ANDREA GLANDON: Good morning, good afternoon, and good evening to everyone. Welcome to the fifth webinar of the 2018 At-Large Capacity Building Program on the topic of KSK Rollover Part 1 on Wednesday the 13th of June at 21:00 UTC.

Our presenters today are David Conrad and Andrei Kolesnikov.

We will not be doing a roll call, since this is a webinar. We have French and Spanish interpretation, so please, I remind you to state your names before speaking to allow our interpreters to identify you on the other language channels and for transcription purposes. Please also speak at a reasonable speed to allow for accurate interpretation. Could I kindly remind all participants on the phone bridge as well as on the Adobe Connect to please mute your speakers and microphones when not speaking. We will also mute all lines on the phone bridge during the presentation. Thank you so much for joining. I will now turn it over to Tijani Ben Jemaa, the chair of the At-Large Capacity Building Working Group. Over to you, Tijani.

TIJANI BEN JEMAA: Thank you very much, Andrea. Good morning, good afternoon, and good evening, everyone. As you have noted, it is part one of KSK rollover. Because we have planned, let's say two, webinars about KSK rollover, one before the rollover and the other will be done after the rollover happens.

So, today we invited Mr. David Conrad who is the Chief Security Officer at ICANN. He is more or less the man of the rollover. We will have him also publish a [inaudible] because he is the most knowledgeable person on this issue.

Also, we will have as a presenter Andrei Kolesnikov, who is the liaison of the At-Large … Sorry. [inaudible] who is already a member now, a member of the Security and Stability—

ANDREA GLANDON: Tijani, are you still on the audio bridge?

TIJANI BEN JEMAA: Do you hear me? Hello?

ANDREA GLANDON: I can hear you now, Tijani.

TIJANI BEN JEMAA: Okay, wonderful. So, we will start the presentation, but Andrea, if you have housekeeping announcements, please go ahead.

ANDREA GLANDON: Yes, thank you so much. Just one moment. I will run through just a few housekeeping items before we start. For questions and answers during this webinar, you can submit these via the chat pod in the lower left-hand corner of your screen. You can also send them through the regular

chat pod in the middle of your screen. These will be directed to the presenters. Please do, however, note that we have a question and answer session after the presentation and the pop quiz questions.

Regarding the pop quiz questions, we will display these after the presentation, so for all of those in the AC room, please be ready to answer the questions via the polling [inaudible]. This will show up on the right side of your screen.

Finally, at the end of the webinar after the question and answer session, we will have a user experience survey composed of seven questions. Please stay around for an extra three minutes or so to complete them. It is important feedback for this At-Large Capacity Building Program. Thank you, and back to you, Tijani.

TIJANI BEN JEMAA: Thank you very much, Andrea. So now, the floor is for the presenters. Who will start? Will it be Andrei?

DAVID CONRAD: So, I think the approach we were going to take is I'll run through this presentation and then hand it off to Andrei, and then I guess open it up for Q&A, if that works with everyone.

TIJANI BEN JEMAA: Yes, please.

DAVID CONRAD: Okay, then I'll get started. I believe I have taken control. I'm David Conrad, the CTO of ICANN, speaking to you today about the root zone DNSSEC key signing key rollover.

To start, to I guess level set essentially, this talk is related to the domain name system, and in particular, the security extensions that were made to it.

When the DNS was originally defined, there was actually a structural bug that resulted in the ability for, at least theoretically, for bad guys to provide a response back to queries and have that response accepted and enabling the cache to be poisoned. That is, that bad data could be inserted into resolvers cache, which would allow for a variety of bad things to happen, like man-in-the-middle attacks and other similar forms of attack.

These forms of attack actually are generally not seen out there in the wild because there are far easier ways to attack the infrastructure, but they are theoretical attacks that have been demonstrated in practice, and personal opinion is that as the easier attacks become harder, as people harden their infrastructures and fix applications and use stronger passwords, that we'll begin to see more and more uptake of attacks that DNSSEC can prevent. In fact, one of these kinds of attacks was actually just implemented recently against myetherwallet.com and that particular attack could have been prevented by the use of DNSSEC.

So, DNSSEC are a set of security extensions that were defined to address this particular bug in the DNS protocol specification. The way they work is they apply digital signatures to DNS data, using the hierarchy that's

inherent in the DNS to achieve massive scale. There are, on the root – well, and other zones, but we're talking primarily about the root here – the roles have been broken up so that there's key signing key which signs a bundle of other keys and the zone singing key which is actually used to sign the zone data.

The reason this particular split was made was to allow for frequent change of the zone signing keys without requiring a change of the key signing key. The reason for that will be discussed a bit later.

DNSSEC actually has sort of two parts. There is the signing of the zone, which is done by zone administrators. So, TLD administrators in the new gTLD space are contractually obligated to DNSSEC [inaudible] their zones. Many ccTLDs also sign their zones. The root was signed initially in 2010. This process is you take the zone data, you compute a hash and you [inaudible] sign that hash and this provides a way of ensuring that the zone data, if it is modified after it has been signed, that you're able to detect that modification.

That detection of the modification is known as validation and it's done by the recursive resolvers. Sometimes, at least in theory, stub resolvers can also check the signatures. And a stub resolver is the software that's linked into applications and the result, if the validation succeeds, then the response is provided back to the application. If the validation fails, then an error is returned back. DNSSEC doesn't actually prevent modification of zone data by the bad guys, but it does allow for the detection of that modification.

The root zone DNS key signing key is the top-most in the hierarchy of this chain of information that allows information to be validated. So, the way validation works is DNSSEC data comes in, along with the signature. The validator checks that, then pops up a level, gets the signature associated with the parent, checks that, validates it, goes up to the next level all the way up to the root. And at the root, since there is no parent zone which you can fetch the relevant information to follow a chain, there is actually hardwired into every resolver that has enabled DNSSEC validation is what's called a trust anchor, and the trust anchor is actually the public portion of the key signing key itself.

