ICANN74 | Policy Forum – DNSSEC and Security Workshop (2 of 2)
Monday, June 13, 2022 – 10:30 to 12:00 AMS

KATHY SCHNITT: Welcome to the DNSSEC and Security Workshop Part 2 of 2. And now I'd like to turn it over to Dan York.

DAN YORK: Hello. Good afternoon, good evening, good morning, wherever you might be joining into this from. This is the second session of the DNSSEC and Security Workshop at ICANN74. I am Dan York. I'm from the Internet Society. I am also a member of the Program Committee, who helped bring this program together.

So, in this session that we're going to have here, we will have myself giving a quick little summary about our numbers and some things. We have an announcement to make. We have a couple of small presentations. And then we also have a longer panel that Steve Crocker will be running around DNS automation here with.

So let's begin and talk through some of what we've seen in terms of the numbers and things that have been going around the world in terms of this. For many years now, we've been tracking the growth of DNSSEC validation because recall again that there's

two sides to DNSSEC. There is the signing of domains and then there's the checking of domains, which we refer to as validation in this industry. And it continues to climb. These are the numbers coming out of Geoff Huston/George Michaelson's APNIC Labs. And we continue to see this. What this says is that, according to their metrics, we are seeing about a third of all DNS queries being validated with DNSSEC. They use a system that goes and measures this. So it's a continued growth. We'd like to see that. Ideally, we'd like it to keep ongoing even higher, but this is a good path to see. It's at least going up and not going up in that kind of space.

If we look at this chart here, this is a summary that shows where DNSSEC validation is happening. And you can see from the very top where you have the highest percentage, all the way down to some areas where there's not a lot of DNSSEC validation happening. And this is good to show us in part where we're seeing this happen. Obviously, a lot of places in Europe, Southern Asia—good percentages there in what we're looking at.

Excuse me for a second. I'm having an issue with my—there we go. This is the observed delegation signer records with are in the top-level domains and secure the global chain of trust. And this has really been a great chart to watch over the while. You can see it yourself at stats.dnssec.tools.org. But it shows the continued growth of DS records overall, which is for both the TLDs (Top-

Level Domains) and the second-level domains and all of that across this. So great to see this kind of thing happening.

This—and Eric Osterweil is going to speaking next of DANE in some of these spaces—is to show that we're seeing a continued increase growth of the use of DANE records for signing Mail Exchange records (MX records) for secure e-mail exchange. The bottom line there in growth is showing the continued growth in valid RPKI prefixes. This is authentication of routes. The Routing Public Key Infrastructure shows us that. So we're seeing a continued growth of that, which is good. Eventually, we want to see the valid go across and replace that. So over the years ahead, we should see this chart continue to merge, where ultimately the green lines goes beyond the yellow line and goes on from there.

This is also another example showing a number of origin authentication signals. You can see them by RIR as it continues to grow up.

And then we also have seen the continued growth—stability, really—of the ccTLDs that are around the world. We've been tracking this really since the beginning of these DNSSEC workshops. We've been tracking this metric. The big one that has been announced since the last DNSSEC workshop was that Rwanda signed a dot-rw. That has been the movement that has happened in this past bit. So it's great to see that.

If you're interested in more of this kind of thing—more information—you can go to these different DNSSEC resources that you see here: [stats.dnssec-tools.org], stats.labs.apnic, and also some RPKI as well.

With that, I want to go on and talk a little bit about the deployment maps that we have shown historically here and are part of what we do here within this broader context. So these deployment maps were first originated back in about 2009 or so with Steve Crocker's Shinkuro, Inc., who started to collect the data, made these maps, and did all of this. We started to include them here in the DNSSEC Deployment Workshop. This presentation used to show the maps broken out by region, by country—all of that.

The interesting aspect about this database, when Steve's group first developed this, was that it included the historical data, but it also included future predictions based on announcements and presentations on places like this that said we're going to be deploying things in the future and all of that. And they were e-mailed out weekly. And Shinkuro continued to do this for quite a long time and presented here at this workshop at various times on that.

