicker. All right. So what's the date of the root KSK roll over? D) October 11, 2017.

Question #2. Which ccTLD was most recently signed? Saudi Arabia, June 12, 2017. So when I did the slides this morning, that's when I saw they were recently signed so I had to change all of this. I updated. I updated this this morning, yes.

Question #3. Which ccTLD was the – I got the clicker – which is not related to DNSSEC? Coffee pot mib. It's a myth to monitor your coffee machine. We get this in the pea trap when it's ready.

Question #4. The first – Namibia, our doctor that did that in six weeks during his vacation or something. Sick leave. So in 2010, almost 3 years before the RTA. That's good.

Question #5. Algorithm 14 is ECDSA Curve P-384 with SHA-384.

Question #6. Which was most recently signed? Liberia in April. That's the African ccTLD. Check.

90% is the answer. We had that in the slide today. And everybody should know that. 90% of TLDs are signed.

So which African ccTLDs are signed with no DS in the root? That would be Sierra Leone and I forgot to put the date when it was there but – B.

RFC 7477 is CSYNC. I guess we had many slides around that today.

Question #10. You need to update your trust anchor. When the validating was over. I'm always right. It's C. It's bold.

You can't get wrong. The one that complains the most does the quiz the next time. Are you complaining? All right. So let's score with this. So who was the last ICANN meeting, the DNSSEC – the last guru? Who won the last session? You did? Okay. Let's see how many people got 5. Good. 6? 7? 8? 9? 10? 9, that's pretty good. Wow. Congrats. Now we're –

JULIE HEDLUND: So thank you very much, Jacques, for a wonderful quiz and still challenging. So next, lunch will be served. It's officially starting at 12:15 on the level 4 foyer. So you can use the escalators at either end of the hall to go up there.

Now, you do need your ticket. So if you don't have a ticket, there's still – I see that there are still tickets or programs with tickets on the table. There's some on the front tables here. We do have just this certain number of tickets. So I do hope you retained your tickets. But look around you if you don't happen to have one because I can see that there are ones out there. And again, it's level 4 foyer. So 2 floors up.

And thank you also for joining us today for the workshop and note that after lunch, the Tech Day will resume in this same room. So if you want to join in the Tech Day activities that will follow lunch. Lunch ends at 1:15 and Tech Day starts at 1:30. Thank you.

[END OF TRANSCRIPTION]