So, DNSSEC uses asymmetric cryptography, so there is a public key and a private key. The public key is the part for the root. The public key is the part that's configured into resolvers.

What does that mean? Well, if we want to change the KSK, the root zone's KSK, that means we're generating a new public key, private key pair and we need to change the configuration of all the resolvers around the world to reflect the new public key portion.

Right now – well, historically – we have a key that was generated when we signed the root initially in July of 2010. That's known colloquially as the KSK 2010. We created a new KSK back in 2017 and it will be put into production at some point later in this year and we call that KSK 2017.

The impact of this is that the operators of recursive resolvers, which are typically Internet service providers or enterprise network operators, although pretty much anyone can run a resolver, they'll need to do either look at their – well, they'll have to look at their configuration to

see if DNSSEC is enabled, number one. And if it is, if the configuration information has the KSK 2017, also the KSK 2010, or if it only has KSK 2010. If it has the KSK 2010, that is only the KSK 2010, that is indicative of a problem.

KSK 2010 is the one that is in current use. There wasn't anything before that. The public portion of that is configured into resolvers and if you have enabled validation, then that is the public key that's used to validate the data.

If you have enabled DNSSEC, and most modern validating resolvers do this, there is an automated system to update that key and some months back – I forget the exact date, but we'll get to it – the new key, the KSK 2017, was automatically inserted into that configuration data. And 2017 is not a typo. It was generated in 2017 and we had intended, as I'm sure you're aware to roll the key in 2017, but as we'll discuss a bit later, we suspended the role to do some investigation and we will be moving forward with that, but it's 2018, so sorry about the potential confusion that the naming of the KSK might cause to folks.

So, the approach to the KSK rollover emerged as a result of planning that began in 2013. When we signed the root in 2010, within the DNSSEC practice statement, which is a policy document that goes along with DNSSEC signing the root, we had promised that we would change the KSK after five years. So, we had actually begun the process thinking about planning for the change in 2013, and then something came up that sort of distracted us, which was the IANA functions transition, so planning was sort of put on hold because we didn't want to have all the balls up in the air during the IANA transition and just have a few of the

balls up in the air. And when the IANA transition was sort of moving along without need for continued close watching by the technical staff, we reinitiated the planning and sort of developed the full plans in 2015.

Due to the way the standard for doing the automated rollover works, we had to take a slow and steady approach and we worked on the normal update cycle associated with the KSKs where we actually go into the key management facilities and use the KSK to sign the zone signing keys, which happens once a quarter. So, every action that we needed to perform for the KSK rollover occurred on a quarterly basis. This standard is defined in RFC 5011 and it is sort of the mechanism by which we assume all resolvers will update the key.

In reality, not everyone turns on the automated update of the KSK for various reasons, including concerns that somebody is remotely changing configuration of resolvers which makes some people nervous. And other reasons.

But, since the protocol was defined, we decided to use that as sort of the rules of the road for doing the signing itself.

The important milestones – and this was the original plan – was we created the KSK in October 27, 2016. We made it production qualified, which means basically that it was propagated – essentially, copied – from the original key management facility, which was on the east coast of the US, to the second key management facility, which is on the west coast of the US, and installed into the hardware security modules that we used to have the highest level of security.

So, in February of 2017, the new key, the KSK 2017 was actually ready to be used. After February 2, we actually began publication of that, which was done in a variety of ways, including printing it on T-shirts and announcing it, putting it on the IANA website and various other ways.

In July of 2017, the 5011 process was initiated and the automated updates started to occur, which meant that the new KSK began to be inserted into resolvers that were configured to accept the automated update.

The plan was that on October 11th of 2017 we would begin to use the new KSK by signing the zone signing keys which would then be used to sign the root zone, but as we'll discuss, we chose to defer the actual use of the zone because we weren't sure exactly what was going on. Subsequent to that, the plan was to revoke the old key, which is basically setting a bit in the key saying that it won't be used in the future, and then in some point after that, we hadn't figured out the exact date because it didn't really actually matter since the key wasn't being used, but then we would actually remove the keys in the automated process, which for folks who weren't using the automated process meant that they would have to go in and actually edit their key – or edit their resolver configuration.

But, as everyone knows, we suspended the KSK rollover. We have since sort of begun the process of restarting the rollover and our plan at this point is exactly one year after we had initially planned on using the new KSK to sign the root zone on October 11, 2018, we will use the new key to sign the zone signing keys.

The revoking of the 2010 will undoubtedly occur one quarter after that and the same issue with the removal of the key. There's no big rush to that. We can do it pretty much anytime. We just haven't decided what the exact dates are.

So, why did we update the milestones? When we started the rollover process, there was no way to actually measure resolver configurations. We were basically shooting in the dark. We had no full idea of how many people had turned on the RFC 5011 stuff to allow for the automated update and we were relying primarily on a communications plan to try to ensure that everybody was doing the updates appropriately.

However, during the KSK rollover project, Duane Wessels and Paul Hoffman came out with an RFC that defined a mechanism that would allow for a measure of how resolvers were actually configured to be published. That was surprising. The specification for this technology, RFC 8145, came out I believe in April of 2017 and then in August of 2017, Duane Wessels, one of the authors, decided to look. Duane works at Verisign. Verisign operates two root servers. And Duane decided to see if he could see any signal from this particular publication or measurement mechanism, and we began to see data and it was very confusing. Not just confusing, but it also worried us.

So, that is why we decided to pause the KSK rollover, just to figure out what was actually going on.