It does have five deployment states, ranging from experimental, announced, partial—we could see that the zone was signed but it wasn't linked into the global chain of trust—DS in root, and then

the operational. And this is how it operated for a good number of years. And we'll get to where it changed. It also included information in the database about how reliable this information was or not.

Then in 2014, the Internet Society became responsible for the maps as part of our Deploy360 Program. We made a couple of changes over time. One was we did have to stop doing that forward-looking aspect, as we just didn't have the time to keep up with what was being announced across the industry in some way. But we did continue to operate them. We have the e-mail mailing list. There's about 70+ people, I think, who still get these maps that are issued every Monday morning. One probably went out shortly before this session started.

And then, in 2021, we also added a sixth state of DS automation. You aren't seeing the maps anymore because, in 2019, ICANN did decide that, as of ICANN65, we can't display maps with country borders. And there's a number of issues around where that happened, but the net is … That's why we don't show these precise maps anymore. We do continue to distribute them to people in e-mail. They're posted to the Internet Society's website, etc. But that's why we don't have the full maps in this presentation.

In 2021, back in March, at ICANN70, we added a sixth state, which was DS automation. That was showing the ongoing growth of

people using CDS records and CDNSKEY records to be able to go and have automation of the updates to the DS records. And with that state, there are currently seven TLDs that are in that state in a database.

So this has been the ongoing evolution, but today we're here to also talk a little bit about where it's going forward. So now that we have a new sponsor, which you're hearing about for the beginning of this—a new operator of these maps—George Mason University is actually going to start assuming responsibility for these maps. They're still on an Internet Society server at the moment, but over the next couple of months, we'll be transferring that.

And to share that information, I'm going to ask Eric Osterweil to open up his mic and speak to us a bit about what George Mason is going to do. So come on, Eric.

ERIC OSTERWEIL: Okay, cool. Thanks, Dan. Hopefully everyone can hear me. Yeah, we're really excited to play our part and help out and take over the DNSSEC deployment maps operation. We have some interesting ideas about where the service and its longitudinal dataset can direct some of our research. So I think, when we heard this might be a possibility, a number of us in the university, from the center that we're affiliated with, up through the dean, were real excited for the prospect of taking this real solid,

historical dataset that really doesn't, as far as we know, exist anywhere else and bringing it in to a couple other of the datasets that we have and starting to put it in front of a bunch of students and doing some deeper basic research on it.

So our aspiration is to take this and use it as an evolution point to build what we're currently calling the Internet Namespace Security Observatory. We want to use telemetry about keys and the DNSSEC hierarchy even below the TLDs[.] Even below the TLDs, some datasets that we have build out some additional telemetry for caches and their behaviors and make it more of a holistic [inaudible] of operational datasets and research that we do on them.

And to really help keep that from getting too far off in the weeds, we're currently investigating doing an advisory board of some folks that I think the people in this room might probably recognize pretty well to provide input, to provide some steering, and to help the research stay both relevant and also cutting-edge at the same time.

So I didn't put any slides together. I'll go ahead and I'll blame the hour. My background is lying to you. It's not a beautiful sunny day right now.

But anyway, we're real excited at Mason, and I think we're planning to come back to this and other forums and talk about what we've started to do, when we started to do it, and where

we're starting to head with it. And every time we show up to talk, I think everyone should realize that that's an opportunity for you to tell us what you think because we want to keep this a live, active research project. So your input is very important to us.

I think that's pretty much it, Dan.

DAN YORK: That sounds great. Well, I'll say, as somebody who has been the point person maintaining them since 2014, I am delighted that you all will be taking this on and bringing it in new directions or doing things with it that are beyond the scope of what we're focused on. So thank you very much, Eric, for taking this on. I think, for the DNSSEC community, it's been great to have these and to go from there.

So this concludes this part, but Kathy, if I'm correct, I think we move right into the next presentations with Eric, right?

KATHY SCHNITT: That is correct.

DAN YORK: Well then, let me turn it over to Eric to talk about a different topic that is part of his research.

So thank you all very much. Enjoy the rest of this morning session. And we'll talk to you in the question & answer session if you'd like.

