MONTREAL – DNSSEC Workshop (1of 2)
Wednesday, November 6, 2019 – 13:30 to 15:00 EDT
ICANN66 | Montréal, Canada

DAN YORK: … didn't have to travel that far.

UNIDENTIFIED MALE: [inaudible] [GPRS to GP] registry.

DAN YORK: Why am I doing this? I don't know why I'm why I'm passing this thing around.

[OLBER JURANSON] [Olber Juranson] from the Belgian Communications [inaudible]

DAN YORK: From where?

UNIDENTIFIED MALE: Belgium.

DAN YORK: Belgium. Great. Hey, Russ.

RUSS MUNDY: Hi, Dan. Russ Mundy. Parsons. I do DNSSEC stuff and other security things.

DAN YORK: That's good. And we know this is Kathy who's here. Hi, Kathy. This is what—

UNIDENTIFIED SPEAKERS: You were going to say it.

DAN YORK: What?

UNIDENTIFIED SPEAKERS: [inaudible]

DAN YORK: Oh, this is Warren Kumari, everybody. Warren [inaudible] the hats. Thank you. Only your hat is pretty tame. I was thinking a pointy hat or something.

UNIDENTIFIED MALE: [inaudible]

DAN YORK: Oh, he promises a hat in Singapore at the IETF meeting. Okay. Barry?

BARRY LEIBA: I'll have to consider bringing a hat to Singapore. This is Barry Leiba. I'm on SSAC.

UNIDENTIFIED MALE: [inaudible] I'm with CIRA. I'm with the SSAC and on the DNSSEC Program Committee.

DAN YORK: There we go. This is called vamping while we get the slides up. But we're all good but we can keep going.

UNIDENTIFIED MALE: [inaudible]. I'm with CIRA.

UNIDENTIFIED FEMALE: [inaudible], Fellow.

DAN YORK: Oh, great. Welcome. Lars?

LARS-JOHAN LIMAN: Lars-Johan Liman with Netnod, also RSSAC, and also the Customer Standing Committee.

DAN YORK: That's a lot of committees and things there.

UNIDENTIFIED MALE: [inaudible]

DAN YORK: Yeah. You need multiple hates, a whole bunch.

[MATT STAFFBURG]: [Matt Staffburg], IS, Sweden.

DAN YORK: The Swedes are sitting together. This is the Swedish part.

DAVID: David from Apple.

DAN YORK: Apple on this side, Google over there. All right. There we are.

LUIGI: Luigi, .va TLD.

DAN YORK: All right.

YAZID AKHANO: Yazid from [inaudible], Fellow.

DAN YORK: Yazid came to the DNSSEC workshop. He and I had been following each other around in sessions these last couple days. It's like, "Oh, hey! Hey, Yazid!"

UNIDENTIFIED MALE: [inaudible] from CIRA, .ca

VLAD: Vlad from .ca.

DAN YORK: Awesome. This also the Canadian side because Jacques over here, too.

Welcome, everyone, to the DNSSEC workshop.

UNIDENTIFIED MALE: Dan, thanks for the wonderful introduction.

DAN YORK: You're welcome. Dave, would you like to start singing?

DAVE: Uh, no.

DAN YORK: Darn it. Okay. All right. Well, now we should be all set. Thank you, everybody, for accommodating the fact that the last session ended at

1:30 and this session started at 1:30, which was a little bit of – are we still good? No?

WARREN KUMARI: It's interpretive dance.

UNIDENTIFIED MALE: Quantum computers will fix that.

DAN YORK: Only if it includes a blockchain. Okay. How're we doing? Well, everybody is here to talk about DNSSEC, right? If you're not here to talk about DNSSEC and RPKI and routing security and things like that, you are in the wrong room. You should run away pretty quickly.

What else could we say? Quick, Warren, this is your interpretive dance. You've got some pylons you could stand on? There's a little ball in the middle there?

WARREN KUMARI: [inaudible] interpretative.

DAN YORK: Parquet. All right. So this is the DNSSEC Workshop but you'll find also we have expanded a little bit to have some additional security and privacy elements around it. We were talking a little bit about some routing security pieces in here and some other parts as well. The names

you're staring at are some of the people who are on that page. Would you raise your hands or stand up or something?

Okay. Russ, Jacques, Andrew, Andrei. These are the folks that have been working. We have a weekly call, actually, that helps coordinate these DNSSEC workshops. We do these typically two to three times a year. We sometimes do not do one at the policy meeting at the middle of the year just because of the attendance, but sometimes we do. We are the folks who you can either credit or blame with the lineup of speakers we have today.

Dave, got any good jokes you want to tell us?

DAVE: BGP.

DAN YORK: Okay. That was a deep insider joke for anybody who was here. Not sure how well it scales. Oh, sorry.

Anybody else?

Okay. [inaudible] secured. Jacques, any more poutine recommendations you want to give us?

JACQUES: There is a tour you can register for with nine different kinds of poutine.

WARREN KUMARI: Really?

DAN YORK: It's Montreal. Of course "really."

JACQUES: You shouldn't have breakfast before that.

DAN YORK: And you should expect to go to your doctor the next day for your checkup on all your levels of cholesterol and everything else.

Has anybody had poutine while they're here in Montreal?

UNIDENTIFIED SPEAKERS: Of course.

DAN YORK: Good. Look at that.

UNIDENTIFIED FEMALE: [inaudible] quiz?

DAN YORK: No.

UNIDENTIFIED MALE: It's going to be a poutine quiz.

DAN YORK: I poutine quiz, yes. I moved to Ottawa back in 2000, and when I was there I had never heard of poutine, of course, because south of the 45th parallel we don't necessary do that down in the U.S.

UNIDENTIFIED MALE: 49th.

DAN YORK: Well, okay. Whatever. For me, it's down below that. We don't do poutine down in the United States, necessarily, although it's getting there. There are some restaurants down in Vermont where I live now that do do poutine, but I looked at that at first and it was a truck outside because of course it's on a truck.

Okay, good. I can stop talking about this. Anyway—

UNIDENTIFIED FEMALE: I can actually just let you keep going?

DAN YORK: What?!

UNIDENTIFIED MALE: Is this really what the program [is]?

DAN YORK: This is not the program [inaudible]. Anyway, I was fascinated by poutine. Had it about twice and said, “Not again.”

All right. Here we are. We want to thank the folks who are listed up here. I would just like to give a round of applause for everybody who is here. The companies that you see here have signed up for a year or two to sponsor these gatherings. What that usually means is that there is lunch. Now, you’ll notice there is not lunch here because of the scheduling on this particular meeting got a little odd and strange did not happen. But they signed up to help us with this and we appreciate the fact that there are so many.