That RFC is, as I mentioned, 8145 and its title is Signaling Trust Anchor Knowledge and DNS Security Extension. Let's see. Oh, rats. So, this is a

really pretty graph that shows the percentage of resolvers that were only announcing the KSK 2010. To the chat, I will post a URL that allows you to see the real-time statistics on the publication of the KSK 2010 and KSK 2017. It's odd that Adobe isn't allowed to deal with PDFs, but whatever.

So, we were seeing back in September of 2017 – so, about a month before we were going to actually use the new key and production – we noticed an uncomfortably high number of resolvers were claiming they only had KSK 2010, not both KSK 2010 and KSK 2017.

At this point in time, we had assumed that the percentage would be below 1%. At that time, it was actually 7%, according to Verisign. We, at ICANN, started looking at the root server data that we had access to, which is not only the root server that we operate, but we had also made arrangements to obtain data from the root server operated by University of Southern California, the Internet Systems Consortium, and the University of Maryland.

So, we began collecting quite a bit of data and analyzing it, and our numbers were showing things that were somewhat worse than what Verisign were seeing. And more disturbingly, the numbers of – the percentage of resolvers that were showing only KSK 2010 was going upwards. As more people began deploying the 8145 code, the numbers of misconfigured resolvers seem to be increasing.

So, when we started looking at this, we had some question about what was the data actually telling us? Remember that this was code that was implementing a spec that hadn't existed until April of 2017. It was very

new code, and that meant that it would only work in resolvers that had implemented it and people who had basically bleeding-edge type code, people who are early adopters and had implemented the latest and greatest that came out of the resolver developers.

So, the indications we were getting weren't making a whole lot of sense. We began looking for a systemic cause within the DNS protocol specifications and we weren't able to … We found a couple of bugs in various popular resolvers, but it wasn't sufficient to sort of explain what was going on, so we started looking – taking the information that we had at the root server level and trying to track down the folks who were generating these announcements that we were seeing, showing only KSK 2010 was configured.

At the root servers, we see the IP address of the resolver that's querying the root server. Unfortunately, it turns out that the simplistic model that many people might have about how the DNS works where you have a client application calling the resolver, which then queries the roots, appears to be relatively rare in the wild and there are whole series of forwarders and other devices that forward the query on so that the result that the IP address that we get from the resolver doesn't really have any relationship to the originating querier, except through a chain of devices.

So, while we would see the last link in the chain, we would have very little information that would allow us to find the source, the beginning of the chain of resolvers. We were able to get a few folks investigating what was the cause. It turned out to be things like virtual machines that had been configured ages ago that people would start up to run some

test and it would have very old software, which had the KSK 2010 only, and that would interact with the resolver on the parent of the virtual machine that would be sending out the KSK 2010 announcement only. But there wasn't any clear picture about what was really going on as we continued to investigate this.

As mentioned, 2017, we paused due to uncertainty. There wasn't really a fault in the project plan or the execution of that plan. The plan itself sort of anticipated issues of this nature. There were always checkpoints and fallback positions that we would entertain, should something unexpected happen. So, the plan actually worked pretty much flawlessly.

So, because we were sort of struggling for an understanding of what was actually going on, we did what the organization frequently does in the case where we're not understanding what's going on, so we ask the community. We engaged DNS technical experts trying to identify a way of moving forward. We prepared an updated plan, submitted that for public comment, got quite a few comments. ALAC included. They provided comments. Eventually came up with a revised plan.

The short answer with that plan was that the 8145 data can't really be relied upon because it doesn't reflect anything particularly useful. It shows resolvers that are misconfigured, but it doesn't indicate how many users are behind those resolvers. And it's the users that we actually care about here. The way DNSSEC has been deployed and the reasons that at this point in time we're seeing about 25% of all DNS queries being validated is because of a relatively small number of very

large resolvers – for example, Google's 8.8.8.8 or the resolvers that Comcast, a large Internet Service Provider in the US – have deployed.

The implication of that was that 8145 since it was resolver announcements, could mean a resolver that was saying that it was misconfigured could be a test system that was running an application that no user would be impacted by, or similarly could be a resolver that thousands of people are relying upon. We had no way to tell.

One of the leading points in the discussion is that since we didn't know what the data was actually reflecting in terms of the impact to end users, it wasn't data that we could actually rely upon.

Here is a graph that is showing the [inaudible] reports from all the root servers. You'll see that sort of at the peak we're looking at around 180,000 unique IP addresses that were indicating that they were … Why is it doing that? Try that again. There we go. 180,000 announcements that we were seeing. The redline is the number of sources that we're reporting only the KSK 2010.

The next slide, which again is not showing the graph very well – I don't know how to fix that. It actually says the percentage. I should point out that these slides were developed for a CERT presentation to the Computer Emergency Response Teams. That's why this slide is talking about what it means for a CERT.

The reality is that if the KSK is misconfigured, if it's only the KSK 2010, and then we actually move forward and sign the root zone with the KSK 2017, it would mean that any resolution that occurs through that resolver would fail simply because the key is wrong.

Now, we have tried to address that by going out to the community and communicating as best we know how, but the reality is that we know that we're not going to be able to get to everyone, and the community when it developed the KSK rollover plan had understood that, and put in a criteria that if less than .1% of end users are impacted, then it was considered a success and we would not roll back the rollover.

The information that we were getting was suggesting that a bunch of resolvers were misconfigured, but that didn't tell us how many end users were impacted. So, it put us into a bit of the quandary. The input from the community was largely because we don't have information about end users and we know that the very large scale resolvers are configured correctly, that the right answer would be to move forward and assume that the breakage would be below .1%.