Over to you, Eric.

ERIC OSTERWEIL: Okay, great. Let me go ahead and try and see if I can mess up sharing my screen. Okay. And—okay, cool. I see my slides on the [inaudible].

So, hey, everyone. I'm going to go ahead and, I guess, kick off the presentation portion of this session with some of the research that we're doing in my lab. This is joint work with students that are working with me—Minar Islam, Josh Yuen, Pavan Kumar Dinesh, Tomofuni Okubo (some of you probably recognize his name), and myself. And what we've been doing is we've been working on is DANE, as Dan alluded to earlier, but a different part of DANE than I think what you're tracking with the MX record-signing. That would be TLSA. We're actually focused on, what is it we can and should be doing with object security more generally in the Internet?

So what's motivating this research? So I think what we start off with is, should we be protecting data as it's in flight, or should we be protecting data when it's at rest? Or both? Admittedly, this is a high-level proposition, so this is just to motivate where we're going with things. We look at the security perspective of what we

have as tool suites in the Internet and see that there's something that's been lacking. And principally, what's going to motivate a lot of what I'm talking about is the observation that, when we want to do some next generation stuff, what we want to do is oftentimes secure data more than necessarily just what's, over the pipe, being transmitted.

So what we focused on here is, if we wanted to look at securing data between different organizations, what that might be for is when they're sending messages to each other. And this is a general, very high-level description of what we're calling object security. So for objects, what we consider an object would be a file, an image, a message, an e-mail, or sensor readings that might come off of devices or vehicles, etc. So these are all things that we are now going to just generally call digital objects when defining object security and for our purposes.

So what really becomes clearer and clearer the closer you look at this is that object security is just different. So we have Transport-Level Security. We have TLS with HTTPS built on top of it. We have a bunch of these tools. And this is hopefully not news to anyone. But when we start actually thinking about what you need to do to secure an object, it's just different. And the more you look at it, the more different it gets sometimes—well, in some ways.

And that's nominally all derived from the fact that these objects persist. They exist over time. When I connect to a website and use

transport security and then I've downloaded the webpage, then that transport session is over. The security proposition is done as well. But objects continue to exist. They sit there at rest, which means the security that's granted over them also has to have a longitudinal value and qualities.

So, for example, what motivates our research in this case is … Suppose I have a document and I want to protect because I want to send it to someone. Well, if I want to use a vendor-locked-in platform, like WhatsApp, then WhatsApp will promise that they'll do end-to-end security. And that's great. Let's say I take it at face value. If later on I want to send that document to someone else and I similarly want it to be protected, there's no guarantees necessarily about whether it'll be end-to-end secured, etc. Certainly, the previous guarantees don't apply outside of WhatsApp.

So this to us looks like, why don't we have that? This is a basic motivation for our object security work. But we don't have a de facto way to do that today. We have certainly lots of tools, but we don't have a de facto way to do that. There's no way to say, "I'm securing this object for transmission at some point on the Internet later on via some mechanism." So that is our starting point.

And in this talk, what I'm going to basically roughly outline for folks is that we think we have a pretty solid idea on how we could

actually propose to this in a general architectural way from the Internet's core upward. And it's DANE. And with that, we're actually planning to look into a bunch of stuff in the very near term, from mobile healthcare, for which we have electronic health records or Electronic Medical Records (EMRs). These are objects. They're going to need the same kind of object security. In some places, we already have some of that. Vehicle-to-everything communication. I think the first slide briefly talked about, what if a traffic signal and a fire engine and my personally owned vehicle all needs to interact with each other because of signaling changes? Smart and connect communities. And a whole bunch of other stuff.

So the first step, I think, is for us to not just build another security tool. In other words, if we have the aspiration to build something that fits as an architectural substrate for the Internet, we really have to understand what fits the setting. In other words, why haven't we got this already? We have things. I mean, nominally you can call TLS the transport-layer security of the Internet as named. So why don't we have an object security layer? We've had really mature cryptographic protections for a long time. We've had S/MIME. We've had PGP. So why is it that we don't have this general substrate?