On this note, there is going to be tonight at 7:00 over in the Intercontinental Hotel a gathering that has been graciously sponsored by these folks where there will be some light food and drinks and things like that. In order to go, you need to see Kathy. So you need to come and visit Kathy at some point and she will give you a ticket that will get you into that gathering. So please see Kathy to get that.

Again, thank you to all the sponsors who do that. They make us able to have these conversations on a full stomach and enjoy each other’s companies that way.

This workshop is sponsored through the ICANN Security and Stability Advisory Committee, with additional support from the Internet Society. I just realized I’m sitting here holding a mic when I don’t need to be.

If you look at what we’re talking about here and you can see what we’re doing, we’ve got first up me continue giving a little bit of a talk about

some of the metrics we've seen and some of the parts that are there. Then Daniel Migault, who's here, is going to talk about an Internet draft that we've been working on around best practices for DNSSEC validating resolvers. We'd like some comment from this space and some placer on that. Russ Mundy is going to talk about an intro to RPKI.

How many people have heard of RPKI?

A good number of people on that side and a few over here. Good. So we're going to talk a little bit about that. Kim Davies – is he here?

No. He'll be here. He's going to come and talk about an update on the next KSK roll, just for everybody to pay attention to. Yoshiro – I did see Yoshiro. Where's Yoshiro?

Oh, he's over there. Yoshiro is going to talk about some of the research they've done which is kind of cool. After a quick coffee break, we've Jaap, who's going to talk about some routing security. I'm going to talk about a new project we've come up with at the Internet Society called MANRS Observatory. We're going to end with Mark – is Mark here? Liman?

Okay, they're not here yet. They're going to talk about some of the work about domain name abuse they've done in the .eu. So that's the bit. Let's just get underway.

One of the things we like to show in these workshops is some of the latest stats of how things are growing. You'll see that, after a bit of a pause, we're seeing a nice rise in the increasing growth of DNSSEC validation – the checking signature side of things – and this comes from

Geoff Huston and George Michaels and their APNIC Labs team and the work they've been doing to collect these stats for a while. So you can see we're seeing a bit of a jump there.

They also have nice stats. If you dive into it, you can see a bit more about what the validation that's happening in those regions of the world. You can see up here on this chart – the slides are also available online – which parts of the world are doing the most DNSSEC validation.

The other column, the second column, is the percentage that uses Google's public DNS. It's actually interesting. I was thinking about this the other day. We perhaps need to expand that to be using public DNS in some way and see about how we can incorporate the other ones that are there. Google's DNS has been showing up to a huge percentage in some countries.

Warren is looking at me.

WARREN KUMARI: Geoff's got a new page that does that.

DAN YORK: So I'll relay because Warren didn't get to a mic. Geoff Huston has another page that shoes the public … okay. So, for the next workshop – I'm looking at Andrew – we will figure out what the right link is to put in here to show this.

The interesting part around this is, if you look at some of the countries, like the top one there – Micronesia in Oceania – you can see they have

a large percentage of the queries that came out of there that were validating. They also have a high percentage of people using Google's public DNS, which probably means that many of their ISPs are just configuring their systems to point to that DNS. We've seen that in a number of countries and places around the world.

This is some stats. Where is Wes?

UNIDENTIFIED FEMALE: [inaudible]

DAN YORK: Wes is over there. If you haven't checked out the new – well, not new; the new and improved – stats page on DNSSEC-tools.org, Wes has come up with a number of different statistics that are quite interesting and cool to look at. This shows the overall growth of DS records that we've seen. Again, we're seeing a nice growth in the overall number of signatures that are happening around outside there.

WES HARDAKER: Dan, I need to be very clear here. Viktor Dukhovni does all the stats collection. I just present it. So he deserves the credit for most of this work.

DAN YORK: If you're listening, Viktor, we agree. Thank you, Viktor, for coming up with all the stats and things. And thank you, Wes, for presenting it on a page then.

If you follow Viktor on e-mail or Twitter or anywhere, he's pretty routinely putting up new numbers of how many domains have MX and DANE records. It's continuing to grow quite significantly, as we're seeing people deploying DANE records for use on e-mail domains.

RPKI. The stats that were here … We're going to talk more about what is RPKI, but this is some stats coming out of NIST around the snapshot of growth that we could see. We could see there's a number of good, valid ones and a number where it's not found. So we still have growth in there to go.

Here's a picture of how many IPv4 prefixes are covered. I think – okay. Well, just a little bit more. We've seen some new deployments in ccTLDs that have changed over this past while, some new ones that have come around. You can see the list there of what they are.

The actual status. We show this map where we've seen many parts of North America, Latin America. All of these – actually, Andrew, we need to work on this, I guess. You and I need to show more on this.

We have a number of resources that are out there that are available at DNSSEC-Tools. The Internet Society has our Deploy360 components. We have some other pieces. And we've added, this time, some information on RPKI resources. So please do go and check those out. These slides are available from the page on the ICANN66 website.

With that, I'll say thank you and we are ready to go to the first presenter. And I will actually defer to Daniel although I am also listed as a culprit in this one.

DANIEL MIGAULT: Okay, I'm ready. Today I'm going to talk about DNSSEC resolver operator recommendations. This is ongoing work we're doing at the IEFT. We expect a draft to be published very soon. If you have any comments, you're more than welcome to comment on the DNSSEC mailing list.

I guess everyone here is familiar with the IETF? Who is not?

UNIDENTIFIED FEMALE: [inaudible]

DANIEL MIGAULT: Well, okay. So I will explain it to you. The intention of this draft is clearly to provide some best practice to ISPs so that they can enable DNSSEC. That's the high-level motivation.

The trust in … I think, Kathy, you … [I don't know] if it's me. I was thinking, "Kathy is going too fast!" So DNSSEC I guess everyone is familiar with here, but the idea is that the trust in DNSSEC relies in validations, which involves signature validation. A signature is basically a binding between a key and a signature, which is the DNSSEC world means a DNSKEY and an RRSIG. But it's only half of the problem because the validation is meaningless unless you have trust in that the

owner of the private key actually owns the domain and generated a signature.

So how you would bind a key to [that] owner? We use a trust anchor, which is also a DNSKEY. We have a chain of trust. So each DNSKEY are recursively being validated. The thing is that each level of the DNSKEY keys is likely to change over time, so one of the main concerns is how we keep that chain of trust trusted.

Next slide. What should the DNSSEC resolve operator care about? It's mostly that the malicious DNSKEY has been introduced into his resolver, in which case validation will fail for a reason that is not excepted. All these recommendations [are on] providing operational recommendations for those operators so that it does not happen. So recommendations are based on provisioning, monitoring, and management.