If you are actually running a resolver, which I know some of you do, then the way you can recognize the KSK 2017 is the key tag, which is 20326. That is found within the delegation signer resource record and it should show up within your configuration. The DNS key resource record is that – it's a little messy, but that's what cryptography does to you.

Let's see. The current state of the system. The KSK is changed under good conditions. The approach that we were taking was slow and cautious and the automated updated system would work over 30 days, but we've already passed that – long passed that. We were ready to go back I believe around August of 2017. The systems that were doing the automated configuration were already set up to have the KSK 2017 and we could have moved forward then, but as mentioned, we didn't.

Let's see. So, the rollover process, looking at it from the validator, it assumes that the resolver is configured with DNSSEC enabled and that automated updates are allowed. All validators should already have the new KSK in it. If it's not there, then you'll have to add it manually.

How can you tell if your resolver is validating? Well, there is a domain name out there called dnssec-failed.org and if you send a query to that and get back an IP address, then DNSSEC is not enabled. If you get back a serve fail error, then that probably means that DNSSEC is enabled. The dnssec-failed.org is deliberately configured so that DNSSEC validation will fail. So, if you get an IP address back, that indicates that validation is not enabled.

If you have access to a command line on a UNIX system or a MAC OS shell or something like that, there's a command called dig that will allow you to send the query. The dollar server there is the resolver, the IP address, or domain name of the resolver that you want to test and you just put plus DNSSEC at the end. And as it indicates in the response, when it has the header line and you say serve fail, that means that DNSSEC validation is enabled.

In the next slide, this is an indication that validation is not enabled. We frequently get the question, "Well, okay, how can I tell if the KSK 2017 is properly configured in my resolver?" The unfortunate answer … Well, historically the unfortunate answer is that you can't tell. There is no way, unless you actually have access to the resolver at a management level, to be able to check to see what KSK has been configured. This is changing. There's a new specification out called KSK sentinel that allows a query, especially crafted query, to be sent that will provide

information about what KSK is actually configured in the resolvers, but that does not have significant deployment as yet and it will probably take some time.

If you do have access to the management level of the resolver, you can do a number of commands -it depends on the actual resolver itself – to tell you what the KSK actually is. That's why there it gives you the information about the various approaches that you can use. But there's no way currently that most people can, as an end user, check to see what their resolver has configured for the KSK. And yes, this is known to be a flaw in the way DNSSEC was deployed.

This URL provides information on how to check the current trust anchors, and this again is from a management level, not as an end user level.

What you should see – should, if you are configured correctly at this point in time – you should see two trust anchors: the KSK 2017 with a key ID 20326 and the KSK 2010 which is 19036.

As mentioned, we will eventually remove the KSK 2010 after we have migrated to the KSK 2017. When that is, we haven't figured out exactly, but it would be some point in the future.

How do you see this information? Here are some slides that I'll go through fairly quickly that show the various ways of viewing the KSK.

So, with bind, it's actually pretty easy to see. Unbound, it's a little more complicated. You have to go digging through the DNS key response to

see what the KSK configured is. But it's still there and it says that both are valid.

If you see both KSKs are installed, then you don't have to worry about it. Everything is working the way it should. If not, then you have to go in and fix it. And the way you would do that, there's a URL at this page with how-tos on the various resolvers on how to actually fix this.

Where do you get the KSK? The official one is available via an XML file on the IANA website. You can also get it via the DNS. Hopefully, you have DNSSEC enabled, because otherwise, someone could spoof that response. And there's some other means that are probably more common because it's with OS distributions. When you update your operating system – for example, Microsoft Windows, if you're running Windows server, the automated … The software update will provide an updated configuration information that includes the new key.

What are the symptoms of a wrong trust anchor? As I mentioned, if you don't have the right key and we have used the new key in signing of the root zone, what you will see is a serve fail. The error message coming back for any query that you would send. If you have access to log files, you can look at that and see what's actually going on. But largely if you're an end user, the failure to look up anything will probably be a good indication that the KSK has not been updated.

So, looking at the future, at some point we're going to revoke KSK 2010 after we use KSK 2017 to sign the zone signing keys. There will be – unless the community tells us otherwise, there will be more KSK rollovers. As mentioned, the DNSSEC practice statement indicated we

would roll the key after five years. Current assumption is that we will continue to roll every five years or so and that will remain true. That is what we're currently planning on, unless the community tells us otherwise. There have been some within the technical community who have suggested that we need to increase the frequency of rolling the KSK to ensure that the KSK infrastructure exists to allow us to change the key if we need to and that it's constantly being exercised with the belief that if you don't exercise something, then when you actually need it, then it won't be available to you.

There have been others who have argued that this whole KSK roll thing was a really bad idea. We shouldn't do it ever again. But, at this point, the belief is that the right frequency to roll is every five years. So, after October 11th of 2018 we will start the clock for another five years and then we will roll in whatever that is – 2013 … 2023! Wow. Math is hard.

Some tools and resources provided by ICANN. There are a number of various tools that you can use to fetch and validate the trust anchor if you happen to run a resolver. We have information pages explaining what the KSK rollover is, why it's important and that sort of thing. There's a Python script, get trust anchor, that you can put in the automated system to run software periodically and it will check and pull down the new KSK if you need it.

We do have an automated update test bed to allow people who are using the automated system make sure everything is working correctly. That continues to run and people are, if you go to that website, it will tell you how to sign up for that automated testing service and that's

applicable to anyone who is running a validating resolver that has turned on the RSF 5011 support.

The KSK rollover page is where you find information associated with the KSK rollover from ICANN. You can find it under the quick links page.

One more time for the URLs that you might find useful, or perhaps not.

With that, I will hand it over to Andrei to talk about the KSK management at a ccTLD. Andrei, if you will.