And I think what really motivates our interest in DANE is that it solves a problem that, I think, when you look closely, we all would have agreed or still agree exists, which is that, if I have multiple

organizations, separate organizations, separate operations, how the heck are we supposed to learn each other's keys? We have the Web PKI for the web. And love it or hate it, that's what we use. But what do I have outside of that, outside of a pre-agreed set of certification authorities that are in a bundle for me? So that's really what has been missing.

But I think what we still need to understand—what still has stymied the development and deployment of an object security layer—is, what are the fundamental needs and obstacles? What is it that has to be there? We've tried a bunch of things, and it hasn't shown up. And I think we could keep trying a whole bunch of new things. But instead, we're taking a more principled approach. We're trying to evaluate, what has been missing and what do we need?

So one thing in that same vein I'll identify is that we are aiming at building a number of interesting tools on object security using object security for, like I said, m-health, B2X, etc. But when we've looked at these nominally separate sectors, what we see is that vehicle-to-everything communication nominally would be a little different than mobile healthcare, but we start to see a bunch of repeated requirements. If you want to do object security for each of these, you'll need to do things like inter-organizational key learning. You'll need to do things like per-entity crypto, which means that, if it's a handheld device, that'll have to do end-to-end crypto. If it's terminal or a computer, you'll have to do end-to-end

crypto. You'll also need usable tools and you'll also need automation.

So to address that last point on the last slide about needing to build something that fits its setting, an architecture should synthesize whatever a repeated set of requirements are. So we've taken this real seriously, looking at these sets of repeated requirements.

So this is for anyone who's thinking, "I'd really like to see a graphic that isn't technically deep." So here's a little bit of a view of why we think this is the right approach. So we have the DNSSEC as a core security substrate for the Internet. And we have DANE built on top of that.

And so, from here, what we are doing is, if we build object security on top of DANE, we're extending security assurances from the Internet's core upward. And at that point, we can secure objects at rest using some of the protocols I just mentioned a second ago.

But here's something else that's cool that we can do that I don't think really exists in some of the other locked-in platforms. If we've got secure objects as a core Internet architectural substrate, then, when you secure an object, you can send it over different mediums, and it's the exact same thing. I can send the same object over chat as I do over e-mail because I don't need to know that I was going to send it over one or the other ahead of time. It's transport-agnostic.

So to do that, this is the tool suite that we've built. So we've built this basically a live, experimental apparatus. We've got this as tools because we're doing some analysis, but we've got this as tools that you can use to secure whatever you want going forward.

And we've started with e-mail. In other words, we've implemented secure S/MIME using DANE for e-mail purposes. And what we think is that this is going to give us a great opportunity to see how anybody who wants to use this is able to (or not) use object security for e-mail. The toolset we've built is called Kurer, and that's, say, a mail user agent plugin that does S/MIME using DANE, and resource certification, a management portal for DANE, in live zones at daneportal.net.

And our view is that the currency of object security for the Internet might as well just be PKCS7. So PKCS7 is something that has been out for a while, a long time, and it basically says, "Here is an object with security on it." So we're experimenting: can we use that as the Internet's parlance for object security using any number of tools and different transports?

And what we're interesting in seeing is, does this work? And can we evaluate whether it's working? Can we actually measure quantitatively that this is in fact beginning to bear weight in scaling?

So recall that we had this repeated set of requirements. So inter-organizational key learning. We are using S/MIME with DANE to address that. Per-user key enrollment. That's where daneportal.net solves a problem for us. Human-usable tools. I believe that anyone who is trying this out … What do you think? Because we find it pretty usable. And a framework to enable security automation is where we're headed with this. In other words, I think object security is absolutely necessary for the future of Internet security, but whether it's sufficient or not will depend a lot on what we actually want to do with it.

And so with that, we have a project that's underway that some call invisible securities. It's first up in what may or may not be a new sort of security approach that we're taking called entity security.