Just to recap what DNSSEC validator is, you have a validation engine. It relies on time and crypto libraries. That's for the signature check. Incoming messages are being validated and added into the cache or rejected from the cache. The thing is that it also has what could be a seen a configuration element, which is composed of the trust anchors. Those trust anchors are maybe evolving over time, so it needs to be updated.

The intention of all these requirements is to minimize the possible intervention of those operators. What we do not what to happen is we end up in instrumenting the resolver. The basic idea is: don't touch

anything. So that's what we're trying to move as far as we can: don't touch anything. If you don't understand, it's fine, but then don't touch.

The other thing is that we don't want operators to be afraid of enabling DNSSEC and say, "Oh, we don't really understand. We need experts. It's going to represent additional cost." We basically want to say, "Well, if you follow those operations, nothing bad should happen. If something bad happens, it might not be your responsibility." So each of these ISP or DNSSEC operators should not feel responsible if anything is going on on the Internet.

If you want to avoid human errors, the best way is to automate all the operations while focusing on configuration prevention or early detection rather than having teams being in alert mode. So that's basically what we're going through.

The recommendations can fall into three categories, one which is a start up health check. So, check if everything is fine and then start. Then, when things are running – the thing running is the DNSSEC resolver – regularly check what needs to be regularly checked. Then, of course, if you want to have some specific monitoring, closed monitoring, or you need to have a deeper, I would say, investigation – that's probably not the right term; you want to do something on purpose – that's what might be the operations you're likely to need to understand what is going on in your network.

The first recommendation we come with is time derivation. At startup, it's good, before you start your resolver, to check the time of the machine hosting the resolver and you regularly check how long the

derivation is. You should have a way to request what is the time for each of those resolvers. So it's not something you should do in an emergency use case. So I think that's pretty obvious.

The second thing is trust anchors. We have two kinds of trust anchors. One is the positive trust anchors. It's an association between a key and a domain name that the operator trusts. We also have negative trust anchors. These are only the domain names you don't want to perform the validation. All those trust anchors, negative and positive, are hosted in what we call the trust anchor store. Usually when we talk about trust anchors it's the positives ones. When we want to say the negative one, we specify the negative one.

The management of those trust anchors. Remember, we mentioned their value might change over time. At the configuration, when you're provisioning your resolvers, you should be able to indicate which trust anchor you're willing to trust and ensure that the resolver starts with those trusted anchors. Then, during operations, they're being updated and the resolver should take into account this rollover and being able to roll over their own trust anchors. And you should also be able to, as an operator, check which trust anchors each of your resolvers is considering.

Because of the configuration, the thing we want to prevent is a resolver starting with an older configuration file. The configuration of negative trust anchors could be seen as a two-step process. The operator should be able to say, "Well, these are the domain names I would start and I would consider more reliable." So, the ones he want to be trust

anchors. You need to then, from that trust model, be able to provision the appropriate values.

So, in a way, it's not that you have a configuration file and you should be able to say, "I am trusting that key." You're basically trusting a domain name and then the up-to-date appropriate value should be filled into that configuration file.

So the process we envision is that you define the name you will trust and the name you want to disable validation. Then you have a process that is retrieving, from those domain names, the appropriate values. Then you generate a configuration file of your resolvers, possibly using a kind of generic language. Then you push that configuration file into the many instances of your resolvers. The resolver checks the trust anchors and then starts.

Two notes about this. With such a process, the advantage is that the operator is only defining the trust model, which is, "I trust this domain name and, from that, I'm going to derive all of those or the others." The other thing is that, any time you start your resolver with the up-to-date trust anchors, basically you don't have to update the configuration files of your resolver when you have a key rollover because you only update it your memory but not the configuration file. So, when you restart your resolver, you will have an up-to-date trust anchor and you don't have the problem of, if you have a read-only file system, the up-to-date key [as] not being considered into the configuration.

The startup recommendation is that the trust anchor should have bootstrapping mechanisms. It's a way to be able to check in an

automated way that you have the up-to-date value. This trust anchor must be validated by the resolver before the resolver is starting. And maybe not just sending a warning when … that's the idea.

To implement this recommendation, now are we trying to boil the ocean? That's the question that people might think of, looking at all these steps. If you have a DNSSEC resolver and the software is invading [some TA], your trust model of the operator is that it's trusting the software developer and the automated process is being made through software upgrades regularly. The only thing is that [inaudible] this software, before it starts, is actually making a check. This way is a simple way to achieve all these steps, I believe.

If the operator is willing to have a more complex configuration – for example, he trusts this ccTLD; he doesn't trust the root or he trusted in some other ways – then he probably needs to put in place some more things. But, if he has a simple configuration, I believe, for most of the software, it just has to have the up-to-date version running.

Update. While your resolver is running, because the trust anchors are changing over time, you need to be able to roll over the values. The operator should be able to regularly compare those values to the one he's trusting. So it's to check whether the latest value is being considered for its trust anchors so he knows which key he's going to rely on and it's something he can check and follow.

As I mentioned before, the only concern here is about the trust anchor into the cache. If you notice somehow that one of your resolvers doesn't have [inaudible] – I mean, the rollover has not been proceeded correctly

– it is considered as a bug. It would be a software bug, so the idea is that he doesn't try to fix that. You shut down, reboot, and restart. You have the up-to-date configuration and you're done. But we don't want to change the configuration online and so on and so on. In some sense, the responsibility of the operator is quite small.

Automated reporting. The idea is that the operator should be able to check that he has the up-to-date anchors. Also, DNSSEC is part of a connected world, so it would be good that the resolver is also implementing – I'm going to say that next time, but now that I've started – when he's also reporting to other authoritative servers, which key he's using or what's the value of that key. There is an RFC to do that. So it's just enabling those mechanisms. So those authoritative servers, which are responsible for the validation – it's not a validation but it'd probably be more [insensitive] of having that domain name being reachable. They can have a relationship with those resolvers.

Negative trust anchors. You should eventually monitor signature failures but maybe not automate the process between the failures, the number of failures, through a negative trust anchor. So it's a hard decision to take: whether you're going to have a negative trust anchor or not. But that shouldn't be done, I guess, with human intervention. Checks validation. That's what I [mean].

So, as an operator, when you really need to put in place a negative trust anchor, you should not try to find out how to insert it. That should be a well-known procedure. How to handle this negative trust anchor

should not be discovered [live]. So, again, it's heavily documented. It's just following those recommendations.

You should be able to add those negative trust anchors, not only for running resolvers but also for starting time, which means you need somehow correlations between the configuration file generations and … When you insert a negative trust anchor, it is also to be inserted into the process that's going to generate the configuration files, which will not be distributed through the software upgrades.