ANDREI KOLESNIKOV: Thank you, David, very much. Well, first of all, I feel pretty safe having all this information on the screen and available online. I really wish that ICANN does its outreach good in order to deliver this important message to the people. I mean, to the people who created the resolvers. It's important.

I'll be a little bit more practical and repeat a couple of things. First of all, DNS is, even [inaudible] system, it's pretty much [inaudible]. Everything starts with a dot. Basically, assigned a dot. You sign the root zone and all the TLDs basically follow the same scenario. Basically, what it does, this DNSSEC is based on asymmetric cartography and we have secret key, which is a secret key. You keep it in a secret and you check the secret key with public key. Basically, there are two keys: [inaudible] which is zone signing key and the KSK K signing key. So, you basically sign the key with a key. This sounds like cryptography magic. I'm not very good on cryptography, but how it's done practically?

First of all, as far as I know, the ceremony which takes place in Los Angeles in ICANN headquarter, it's kind of show. It's kind of party. So, the whole thing, when you sign the zone, is going with a presentation with the trusted community representatives. Those people are [inaudible] who's not connected to the root zone management. It's one of the important aspects. [inaudible].

For the root zone or for the TLD, [inaudible] the guys who are keeping the recovery key, and they basically hold some physical thing on a flash disk, basically, which is a part of the KSK. So, when all the guys get together, they can recover the key.

But, if we go down to the next level, to the TLD, what is done in 2012, we actually signed a zone with [inaudible]. It's a big zone. It's about five million domain names. It is done in a locked room with no Internet connection on a computer connected to what we call the [inaudible] module. Basically, the secret device generates [inaudible] or [got] keys, [inaudible] for the rest of the world and you generate a key. You take it out on the flash, basically, and you upload this key to your resolver, main resolver, for the zone. Then, you deploy the key to the trust anchor.

So, basically, the whole system works under the [inaudible] that the TLD trusts the root. The domain, the guys who [inaudible] hold the domain name, they trust the TLD. That's the basic principle. You cannot pull out single parts on this wonderful schema. But the thing is that many domains still operate without DNSSEC records. I mean, the individual domain names. But, a lot of zones [inaudible] within this environment trust to the anchor and this important to maintain this [anchor trusted].

That's why the whole thing about the KSK signing is basically reflecting on the whole infrastructure, and first of all, on the resolvers [inaudible] using these trusted anchors.

We have one of the greatest experts on the DNSSEC with us tonight. Well, it's night in [inaudible] and day in America. Russ Mundy is with us and he is one of the guys who really understands every detail of the DNSSEC from the very beginning. I don't have any presentation. I am a SAP guy, but I am also ALAC guy and liaison to the SSAC. The idea was to give as much information about the DNSSEC to the ALAC and At-Large people because with this KSK process, it's important to understand not just how you [inaudible] the key or technical problem related to 2010, [inaudible] to really understand more than [inaudible]. It's very important to understand how the whole DNSSEC system is being [inaudible], how it works in the general. That's why I think we're doing a great job and Tijani proposed to run this webinar.

With this, I'm done with my presentation, which is very short. If anyone wants to know how to sign the TLD zone, they can contact me separately and I'll explain it in detail. Thank you. Tijani, back to you. I think it's question and answer now. Hello?

ANDREA GLANDON: Tijani?

UNIDENTIFIED MALE: His microphone appears to be muted.

ANDREI KOLESNIKOV: Why don't we just, since we have a [inaudible] with us, I think we can carry some questions from the participants of today's webinar.

ANDREA GLANDON: Would you like to go ahead and do the pop quiz now?

DAVID CONRAD: We have a couple of questions, a couple of hands up. Do you want to do those first?

ANDREA GLANDON: Sure.

DAVID CONRAD: I guess, Alan, since you appear to be first on the list.

ALAN GREENBERG: Thank you. I have two questions. We'll do them one at a time. You mentioned in passing a criteria that you might have used or might still use to roll back the rollover. Can you tell us just what rollback the rollover means and what kind of timeframe? Is it something you can roll back and what are the details?

DAVID CONRAD: Right. The original rollover plan, as defined by the community, had in it a criteria that said that we would roll back to the KSK 2010 if more than .1% of end users were negatively impacted by the rollover. That was the

criteria that the community had defined, but sort of in retrospect, it wasn't really very clear what that actually meant because, at that time and still today to any real extent, we don't have an easy way of establishing how many end users are being impacted by the rollover.

As mentioned, that will hopefully be remedied to some extent with the KSK sentinel work that's being standardized right now. I believe it's just about to go into last call within the IETF and then it will be implemented. It's already implemented in a couple of resolvers as a pre-standard, and unfortunately implemented – an earlier standard was implemented by a third resolver.

But, once that standard comes out, then more people with implement it and then there will be a delay, a period of time in which that [inaudible] gets deployed out there. At that point, we would be able to actually get a better sample of end users who are being – that configuration of resolvers by the end users, which would lead us to be able to estimate more accurately how many would be impacted by the rollover failure, if it did fail, of course.

The timeframe … So, the implication of doing a rollback would mean that we would stop using the KSK 2017 to sign the root and go back to using the KSK 2010 to sign the root.

So, in theory, that would mean that there would be one or two zones, however long it took us to identify there was a problem that would require the rollback. Then we could, since both keys are supposed to exist but the misconfiguration would only be having the old key, then falling back to using just the old key to sign the root zone would be a

relatively straightforward and, hopefully, relatively painless action. But, unfortunately, the way the DNS works with caching and all that sort of stuff, there would undoubtedly be a significant amount of disruption.

So, rolling back is something that we would prefer to avoid, of course. That's one of the reasons we decided to suspend the rollover until we had a better understanding of what the data was showing.