So anyway, I'll move really quickly. I don't know how I'm doing on time, but I suspect I'm probably running over. So real quickly, daneportal.net addresses this part of the proposition. Kurer addresses this part of the proposition. And with that, we've got end-to-end crypto—tools I hope you all will consider taking a look at. And so daneportal.net is the URL. And this is where, for any domain name that you hold if you're the administrator, you can go and you can actually get DANE going for it right away.

And Kurer you can install on either of the two of the growing set of platforms. So we have it working across platform Outlook—so

I don't need to know whether you're doing that on a Mac or a Windows machine; just Outlook—and Thunderbird.

And I'll do a quick walkthrough, as I'm starting to sense that I probably am short on time, but I don't have my timer up. So daneportal.net looks like this. There's actually a fulsome guide on there if you want to actually play on it. And what I say here doesn't give you enough details. You can go and actually see the details. Students have done a great job, who I believe are online today right now. So create a user portal account. Add the zone that you administer. We use the ACME protocol to verify that, if you say you are the domain holder, you are the domain holder. It doesn't go live on our system until you've actually passed the ACME challenge.

At that point, it basically builds a zone cut at _smimecert, which is part of the RFC for where S/MIME-DANE will start managing things. And we'll run that for you if you want. That's the point of using this portal. We'll run the _smimecert part. You still run your zone and manage DNS wherever else you want to do it. But this is how you would delegate to use the DANE gobbledygook.

And that point, what the real innovation of this portal is, as the domain holder, I don't necessarily want to run the management of the crypto keys for those that are below me. I want to give that responsibility to the e-mail holders. I can give e-mails to lots of people under Osterweil.net. And so they become denizens, they

create portal accounts, and I authorize them to manage their domain name under my domain name. And that's what they do. And they can create certs. You can create certs through the portal or you can use your own. So you can have us generate your secret key or you can keep it secret yourself. We don't really care.

We just help out. And then you upload it to the portal. And this is where one of the tools of the DANE that I think has gone unheralded a little bit. When you put your keys up, you can put a whole bunch up and you choose which ones to authorize as being live in the zone. And certainly at any point you can deauthorize them. So it's not revocation. You're not revoking. But as soon as you deauthorize a key, it's no longer going to be used. So as we go through with object security, it's like pausing your credit card. I'm not sure if I lost my credit card. I'm going to pause it so no one can use it. Oh, cool, I found it. Thank goodness I didn't cancel it. Same kind of thing here. DANE gives us that tool, but until we've played with DANE in object security, we haven't really known if we needed it or not. So here we are. We're evaluating whether or not this is actually a cool idea or if it's just noise.

And Kurer is a snap as well. Here's the webpage to download and install it if you want to give it a shot. It's kurer.daneportal.net/install. And it's a snap to put in Outlook. You go to My Add-Ins, you add from a URL, and, after you've configured your crypto keys, you're done. You can send things on its way. We can do signatures, encryption—like I said, PKCS7. And

it automatically looks at what comes in when you've got the add-on installed, and if it sees Kurer messages, it'll go ahead and do the crypto verification for you if it can or it'll try and tell you if it does work or it doesn't work.

So we would love it if you would all consider using it. It's live today. When you configure it, there's an opt-in option to participate in our user study. We'll never look at your e-mail. We'll never look at any sensitive information. We care only about configurations. And you have the right to be forgotten through our framework. So it's something where if you want to play with it, it'll actually help us do an analysis of whether this is really the right object security architecture for the Internet.

But win, lose, or draw with that particular study, what we're really excited about is we think that this is finally closing some loops. DANE and DNSSEC have offered a lot of promise for a long time, and I think this is the kind of approach that is poised to deliver on those promises.

We're real excited about plugging this into mobile healthcare. Like I said, electronic health records/electronic patient records are things that are very sensitive. And having object security guarantees I think are really critical. Same thing with smart and connected communities. 5G, next G and the IoT that's associated with it, vehicle-to-everything communication—all of these would require object-level security, I believe. And so just like those,

they've got the same repeated set of requirements that we've seen in other places and addressed with this framework. And so what we think is this will pave the way for real exciting stuff right around the corner.