Intervention with the cache. You should be able to flush the cache or maybe flush all the validation associated with trust anchors. The thing is that you don't really keep which keys have been used in all the caches. Even if the only way to flush one corrupted domain is to flush the full cache, it's still something that is viable, I guess. Don't try to make things too complex.

Then we have all the keys that are not trust anchors but are still all the same keys. What can we do with that? Essentially, what we want to try to avoid as much as possible is to have a key that is not really validating the new … So an old key with a long TTL that is not able to validate the new request of that zone. So these kind of things. That's what we try to avoid. So maybe the only thing that could be done is checking that the TTL is not an [inaudible] value and that the coming requests do not go too way long beyond the TTL of the key.

Reporting the keys. If you see a problem, you should be able to check which key is being used for these kinds of domains. Again, we use a DNS

interface and we recommend to. Maybe having a look at validation failure would be interesting.

The last recommendation. We're recommending observing which signatures [can be] used for the validation. This is mostly to be able to deprecate old cryptos from software so that software don't have to carry the history of cryptography and can run the latest version.

The next step. All these have no been inserted into the draft. I'm going to include all your feedback. Feel free to comment directly on the mailing list or reach out to me for anything.

DAN YORK: I want to just emphasize this point. This draft that Daniel, myself, and Ed Lewis have been working is designed to help people who are doing DNS resolver operations, as Daniel mentioned here, to help them get a sense of what do they need to do to do DNSSEC validation checking, what are the steps, because we often say, "You just [un-comment] one line in your config file and, boom, you can start validating." Sometimes it's that easy, but we also want to provide a checklist of people that are there. So we would really like the feedback of the smart people in this room and anyone listening to be able to tell us, are these the right sets of recommendations? If we were to give this document to somebody who's out there providing a DNSSEC-validating resolver, would they be able to use it? Are these the right things? Are we missing anything? Please do let us know. That's the link up there. Our addresses or all at the back of the draft and we would love to hear from you.

DANIEL MIGAULT: Also, the clear idea is to also mention that, yeah, there is not much to be done. So that's the message the draft should carry. Someone just told me a few minutes ago that he observed that an ISP was during DNSSEC validations, so he asked them, "Oh, you're doing DNSSEC validation?" and they said, "No? I don't know," because it was a default configuration. So they didn't even know they were running the DNSSEC validation.

DAN YORK: Thank you, Daniel, for bringing this here for people to take a look at. I would encourage everyone to go and take a look at that draft.

Russ, are you going to be next?

Okay. So next up, you'll notice that we're not talking about DNSSEC, which would seem strange at the DNSSEC Workshop. But, over the last year, the Programme Committee looked at things and said, "There's a range of other connecting technologies that also connect into where DNS and DNSSEC are." People at the last session that we did at the previous ICANN meeting will remember we had some talks around DoH and some other things like that. So we're looking to expand this to be a bit more about DNS security, privacy, and related kinds of technologies. So I would just put the plug out here to that, if you're looking at these presentations and thinking, "I might have a topic I'd like to present to this group, consider it because we'll putting out the call for proposals for the next session soon after this one is done."

With that, I'm going to have Russ talk to us a little bit about what is RPKI. Also, for speakers, we do have a portable mic and a clicker if you want to stand up and move around. You do not necessarily have to sit at a chair if you do not wish to. Over to you, Russ.

RUSS MUNDY: Thank you, Dan. We had, as Dan said, several requests for expanding the things that we talk about in the workshop. This is what was felt as an area that people wanted to know and hear a little bit more about. We actually have a couple of routing security things that are on the program today. As the first one is really here as the introduction to RPKI, even prior … okay. Where are we pointing the clicker to? What makes it …

UNIDENTIFIED MALE: I don't know. [inaudible]

RUSS MUNDY: Oh, there we go. So what I want to first talk about here is, what is routing security? People say it, hear it, talk about it, but where is it written down? Where it is described? Well, it largely comes from work that's done by the IETF. The first sort of substantial document that talked about the security aspects of BGP was RFC 4272, published ten years ago or so. I'm don't remember for sure exactly when. That was the first IETF product. It's generally recognized as the starting point for getting things written down for what needs to be fixed.

Now the way that the IETF has approached the BGP-related fixes is in incorporating some crypto technology to allow validation of updates that routers get relative to their running operation. This is what we're going to really talk about today: the RPKI piece, which is Resource Public Key Infrastructure. That's what RPKI actually stands for. The [Cider] Working Group, which has now completed its work, and the current active working group, [Cider] Operations, are the active groups publishing RPKI-related and BGPSEC-related documents.

So that's really, in the view of the IETF, what constitutes routing security. Now, other people may have other definitions of it, but that's the approach that the IETF is taking.

I won't try to read this list. I would put it in this slide deck so it would be in this slide deck so people could look at it later and squint and say, "Oh, my goodness. There's been a lot of things." In fact, if one follows the NANOG list, seldom does a day go by where there isn't some kind of routing problem identified. So the list is long. Many times it be accidental. Sometimes it may be on purpose. But one of the problems if you can't tell the difference between what's an accident and what's intended.

Today, the way routing security gets done is basically is, "Trust me. I'm your ISP. I will do the right thing for you until you get into the deployment of the RPKI." The RPKI deployment is moving forward and being put out there. We'll talk a little bit more later on about that. Some of the other presentations will cover it, too.

So there's really a set of steps that you need to go through. The first step is to have some way to bind or connect in a cryptographically provable way that some entity is the appropriate entity to be using a resource. A resource could be an IP address or a block of IP addresses or an autonomous system number.

The next step is for people out on the network to be able to determine that those resources are being used by the appropriate people and, when you get an update that says, "I want you to put this kind of information in your router for how to get packets to me," the people that run the routers can verify that, in fact, it's coming from an appropriate place.

The third step is to verification of the actual path that's taken by the routing update itself. Now, that's a further-out step. There's experimental activity. There are some early implementations of that. But that's really a future piece I won't spend much time talking about. It comes and makes use of the RPKIs. Everyone who has ever heard the term "BGPSEC," that's what Step 3 is: BGPSEC.

So what is Internet routing? How does it actually work? I won't go through the detailed steps, but you can see that the little blue arrows are pointing to things out there that are little router things that sit there, take packets in, push packets out, and move things around between one interface and another. How do they know which interface to send things to? Well, that's what routing is all about. The path that goes from the client on the left to the server on the right is what the routers are

doing: moving packets through the Internet. You'll see that general picture is repeated here through the slides.

What are autonomous systems? They also get involved in the routing world. Autonomous systems are really collections of routers that are operated by a single administrative authority that are operated – at least supposed to be operated – in a consistent and cohesive manner. Occasionally they're not and that tends to break things, too, but they are supposed to work jointly together.