One thing that I did want to say that I forgot to mention is that – it isn't in the slides that I gave. But one analysis that we've done with folks at APNIC actually tries to correlate the 8145 announcements that we see at the root servers and data that the APNIC folks are able to get using Google Ads and doing DNSSEC enabled queries from browsers.

The information that we've been able derive using that approach is much more reassuring in the sense that the figures that we've been able to establish look like less than 0.05% of resolvers are currently misconfigured which is a much - sorry, end users would be impacted at 0.05%, which is obviously below the .1% that would trigger the rollback.

ALAN GREENBERG: Thank you. So, if I can summarize, to rollback the rollover, essentially it is not physically, but conceptually, flipping a switch but then there would be a lag due to caching, which might be hours or conceivably even days.

DAVID CONRAD: Right.

ALAN GREENBERG: But, in general, it would start rolling back pretty quickly, other than any long-term caching that there may be at various places.

DAVID CONRAD: Right. It would come into effect immediately on the publication of the zone that was signed with the KSK 2010 instead of the KSK 2017.

ALAN GREENBERG: Okay. The second question is, from the answer to that question, I would presume that the only impact ... Revoking the 2010 KSK three months later simply means you cannot do a rollback at that point. There's no other implication associated with the revoking. Is that correct?

DAVID CONRAD: Exactly.

ALAN GREENBERG: Thank you. Okay. I have more, but there's other people in. Let them go first.

DAVID CONRAD: And I should say, since Russ is on the call – Russ, feel free to correct me where I get things wrong.

RUSS MUNDY: Well, not a correction, David, but I did want to add just a little bit to the question of the impact on end users. That is that almost every end user, whether they know it or not, have available to them two or three (sometimes four) DNS resolvers. If any one of those resolvers is functioning properly, they may be a little slower in getting the answer, but they will still get the answer.

So, if they, say, have three resolvers configured and two of them are in only KSK 2010 nodes but the third one has KSK 2017, they will be served and service will continue because the resolver itself will then be using the 2017 key from that one successful resolver.

So, it not only makes it harder to figure out what the end user impact is, it also makes it easier for end users to get it right by having multiple resolvers. Thanks.

TIJANI BEN JEMAA: Thank you very much.

DAVID CONRAD: Yeah. One of the joys of the DNS is that it is almost a chaotic system in terms of the behaviors and it all depends on what end users have configured. So, as Russ says, it's sort of hard to figure out exactly what's going to happen. Olivier?

OLIVIER CREPIN-LEBLOND: Thanks very much, David. Can you hear me?

DAVID CONRAD: Yeah.

OLIVIER CREPIN-LEBLOND: Excellent. Thanks. I've got three small questions. The first one is why does the rollover itself need to be on exactly October the 11th?

DAVID CONRAD: When the community sat down and tried to identify the best dates, taking into account the need to do the rollover with the quarterly key ceremonies that we use to sign the zone signing keys, it basically … Folks sat down and tried to dodge all of the holidays and weekends and all that sort of stuff and came up with October 11, 2017. Since we suspended, the decision was to just move it one year to simplify the communications aspects. Some people had suggested we could pretend that the actual date was 2018 all along and we made a typo, but we would never do that.

OLIVIER CREPIN-LEBLOND: Thanks, David. Now, the reason for it being October the 11th in 2018 is actually later in the week than in 2017.

DAVID CONRAD: Right.

OLIVIER CREPIN-LEBLOND: So it may be a bit tight in going on to the weekend if there are troubles. That would be … I wondered whether there was a specific reason why it would be on October the 11th.

Second question is you've shown us some graphs on your presentation.

DAVID CONRAD: Tried to, yes.

OLIVIER CREPIN-LEBLOND: Yeah. I've looked at the original presentation that's downloaded. There's a spike on the, [inaudible], the first of April or is it in fact the day before, the 30th of March? Have you isolated why there is a spike there? 31st of March.

DAVID CONRAD: We tried to look into that. It turns out that one of the reasons that were much more comfortable – and I apologize, I forgot to mention this – was it turns out that there is a VPN software that had hardcoded the original KSK into their VPN software. And the reason that we were seeing the number of IP addresses that we were, the unique source IP addresses that we were seeing, is because this VPN software, which was in client machines, would connect up to the Internet, do DNS queries and announce the 8145 data indicating that they have only the KSK 2010 configured. So, it was actually a relatively small number of machines that were roving around the Internet and connecting at different places and showing up as unique IP addresses.

We believe that spike that you see was the VPN software developer releasing a new set of code that impacted a larger number of the VPN clients, which caused them to notice something odd going on, and the statistics that we're getting since around that time have been going in exactly the correct direction that one would want indicating that going back down.

At one point, the number of unique IP addresses that were showing the misconfiguration was upwards of around 20-22% and we were getting quite nervous about that. But, then, I think Wes Hardaker at USC, University of Southern California, doing a bit of research discovered this VPN, contacted the vendor, and things all of a sudden started getting better.

OLIVIER CREPIN-LEBLOND: Okay, thanks. The third question, I'm going to be a bit provocative of this – purposely provocative. I've just come back from RIPE meeting in Marseille and there was a presentation by Willem Toroopthat was entitled "Sunrise DNS-over-TLS! Sunset DNSSEC?" Effectively, it was making the point that if you would do everything as DNS over TLS, then effectively you could just get rid of all this DNSSEC stuff. What's your feeling on this?

DAVID CONRAD: So, that has been a suggestion that's been made on multiple occasions. My view is that DNSSEC protects that actual data, not the transport of the data. DNS over TLS is protecting the transport of the data. This is similar to arguments that were made quite some time back with the use

of DNS Crypt, which was specified by a scientist called Dan Bernstein which is a very interesting technology, but it protects the transport, not the actual data itself.