And the next step that we think this might lead to is something we're calling entity security, where policy semantics around what the crypto needs to do and not do is really critical. And that's some work that hopefully you'll be hearing about in the upcoming future [inaudible].

And with that, I think I'll pass it back.

DAN YORK: Actually, Eric, you have five minutes.

ERIC OSTERWEIL: Oh, man.

DAN YORK: You did a good job.

ERIC OSTERWEIL: The coffee works.

DAN YORK: It's exciting. I guess, before we go to the DS provisioning panel, does anyone in the session there or online have questions for Eric?

Russ has his hand raised. Go ahead, Russ.

RUSS MUNDY: Thanks, Dan. And thank you, Eric, for the presentation. Extremely interesting. As you know, I've been interested in DANE for a very long time. This looks like a really cool advancement.

I did have one question in terms of the operational flow. If one chooses to participate in your experiment, does that mean then that all of the e-mail would need to flow through your portals? Or is it architected in a way that you aren't in the critical path?

ERIC OSTERWEIL: Excellent question, Russ. And it's great to hear from you again after a long time. No, absolutely not. We have absolutely nothing to do with user data. So, yes, if you use our tools, we are not involved in anything, except [our] zone might help you look up the crypto keys. And we don't track that.

So, no, we stay all the way out of it. The user study just simply says, when … There's some interesting configurations. Like I said, when we looked at object security, we looked at some things that were non-obvious and were like, "Oh, that's cool." And so look at,

do you want do default signing? Do you want to do default encryption? Do you want to preserve encryption? What do you want to do when encryption fails? Etc. Those configuration options we're looking at if you let us. If you want to opt out, we don't look at anything. But certainly we never look at your e-mails. We are not involved in the control path of e-mails. Those stay wherever you send them, and we are way out of that. So it's a very good clarification. Thank you.

DAN YORK: Shumon?

SHUMON HUQUE: Hey, Eric. Great presentation. So I had one question for you. The IETF has a current working group, DANCE, which is focusing on DANE [for] client authentication—mainly focused on transport initially, but I think they want to branch out eventually into object security. Are you planning to bring your proposed work and architecture for discussion at the IETF?

ERIC OSTERWEIL: Yeah, we definitely want to do that. In fact, I think, for the upcoming Philadelphia IETF, we were thinking of plugging into that, absolutely. It's not a side thought. That's a primary motivator for a lot of this work. So we haven't plugged in there much yet because we've been getting things cobbled together. As

opposed to showing up with vapor, we wanted to show up with something we can kick around. But, yeah, we'd be real excited to get it. I didn't realize the working group was thinking about object security.

SHUMON HUQUE: Great. Thank you.

DAN YORK: Excellent. And if you have a question, as Kathy mentioned in the chat, we are using the Q&A pod. We have probably time for one more if somebody has one.

I have a question, Eric, which was, what was the genesis of the Kurer name?

ERIC OSTERWEIL: So I'll give full credit to my students. They came up with it because they were looking at … Apparently, it's a Danish Viking rune, as I understand it. So if you look at Kurer, it's got a little "k" kind of thing. They're way better at that kind of stuff than me. And apparently that is actually a rune that just happens to look like a "k." And as I understand it, that's where they got it. They got it from the fact that our reference library is called Lib Cnut, which is a Danish Viking. So, yeah, we kind of went heavy on that.

DAN YORK: Ah. Okay. All in on the Danish Viking side of things.

ERIC OSTERWEIL: Yeah. Hey, why not?

DAN YORK: All right. Well, thank you very much—oh, Wes Hardaker has dropped a note in the chat. He is the working group chair of the DANCE working group within the IETF. And he says, "Object security would be a bit outside of our current charter, but it's well worth rechartering for some future point." So that's the word from the chairs/co-chairs of the DANCE working group.

ERIC OSTERWEIL: Well, I think we would look forward to working with you all. And maybe we'll talk about the IEPG or something like that or whatever you all think in hallways, etc. So look forward to engaging with you all and figuring out how the stuff we're doing could help fit, etc.