Up there, the little blue circles wit the AS arrows pointing to them – there are three autonomous systems up there. It's Autonomous System 5, 6, and 7. And the other individual ones – although they're routers they function essentially as a standalone autonomous system.

What is it that a Border Gateway Protocol does? Well, it ties together all these autonomous systems. As the packets get moved from router to router, the way that first determination is made by routers for what direction they ought to send it is that it ties it to the autonomous system. Then, when those autonomous systems talk to each other, that's really what the BGP routing protocol is all about. The BGP routing protocol is a protocol that defines where the autonomous systems do their connections. It does not itself move packets. The routers move the packets around but the routing protocols are what carries the information around to the various routers that let the routers know how they need to move the packets.

Right now, there's very much of this, "How do I know it's okay?" "Well, trust me. I'm a good guy. I'm your ISP (or I'm your friendly next adjacent

autonomous system. I'll do the right thing for you." Well, how do you know? The fact of the matter is, just like DNS without DNSSEC, the ones that receive the answers have no way to know. So, for people that are used to thinking about DNS and DNSSEC and how it works, it's at a very high level of abstraction. It's the same sort of thing for the routing system.

What the signed records in DNS do are essentially the rough equivalent of what signing your resources do.

What we've got in this picture is an additional unknown entity down at the bottom. That unknown entity is connected to Autonomous System 5. Now, the client in the upper left-hand corner wants to actually get packets from itself to its server that happens to be connected to Autonomous System 4. When things are all working properly, there are several different paths that can be taken by the packets as they pass through the routers. The shortest path is, most of the time but not always, the preferred way that the routers will send the packets through the network.

As you can see there, if Autonomous System 1 gets the packet, it says, "I can only send it to 2, so I'm going to send it 2," but then 2 gets it and it can send the packet to Autonomous System 3 for delivery to Autonomous System 4 and subsequently the server, or it can send it to go to 6 and then 7 and then back to 3 and then 4, usually the path will be 1 to 2 to 3 to 4. So that's a very brief, simple explanation of how the routing works with BGP.

But what happens if somebody jumps up and starts lying about something and the routers don't have any way to figure out if they're lying or telling the truth? This bad guy sitting down here in the lower left-hand corner says, "Hey, hey, hey, I've got a path to Autonomous System 4," but there's nowhere out in the routing system that they can determine that. So what happens when Autonomous System 5 claims a path of length that's shorter than anything else to Autonomous System 2? Guess what? Autonomous System 2 is going to send the packets to 5.

So where does the RPKI fit in all of this? That's where we can see that the first thing is to get the routing resources signed cryptographically. That's Step 1. Secondly, have the routers that are getting the information for the updates do validation of the update information that they receive. Again, putting it in a DNSSEC context, this is a singing of a zone and getting resolvers to validate. It's roughly the equivalent of that. The function is different in the real world, and the implementation is different in the real world. Conceptually, though, it's the same sort of thing.

So what you have now is, when you insert the RPKI and do the signing and do the validation, the Autonomous System 2 router will see that the resource that Autonomous System 5 is claiming to be able to route to is not appropriately certified to go to that AS and therefore Autonomous System 2 can make the appropriate choice of sending it either to AS 3 or AS 6 and go on around. So effectively you've got an invalid route being advertised that, because the RPKI is in place and being used, was not able to disrupt the routing.

These are again a summary of the three steps involved. The resource certification – that's, again, roughly analogous to signing your zone. You sign your resource authorizations and get those into the RPKI. The origin validation is done by the machinery that's out associated with the autonomous systems that will validate the routing updates prior to having those updates being actually used by the routers themselves.

Eventually where we want to get to is to have the path that the routing updates use be able to be validated as it passes through the network because there are some organizations that want to have only certain paths taken. That is what the path validation is intended to provide. Like I said earlier, that's a further-out set of steps.

So where do the certificates get generated from? What are they associated with? Well, it all starts with the IANA. The numbers are allocated to the Regional Internet Registries. The regional Internet registries in turn allocate them out to enterprises of various types, whether it's directly to an enterprise or an ISP.

So, logically, where do the certificates associated with the RPKI go? Well, right now they start … The picture was hard to draw. The certificate chain actually starts with the RPKIs rather than the IANA's. So there are really five routes in the RPKI certificate structure, but mirrors the actual allocation of the resources that, again, starts with IANA and goes to the RIRs. The RIRs are the entities that operationally are issuing resource numbers. So they're the ones that are running the RPKI certificate at the top of the certificate hierarchy.

So the first step that an entity has is to work with the people that they get their number resource from. There are several different ways in which you can do the resource certification, and some of the other presentations get into more detail about that later. Once you've got them signed, then you, if you have any responsibility in the routing world, should undertake to validate the updates. This is happening on a broader scale as we go forward.

So I think about it as, up at the top, there's this set of RIRs and the giant cloud in this case is representing all of the probably millions of routers out there that are actually doing routing in the Internet.

The thing that is the important part about this slide is that, for the RPKI itself, there is no crypto required to be done in the routers. So the design has been such that the validation of the routing updates is intended to be done in a way that routers are not burdened with cryptographic processing.

So where are the RIRs? In case you're not familiar with them, there's the URLs for their main website. The Numbers Resource Organization, which is really the collection of all the RIRs, has a statistic page. So you can go to one spot and get a look at what the statistics are for the RPKI deployment.

So that's a very quick run through. We maybe have time for maybe a question, but that's about it.

DAN YORK: Yeah. Anybody have any questions for Russ? Did this help anybody understand what RPKI is?

Down there.

UNIDENTIFIED MALE: Thank you very much for your presentation. Can you please go back to the previous slide and re-explain it to me so that I really understand how the validation is done by the routers? Thank you.

RUSS MUNDY: Okay. As the BGP updates are sent amongst the various autonomous systems, this general design of the RPKI is that these updates can be validated external to the actual routers that are moving the packets around. So most of the autonomous systems structure themselves so they operate cohesively, where they build their routing tables before they load them into the routers. So that's the place where you do the validation that the routes are appropriate.

Is that … okay. Good.

DAN YORK: All right. Russ, I liked what you said there. Our PKIs are very similar to DNSSEC in the sense that you do have the validation side and you have the signing side. Very similar. Just different mechanisms for how it all works, although, as you said, there are five roots, which would require five rollovers, etc.

So that is a perfect segue to bring up our next guest. But let's give a round of applause for Russ, please. Russ also sets up very nicely because after the coffee break, etc., Jaap will be up to talk about new adventures in RPKI. I'll also be talking about the MANRs Observatory. But here to talk about what could be the next roll of the 1 route in DNS is Kim.