My view is that it's better to protect the data, but regardless, both should be done. The advantage of defense in-depth is quite significant, and while DNS over TLS solves a particular set of problems, it doesn't necessarily solve all of the problems and it does not protect the cache the way the DNSSEC does.

I might ask Russ if he has any thoughts on that particular topic.

RUSS MUNDY: Thanks, David. This has been an area that's been debated significantly over time, and what seems to be the end position of people that are pursuing different types of mechanisms is they have, through various sets of research and testing, developed their ideas of what their best mechanisms are and it tends to be whatever the mechanism is that they're most familiar with.

It usually ends up where there's some weakness at some point that, for instance, in an earlier alternative name system approach to DNS, they actually wanted to make use of a technology that was even different than the one that David mentioned. I think it was called Beehive or something. And it needed an initial security mechanism like DNSSEC. So, in the end, they still developed the mechanism [inaudible] technology.

ANDREA GLANDON: Excuse me, Russ?

RUSS MUNDY: Yes?

ANDREA GLANDON: I'm sorry, can you speak up a little bit more or pick up your handset if you're on a speaker phone? We're having a very hard time hearing you.

RUSS MUNDY: I apologize. I normally blast people out. Sorry. In the end, you had to come up with a mechanism to essentially do what DNSSEC is doing to facilitate this to begin with. So, there are, as David said, huge advantages to [inaudible] and that's often the best answer for questions when people ask about the different technologies. You need to use multiple of them all the time. Thanks.

DAVID CONRAD: I'd also like to add that one of the things that DNSSEC provides, that DNS over TLS cannot provide, is a mechanism that actually protects against the class of denial of service attacks. DNSSEC provides back an answer that, when you query for a name that does not exit, that actually allows you to reduce the opportunity for a type of denial of service attack that relies on flooding name servers with queries for names that do not exist.

This particular feature, which is implemented in recent DNSSEC capable resolvers is known as [NSEC] aggressive use actually provides a way of stopping attacks, a particular form of denial of service attack, that's

quite common with IOT devices, where IOT devices configured to query a whole bunch of random names. The DNSSEC [NSEC] aggressive use actually allows for the filtering of that level of attack at the resolver level.

Hadia?

HADIA ELMINIAWI: Hello, David and all. We are now rolling the root zone KSK because it's about time. More than five years have lapsed. But, what if the key is compromised? In such a case, I assume that they key has to [inaudible]. So, actually going through the root zone KSK at least once is a necessity. And aside for that, I assume it's already in place.

So, I know the data available is not assuring, but still my question is why are we so hesitant about all of this?

DAVID CONRAD: So, from my perspective, the reason for hesitancy is because we're sort of playing around with the airplane engines while we're in flight and you want to be really careful when you do that because, when you have one engine, if you break it, you're going to have a bad day.

The community came up with a very careful and considered plan to do the KSK rollover and we're following that plan as carefully as we can. As I mentioned during my presentation, there are folks who suggest that we need to do the KSK rollover more frequently to exercise the infrastructure.

Personally, I would be interested in something slightly different, which is the algorithm role, to actually change the algorithms because I'm reasonably confident that the infrastructure that we've built is resistant to key compromise, but it's always possible for someone to come up with a new factoring attack that would require us to change the algorithm to one that is not susceptible to a particular attack. In addition, there are better algorithms for security that have been invented since the algorithm that we're currently using was deployed and I look forward to actually changing the algorithm to one of these new algorithms.

Alan?

ALAN GREENBERG: Thank you. One of the comments that the ALAC made was asking ICANN to provide some sort of utility or URL that would allow someone to do the kind of query you are talking about and verify if indeed the resolver they were using was DNSSEC enabled.

The belief is, for large parts of the community, certainly in the developing world, it probably isn't. And knowing that your resolver is not DNSSEC enabled essentially takes the pressure off in your particular region.

However, the catch is what to do if it says it is DNSSEC enabled and you don't know whether the new key is installed or not.

What we were looking for ICANN to do was to provide something which doesn't require a technical person to interpret the answer that comes

back, but moreover, will provide some guidance for what they do at that point.

Now, if you as regular user call up your ISP help line and start asking them about KSKs, the person you're talking to won't know what you're talking about and you won't know enough about it to be able to talk to someone who might know.

So, we really almost need a script or something to point someone to that's really very, very turnkey and will enable a large number of users around the world, and certainly At-Large, although we are not unique – At-Large has access to a very large number of people who could do this kind of test and alert their own ISPs if indeed there's potentially a problem. But we really need something that's turnkey and simply pointing to someone who will – a URL that will issue a technical message and then not know what to do with it is not sufficient.

So, I'd really like to reopen that issue and see if ICANN can do something. It might well relieve the pressure and it will provide some guidance to fix problems if indeed there are some around the world. Thank you.

DAVID CONRAD: Thank you for the question. That question of what to do for end users was something that has consumed quite a number of cycles internally within the organization, but also within the technical community that we consulted.

Part of the challenge was that we had initially within the organization had proposed that the coms planned tell end users to call up their ISPs and ask. But the network operators community thought that was a stunningly bad idea because they would then be inundated with questions that the person who was asking the question didn't really understand what they were asking and it's unlikely that they would understand the answer, and even if they did understand the answer, one of those questions would be sufficient. You didn't need to bury the support line with those questions.

So, coming up a tool was something that we had discussed internally. The challenge at the time, still remaining a challenge, is that while you can tell remotely whether a DNSSEC validator is enabled, you can't tell what keys they're actually using because of the way DNSSEC was implemented.