DAN YORK: Sounds great. Well, thank you, Eric, for your time here and brining this work to us. And, people, please do check out daneportal.net to learn more about Eric's team's contributions and how you can get involved.

And with that, we will conclude this part of this morning's session, and I will pass it over to Steve Crocker to begin our next panel. Thank you, all.

STEVE CROCKER: Thank you very much, Dan. I'm trying to figure out how to turn on my video, which is not essential. "Start Video." There we go. Thank you.

So now we move into the long-running Saturday morning cereal kind of presentation on a very particular portion of the deployment problem space. Shumon Huque and I have been running this panel for … I guess we're now in our third year. And the focus is on how to automate the ragged edges of the DNSSEC provisioning process that we're not fully appreciative of at the outset of the design of the protocol. So there are two aspects of that. Oh, so, Kathy or Kim, you're running the slides? Please. Thank you. So one aspect is automation of the DS updates, and the other is automation of coordination between multiple DNS providers when you have independent DNS providers signing the zone on behalf of a customer.

Now, both of these are in some sense an outgrowth or consequence of the fact that DNS providers were not taken as separate entities in the original conception. So in particular, in the registrar/registry model, a lot of registrars provide DNS service for their customers, and there's no problem if they do the

signing and they can convey the DS record up to the parent zone, up to the registry. But if there's an external DNS provider that is signing the zone, then there is a gap, a little gully, that you have to cross over to get the keying information. So we'll go into this in a little more detail.

But that's the genesis of the problem space that we've been tackling. And this panel has been the place where we focused on the progress being made in both of those directions.

Next slide, please. Here's the agenda for the panel. We organized this at the expert level, which means very little tutorial-level information and short and fast-paced presentations. The slides are all available. Contact information is all available. And as I said, this is an ongoing series. This is Episode 8. And we intend to continue for however long it takes to wrestle these problems and get them under control.

I'll talk for just a minute or two further, and then we'll go through the individual presentations. Next slide. Next slide. So as I mentioned, one challenge is, if you have a separate DNS provider, and they roll the key, one of the consequences, one of the requirements, is that a DS record has to be created in the registry that matches the new keying information.

What are the ways to do that? The red arrows on the right indicate solutions that are based upon polling or pulling, if you will, information from below. And the dotted line on the left—the blue

line—represents pushing the information upward. And the difference between the solid line and the dotted line, which is a shift from the previous nomenclature that we've been using, is that the solid lines are automated solutions, and the dotted line is a manual solution. So it's certainly possible for the registrant to take the keying information out of the zone that is being assigned and served by the DNS provider and manually copying the keying information—everybody should be pausing and grimacing at this point—and provided through, say, a web interface to the registrar, who will then push it via EPP up to the registry. Not a desired solution, doesn't scale well, is error-prone, and so forth.

Next slide. So we now have, in the maps … As Dan explained, we've recently added—Dan and his group at Internet Society—to the longitudinal database, tracking of those ccTLDs—in fact, all of the TLDs—that implement automated polling from the registry level. And you're going to here next, I think, from Brian Dickson about what GoDaddy is doing at the registrar level to do something comparable.

Next slide. And this is just an expansion of that particular configuration in which the registrar is not providing the DNS service for certain zones. They brought it for some but not for others. And if you have a separate DNS provider that is assigning the zone and then publishes, within the zone, CDS or CDNSKEY records and then a polling process that pulls those out and gets those up to the registry …

Next slide, please. Here is one of the forbidden maps that has country boundaries. The point of showing it here is that this is one of the more recent maps that shows the implementation of this automated DS provisioning for a few countries in Europe.

Next slide, please. Progress is moving forward. GoDaddy is now testing, as you're going to hear. And the Security, Stability, and Advisory Committee, of which this whole workshop is part of, is exploring recommendations in the usual formal way that SSAC operates to highlight the need for DS automation and give advice to the various parties. You'll hear a little bit more about that shortly.

Questions that will arise or are arising is, does scanning for these records work well or is too time-consuming? Does it scale properly? We'll see as we go forward.