KIM DAVIES: Hello, everyone. My name is Kim Davies. I work on the IANA team. I'm here today to present to you our thinking on how we wish to do future root zone KSK rollovers.

Just for background on DNSSEC and the root zone, since 2010 is when we generated the first KSK. After some years, there was originally a target to roll the key after five years. Events happened, but at some point around that time a design team was formed compromised of the root management partners involved in actually doing operations plus community volunteers to provide their expertise. That design team came up with a set of recommendations on how to perform the first rollover of the KSK.

We use those recommendations to create a plan, and that plan originally scheduled the KSK rollover to happen in 2017. Ultimately that was paused for a year due to some anomalous telemetry data that we wanted to better understand before progressing. But nonetheless, with a one year delay, the KSK roll happened and we started using the second KSK that we refer to as KSK2017 actively on October 11th, 2018.

The rollover generally – I think everyone would agree – was successful. There was minimal disruption as a result of the process that we used. Just recently internally we cleared the project complete with a final destruction of the original KSK2010. That happened in August of this year. So it's now down. That project is now complete, and it begs the question of what do we want to do now? How do we want to address rollovers moving forward?

We made a decision around the time the rollover happened last year in October 2019 to try and capture the community interest at that time. We recognized that there was a lot of intense focus on the rollover, not just the inner sanctum of the community that's probably highly represented in this room but is the broader operational community of ISPs and network operators that were paying particular attention due to the outreach efforts that we'd performing leading up to the event.

Around that time, we made an explicit effort to channel feedback on how things were going to the KSK rollover mailing list that we have. The idea there was simply just to capture the feedback at the time and shortly thereafter, not with a notion of immediately acting upon that feedback, provide a mechanism for capture and then come back and analyze that later once our project was complete.

As I mentioned, we finished our operation in August of this year. Then we switched gears to going back to the feedback from the first half of this year, distilling it, analyzing it, considering it, and turning it into a proposal.

Some of the common themes in that early commentary we received where that the KSK rollover should be a routine event. There was a lot of comments suggesting that the KSK rollover should be annual. There was some comments suggesting we should consider a backup or a standby key configuration. There was a suggestion that we need to perform more monitoring of larger key sets and their impact and that we should consider alternative key signing algorithms. We're using RSA today.

So we've developed a proposal and the proposal does seem to create a predictable approach to future rollovers. We hope this will be a model that will be repeated and predictable moving forward. The plan we put forth for public comment a couple days ago proposed a three-year rollover interval. We've considered the feedback and considered the operational impacts, and we believe three years is a good interval to balance the desire for more regular repeated rollovers, balancing that against the operational complexity. The operational complexity for the root zone management partners is quite intense and we need to consider that as part of the dynamic there as well. Our proposal should have a new trust anchor published at least two years in advance. This gives a long time for propagation and adoption of the new trust anchor.

But other than those key elements, I think it would be fair to characterize what we've suggested as a very similar approach to what you saw last time around. We use a similar phasing model to the one that we used last time and we adhere to the quarterly key ceremony schedules we've used in the past.

This is a graphical illustration. I appreciate it might be a little hard to decipher, so I'll walk through it a little bit. The key point here is Phase A. That's the blue phase. That is the phase that the KSK is active in, but the prelude to using it in an active state is, of course, generating the key (what we call Phase B replicating the key). We have two different facilities the key needs to be maintained in, so we replicate it to the other facility. Then there's the first singing operations on the key that happen in Phase C. Phase D is that pre-publication process, where, in that first quarter there, the RC 5011 adoption happens. Then it's in a standby state for an extended period of time. We have the rollover to an active state in Phase E. Then Phase F is the revocation upon the successful activation of the next subsequent key. G and H are the phases where we delete it from our facilities. We perform destructive operations to ensure no trace of it exists anymore.

In summary, it takes at least three calendar quarters to generate and successfully replicate the new KSK. So you've already bitten off seven or eight months in that process. We have at least, according to this timeline, seven quarters that it is in a standby state where it is prepopulated and capable for an unscheduled role. We have twelve quarters that would be the target for then active state – that's when it's actively signing the zone – and then the following three quarters to revoke. This is where we delete the key and publish it with revoke-it set and so forth.

The prelude and the post-rollover activities are important. When you look at the next graph, you see the cascade of all the different KSK operations and how they interact with another. You can see here that,

any one time, we're often interacting in this model with up to three KSKs operationally.

This – I'll get to it in a little more detail – is why we felt that an annual rollover would be impractical because, if you're just looking at Phase E, that's one year long, but if you consider A, B, C, D, F, G, and H, you have a lot of KSKs running in parallel if we have such an aggressive schedule as annually.

A little more on the choice of interval. A common suggestion was to perform it annually because we need those quarters to generate, prepopulate, pre-publish, revoke, and destroy. We could see with an annual cycle we would have four or more KSKs in place simultaneously. We didn't want to have such a situation.

Some of the reasons why its complex for our operations is because key ceremonies, for those that are not familiar, are quite detailed and time-intensive. Every new step or new act we place in a key ceremony makes them longer. It's hard enough having everyone's full attention during a ceremony when they last four or five hours, which is typical today. But as we add more and more signing operations, more and more need to interact with the different elements involved in this – destroying keys, generating keys, key-signing request, and so forth – I can totally foresee it being six, seven, or eight hours in some ceremonies. I'll speak for everyone on the ceremony: no one wants that. But for sure operationally we don't want that because we think it's critical that those that participate in ceremonies do have their undivided attention the whole time successfully execute their roll, and we don't want to put

people in a position that they have to monitor all these keys in flight in a serious way over such a long period of time.

Do bear in mind that, when we're doing key-signing operations during a rollover event, Verisign and its role as the ZSK operator sends us multiple KSRs for different fallback strategies so that, should something anomalous happen during the quarter, they have the capability of rolling back to an earlier phase. So you can multiply out the different levels of complexity, should we have a lot of KSKs in flight.

The other thing to keep in mind is that, in addition to all the key life cycle management that we do during key ceremonies, we also do life cycle management for all the other elements in the ceremonies, where we're destroying and inducting your HSMs every few years. We're adding and removing TCRs. We're replacing safe locks and all sorts of other things. That all adds time to the ceremony as well.

The next element that is different here is we do propose creating the KSK earlier in that life cycle to allow earlier adoption. This ideally will provide at least two years for software vendors and other distributors of the trust anchor to put it in their distributions and get widespread adoption. That's different from the last plan.

Another benefit that it provides is, should we have a need for an emergency unscheduled rollover, if we're confident that it's being pre-populated out there, where in a much better position to roll over rapidly to that new KSK. So having it in [advance] in a standby state allows us to do an emergency key roll much quicker. Any negative impacts on

allowing factoring in attacks on the key and so forth by publishing it earlier have been considered negligible.