So, we could tell them that DNSSEC is enabled and that, as you point out, would reduce sort of the universe who would need to investigate further, but you would still have the problem that you would, particularly if some ISP had configured, say, Google's public DNS at 8.8.8.8 or the 1.1.1.1 of Cloudflare or 9.9.9.9 of the Quad9 folks or any of the others that have enabled DNSSEC.

The end users would then be in a position of having to call their ISPs and flooding their ISPs with these questions which would almost invariably irritate the ISPs to no end because they'd get very tired of that question very quickly.

So, we're still struggling with what do you do if you find out that your ISP is actually doing the right thing and turned on validation, but you don't want to annoy the ISP. So it's still something that we're trying to figure out.

ALAN GREENBERG: We have a session in Panama. We'll talk more about it then.

DAVID CONRAD: Sounds like a good plan. Andrei?

ANDREI KOLESNIKOV: Yeah. Thank you. I have a question [inaudible] can answer or Russ regarding the [inaudible]. This is like a dual problem. Is it because of the [inaudible] material being collected by the resolvers or is it based on the [inaudible] cryptography algorithms will be broken sooner or later? What [inaudible]?

DAVID CONRAD: Yeah. It's the latter. At least the understanding that I have with the RSA SHA256 algorithm that's being used is that it's stunningly unlikely that it will ever be compromised with known technology. However, RSA is known to be vulnerable to quantum crypto. There are people who believe that's a risk.

One of the challenges with the existing algorithm is that the signatures are really big, which means that it gets harder to … You end up having

bigger packets which cause problems, particularly with IPv6, so there is a desire to move to a newer algorithm that has better characteristics in terms of resistance to crypto breaking, but also has much, much smaller signatures, like half the size of signatures and that would make a number of things much easier from an operational perspective.

TIJANI BEN JEMAA: Thank you very much, David.

DAVID CONRAD: Sure. Were there any other questions?

TIJANI BEN JEMAA: I don't see any questions. If there is no question, I will ask Andrea to go to the pop quiz, please.

ANDREA GLANDON: Yes, thank you. The first question you'll see on the right side of your screen at the bottom. Why is it important to rotate the key? You can type your answer and make sure you hit the submit button next to the open pod.

We have two people who have answered, so I'll go ahead and broadcast those results so that you can go over those, David. Oh, we have a few more. Can you see those answers, David?

DAVID CONRAD: Sure. The answers that were … Oh, still more answers coming in. The answers provided include DNS security, because new crypto algorithm is good for security, security concerns, motivation to comply with community directive to change every five years, something in French I think, and to prevent any.

The answer that I would probably put is to exercise the infrastructure in case we ever need to roll the key, although having the ability to change the algorithm is good. Similarly, as just came in, changing a passcode … Changing your password frequently is sometimes considered good crypto practice.

ANDREA GLANDON: Okay, we'll go to the next question, and that is what does DNSSEC prevent?

Go ahead and type your answer in the spot and then make sure you hit the submit button. I'm going to go ahead and broadcast the results, David, so that you can see them as they're coming in. It looks like we have three answers so far, David. Are you able to see those?

DAVID CONRAD: Yes, I sure am. Got another one.

ANDREA GALNDON: Great.

DAVID CONRAD: Up to five.

ANDREA GLANDON: Okay. You can go ahead and go over those answers.

DAVID CONRAD: Okay. The answers provided: cache poisoning and the Kaminsky attack, bad stuff, one risk is cache poisoning, something I guess in French (not sure), DNSSEC protects the actual data so it prevents the alteration of data.

I would probably agree with that last one. DNSSEC protects the actual data. So, it prevents the alteration of data. That's how cache poisoning is prevented, so that's also a good answer. And of course, bad stuff is always a good answer.

ANDREA GLANDON: Thank you. Those are all of the pop quiz questions. Tijani, did you want to go ahead and close the call?

TIJANI BEN JEMAA: Yes. Thank you very much, Andrea. Now, can we please go to the evaluation questions?

ANDREA GLANDON: Yes, thank you. Just one moment. Okay. For the first question, how was the timing of the webinar today, 21:00 UTC? a) too early b) just right c) too late.

I will go to the second question. One moment. How was the technology used for the webinar? a) very good b) good c) sufficient d) bad e) very bad.

Question three. Did the speakers demonstrate [inaudible] of the topic? a) extremely strong b) strong c) sufficient d) weak e) extremely weak.

Next question. Are you satisfied with the webinar? a) extremely satisfied b) satisfied c) moderately satisfied d) lightly satisfied e) not satisfied at all.

Next question. What region do you live in the moment? a) Africa b) Asia, Australia, Pacific Islands c) Europe d) Latin America and the Caribbean Islands or e) North America.

Next question. How many years of experience do you have in the ICANN community? a) less than one b) one to three c) three to five d) five to ten or e) more than ten years.

On the last question, what topics would you like us to cover for future webinars? You can go ahead and submit your answer in the box and make sure you hit the “send answer” next to it.

Thank you, Tijani. Those are all of the evaluation questions.

TIJANI BEN JEMAA: Sorry, I was speaking to myself. Thank you very much, Andrea. This question is important to answer. If you don't answer it now here on the Adobe Connect, please send what are your preferred topics that you want us to address in our future webinars. This will help us to figure out our [inaudible] program.

Thank you very much, all. I would like first to thank David for his very good presentation, and for his answers. I would like to thank our wonderful staff, our interpreters, and all of you who attended this webinar. Thank you, all. This webinar is now closed. Thank you very much.

DAVID CONRAD: Thank you, everyone. Bye-bye.

ANDREA GLANDON: Thank you. This concludes today's conference. Please remember to disconnect all lines and have a wonderful rest of your day.

[END OF TRANSCRIPTION]