One of the asks or one of the suggestions was a backup or a standby key. We've not proposed doing that at this time, other than what I just mentioned: by having a pseudo-standby capacity by issuing the KSK earlier. The primary reason for this is that we don't have a good alternate facility to store it in. So a lot of the scenarios that this might guard against would have fate sharing with the actual KSK. So we saw a limited benefit in doing. That would suggest it's not overly beneficial compared to the operational complexity it would introduce. But we think this does deserve more study moving forward, so we'll take it under consideration in terms of other kinds of storage, additional KMFs – this kind of thing that might change that dynamic a little bit. But as of right now, operationally we only have two KMFs. They're designed to be exact mirrors of one another. If we were to create a backup or standby key, it would sit in those exact same KMFs and would have that fate sharing element.

Algorithm change. We agree this needs to be investigated. However, our assessment is that other algorithms are not mature at this time and therefore it would inappropriate for us an the IANA to start operationalizing an algorithm change. Rather we see this is more ripe as a research activity now to better understand how an algorithm change would be done in the root and what the impacts would be in a research test environment, not in production. So we propose that thought into this continues but that that be done as a separate activity,

perhaps in ICANN's OCTO department or something else, which would be comparable to how the original rollover was first started.

This approach that I've just described we've put in a fairly short paper that outlines the approach. It is now open for public comment. It's available on the ICANN website. Available now. We've, based on some feedback yesterday at the meeting, extended the deadline for feedback to the end of January. So you have the next three months to chew on that a little bit.

Our goal as staff is to look at that feedback once the public comment period has concluded and then try to turn it into a final operational practice. My expectation is, barring significant change to the approach, that we will be able to put out some tentative timelines in the new year based on that feedback.

In summary, we think the rollover from 2010 to 2017 KSK was considered successful. We're seeking to take all the good parts of that and replicate it whilst additionally adding some predictability and a regular schedule to the KSK rolls. We think annual rollovers are too complex for the reasons I outlined, but we will seek to create the KSK early to allow greater adoption of the new KSK.

Please do provide feedback to us. If you don't like it, that's fine. But we're always willing to hear constructive feedback. What would you suggest as the alternatives if you don't think this is the right approach? Please think about the operational impact of the proposal if it is quite different from the way we do it today. We'll try to finalize the approach

once the public comment period ends and communicate to you our operational plan.

So that is it – no? Oh, [burner] slide. Completely out of topic, but this seems to be exactly the right community to bring it to your attention. We're almost at the ten-year anniversary of KSK operations. A lot of the community volunteers – the so-called trusted community representatives – having been doing this for all those ten years. We know privately a lot of them are interested in new challenges, so we've been mindful of this and we developed a plan to start replacing the TSRs a few back.

The way we've done this is created an evergreen process where, at any time, folks that are interested in this role can submit a statement of interest. We keep them on file and, whenever a new vacancy becomes available, we go through the on-file SOIs and rank them according to set of criteria: geographic balance, skills balance, reputation in the community, and so forth.

We've just made our first selections with this new process. It brings up our backup pool back to ten people. This gives us a ready of supply of people that can replace a retiring TCR at any time. But given what we know, we expect to make some new appointments to the active TCRs in the next few months and certainly throughout 2020. So that backup pool will shrink.

I encourage anyone that has an interest in this to submit an SOI. We're always willing to have more candidates to put in the pool. So, if you're interested, please apply.

With that, I'm happy to take any questions on any of the topics I've discussed.

DAN YORK: Thanks, Kim. I think you also get credit for having, for the first time, I think I recall, a presentation here that used "destroy" and "destruction" so many times in the wording here. It was entertaining. I can't understand why people wouldn't want to be stuck for eight hours in a small room inside of a data center.

I do have one question for me. The process is that this operational feedback will go until next January or February or whenever it is. When do you anticipate you might start his process if it goes ahead to be approved? Because I realize it's a two-year pre-publication type of thing.

KIM DAVIES: My assumption – this is purely me personally; it would be a team decision – is I don't think this time around we're going to hit the three-year anniversary of the last rollover because we're already well into this period. But I think we're not looking to unduly delay the process. So if there' s general consensus for this approach, I think we'd be timing that first quarter within the next few quarters afterwards. So it could well be that the next KSK could be generated sometime in 2020.

DAN YORK: Great. Dmitry?

DMITRY BURKOV: Kim, I have a few points for new. They're not new but I want to repeat them in public. Before you begin something new, can you check if it [works with] all recovery? Because we never had a meeting of recovery holders. We don't know what the real situation is with their smart cards. Before we start, [inaudible] estimate any life cycles, any change. Maybe it will require [inaudible] the whole procedure. It's just [inaudible] life cycle.

[inaudible] a KSK generation – I'm a crypto officer – I know it's not so time-consuming. In practice, we achieve enough short times during. Maybe an HSM replacement might be the longest, but now you propose – well, I don't know why. What's the idea? Each four years to replace each time? For me it looks very crazy. We still know nothing about smart card life cycles. We have ten years for previous smart cards, and only one not in fault. But in usual life, nobody cares about such things.

I want to remind you about HSMs. The reason to replace this HSM is the battery life. I don't like this jump from ten years to four years or to three years. It should be more reasonable. Of course, for example, the life cycle as one year is too short. It's just not necessary. I can tell you the maximum, extreme position is that we can generate a KSK each ceremony, no matter where. We have no need for facilities. It depends on our decision. It should be reasonable. It's just more advice and more questions regarding recovery. I think, first of all, before beginning any transition, any replacement, you should do it. It's necessary. Thank you.

KIM DAVIES: Thanks. Just briefly, because you touched on a lot of topics there, the HSM life cycle is five years. That's what we're seeking to achieve now, as you note. The very particular situation we're in with HSM management is that, unlike most other HSMs, ours are never plugged into the power, except for about an hour every six months. The problem with this is that there's not really the standard operational model for HSM. The battery is there more as a backup, but relying on it for its primary function. We're finding that the batteries have a certain life cycle. So we are being conservative by doing five years. That means, because there's four across the two facilities, we're replacing one HSM every five ceremonies or every 1.25 years. So it's a factor. It's not that we replace HSMs every ceremony and it's highly impactful. But that's what we're trying to achieve: that we have a five-year life cycle.

Smart cards. That's something we're researching based on recent events. It's parallel activity to this. All I really wanted to share with the life cycle comment was just that we do have these other operational events that take time. And it does distract from the core operation of the KSK ceremony, which is performing the signing operations. But we do, in terms of planning our approach, have to be mindful that these operational activities need to do be done, and we believe they need to be done under audit – so, having witnesses present, having the third-party auditors present, having the TCRs present. People's attention span is finite, so we want to make the best use of that time.

DMITRY BURKOV: Really, [for me the surprise that it's the last] idea to generate all types of cards and have sets of all types of cards which can create some organizational problem from the beginning. Please be careful with such radical ideas.

KIM DAVIES: Well, on that front, we're exploring the idea with the TCRs. We've no to decided to do any particular approach yet. We're precisely getting feedback. Thank you.

RUSS MUNDY: Do we have other questions? We're a little bit over – Jacques? – but we can do some here.

JACQUES LATOUR: Can you go back to your schedule? Because I was going to … no, the … yeah, there. In there you finished the last one – destruction – and then, right after in parallel, you have the creation of the new one. Have you thought about, after you've done H, that the next quarter you do A so you don't overlap the two processes? It would make things shorter and more sequential.

KIM DAVIES: Yeah, that is possible. We're trying to strike a reasonable balance where there wasn't too much overlap. We thought this was manageable, but it can be extended even longer. Bear in mind that the predominant view we saw in the public comments was an annual rollover. I know that's

not a hard-fought position for most people, but that was a starting point for discussion. So we definitely got the impression we should be going more often than less often. This here seemed to be that we wind up one key and we start generating another so we don't have too many in flight. But we could expand it even a couple more quarters so there is no overlap.

JACUQES LATOUR: Yeah. To avoid the overlap and make this session smaller. Your observation is that you can keep the HSM in the safe plugged in?

KIM DAVIES: I don't think that's practical, but we'll look at that.

DAN YORK: We do have time for more. We've made a decision. We have a mandated coffee break from 15:00 to 15:15. Given that Yoshiro cannot talk that fast and that it's probably more interesting for him to talk slower on that, we're going to continue this for more questions. So, if there are some, please feel free.

JACQUES LATOUR: I'll keep going. Do you really need to have a year exactly or three years of operation? Can it be three years and two quarters or one quarter? And then making your timing work for the process? Because the duration is not important. What's important is that it's regular. Two year plus or minus whatever. It doesn't really matter. But if it makes it

easier for IANA and other volunteers to work in the process easily and no get bored by super-long sessions, then I think that's a factor that we should consider in this.

KIM DAVIES: It's not operationally necessary that we align to an exact number of years. However, our thoughts on this were that, from a public reporting and awareness perspective, given that we're trying to align to our quarters, because that's an operational method we have established and it seems to work quite well, it's better if it's the same day every year, like October 11th, for example. If we said it was October 11th, it's easier to communicate that message that, this year, October 11th I keep in mind. If it jumps around – it's April 11th one year, July 11th another – that makes communication a little more challenging.

I will make a caveat here. I think it's important that one thing we do try to convey is that these are target dates. If there is an operational anomaly that warrants us pausing, we won't hesitate. The criticality of the root zone is obviously key. Should we need to defer an action for a quarter or two or three or four, we will. So, whilst we target these dates, in practice there could be a decision to defer, as happened with the last KSK roll.

But, setting that aside, having a schedule that's predictable – that every third year on October 11th it happens – probably is an easier message to communicate.

DAN YORK: Plus DNS administrators wouldn't know when to schedule their vacations, right?

Any other questions, comments, here for Kim? Anybody going to apply for the TCR positions? Seriously. Kim's message is a good one. We do need people to apply for those positions.

Go ahead.

UNIDENTIFIED MALE: No, I'm not applying for [inaudible] exchange. I'm not applying but I was curious, since I gathered that the ceremonies are always happening in the U.S. because I think they have to be [inaudible]. But maybe having them more distributed would make it easier for non-Americans to apply.

The other comment I had is, thinking how hard it is to push any kind of software update or configuration update in the real world in the [inaudible] platforms that serve several millions queries per second in big ISPs, I think that the [inaudible] is already near to the minimum to [inaudible] limit. Maybe two years could be done, but not less than that. So I think I would like to say it's reasonable that you capped it in that way or not went down to one year.

KIM DAVIES: Thanks. On the location question, there was a historical necessity that the key management facilities be in the U.S. under the previous IANA contract. That's no longer the case. Obviously we could reconfigure the

locations with some significant expense. We've certainly heard that feedback from others. Yes, that's possible, but with the current configuration, obviously not.

DAN YORK: I have Warren and [Sergei].

WARREN KUMARI: I was going to suggest maybe having a key management facility somewhere else, but Kim has already answered that.

DAN YORK: Okay. [Sergei]?

[SERGEI]: A short comment about TCR selection procedure. I did apply for it, but as for me, there was small feedback from the PTI side because it's completely unclear whether my statement is still in the line or not.

KIM DAVIES: Thanks. If you applied since 2017, it would be. That's when we started this new process. Our goal is to send an annual reminder, saying – I think we did this last year – that – correct me if I'm wrong; we can talk in the corridor … So we did an original call for volunteers in 2009, I guess it was, and we picked the original class of TCRs. We did no solicitations whatsoever right up until 2017. 2017 is when we started

this evergreen process. We have a form on the website and we accept people volunteering any time.

Moving forward, our goal is to send an annual [tickler] to volunteers to say, "Hey, we still have your SOI on file. If your circumstances have changed, please let us know. Maybe your resume needs updating. Maybe your circumstances have changed and you're no longer interested. We can take you out of the pool." So that's the way we plan on operating moving it forward. If you've applied and haven't heard anything back, maybe there's an issue. But I can individually check out our ticketing system and check if your application is still in there.

[SERGEI]: Thank you.

RUSS MUNDY: Well, I'd like to extend my personal thanks to Kim for doing this presentation and bringing it to the workshop because this really was a great opportunity to roll this out. I know you talked about it earlier in some of the sessions. That's why we wanted to give you as much time as we needed here to cover this. I also wanted to acknowledge that ICANN and PTI and Kim delivered on time, on schedule. So thank you.

DAN YORK: With that, we need to go to a break. Thank you, Kim, for being here. Please, everybody, download this presentation and look at it and go to that document and provide comments. Kim's team needs these

comments to help him understand. Even if you say, "Yes, I agree. It's all good," I know that that's valuable input into Kim and him team. That could be as simple as it is.

So it is breaktime. Let's go. Let's try to be back here right at 3:15. Jaap will come up and talk about new adventures in RPKI.

I'm sorry. Actually, no. We've changed things. Jaap will not talk first. Yoshiro will talk first. He graciously is moving right after that and he has to leave shortly after he talks. So he'll be here and then Jaap and then me.

Oh, yes. And if you want to come to the implementers' [gathering] …

**[END OF TRANSCRIPTION]**