Next slide. Here's a little bit of the status. The DS update process can be broken into two parts. One if the update process itself based upon the scanning. And the other is, how do you initiate this process? How do you build the chain of trust? Design is done. Specifications are written. The specification on the bootstrapping side is still in Internet draft form. And there are implementations in progress.

Next slide. All right. The other half of what we're working on is how to have multiple DNS providers sign the same zone on behalf of a common customer, and how do you coordinate all that?

Originally, this problem came up under the focus of, how do you move a signed zone from one provider to another and do so in a way that doesn't lose resolution and doesn't lose validation? It then became clear—and I have to credit Shumon and his colleagues—with observing that that's really a subcase of the more general problem of, how do you have multiple providers involved? You may want to do that because you're transitioning, as I said, or you may want to do that because you want to have the service continue with multiple providers. And so the transfer case is just a limiting case of having multiple providers.

Next slide. There is a very, very solid project underway, spearheaded by the Swedish Internet Foundation but involving several other parties listed here. And you're going to hear from … You've already heard from Eric Osterweil at George Mason University, and you've heard from Johan Stenstam at the Swedish Internet Foundation. And you're going to hear from Peter Thomassen and deSEC. And Shumon is at Salesforce, and I'm at Shinkuro. So a regular project is going on and software is being built that is going to demonstrate and spur this whole process forward so that it can become part of the standard operating environment.

Next slide. We'll just go very quickly through these. So here's some diagrams about the multi-signer coordination.

Next slide. And we're aiming at building operation demonstrations or an operational [meeting that] is repeatable and not just a one-shot thing but an ongoing basis. And that provides a basis for other people to adopt the software or adapt the software as necessary.

Next slide. This is another cut at where we stand. Protocol definitions, we think, are in good shape, but the proof is always in the pudding. The square boxes next to the [inaudible] as bullets are things that are in progress. And then there is some work that is not yet really started but is anticipated mainly on the observation side to be able to watch and observe that these kind of transitions and coordinations take place and do so without any sort of glitches.

Next slide, please. And with respect to that latter point, that will involve setting up and operating various testbeds.

Next slide. So here's components that are going into the multi-signer software. I think Johan Stenstam is going to talk in more depth about this.

Next slide. And here is a scorecard on the software and specifications, again, in more detail, to these. And the checkmarks under the Designed column suggests that that's in decent shape. And then we have lists of documents and more particularly organizations that are moving their software forward.

Next slide. And with respect to particular server software packages—Bind, Knot, PowerDNS—you see the level of implementation. And we have room here for others to be added. And if anybody is working on these or thinks that there is progress that is not shown, please do contact us. We will keep these up-to-date as best we can.

Next slide, please. And, similarly, for operational systems, deSEC, NS1, Neustar, and Cloudflare, you see the level of progress here. deSEC is up and running. Cloudflare has part of it up and running. And NS1 and Neustar are working on it. One hopes that, over time, as we show this slide in the future, there'll be more green and less orange.

Next slide. I'm not going to try and run through these slides except just to flip them quickly and show you, but for reference, there are various pointers. Just flip through them, please. So this is the list of the episodes. There will be a tiny URL for this episode later when we get around to figuring out how to post that and gather all of the material together.

Next slide. And then the agendas for each of the previous episodes are here. Just flip through them fairly quickly, please, Kim. There's Episode 2, 3, 4. Keep going. We were at The Hague a year ago, I guess, and then we're back. That's Episode 6 and 7. And now we're up to our current episode. And there will be tiny

URLs associated with all of these when we recover from running the meeting here.

Next slide, please. So that's the opening material. I'll move directly into the presentations by each of the people. Brian Dickson from GoDaddy is next. Brian, I'm turning it over to you.

BRIAN DICKSON: Great. Glad to be here. So this is really an incremental update over previous presentations, so we can quickly flip through the summary information that Steve already provided.

So this is focusing on Scenario 3, which is the registrar polling managed DNS from a third-party or fourth-party DNS operator of a signed zone, where GoDaddy is the registrar and is using CDS/CDNSKEY polling to turn around and update the DS records at the registry using EPP.

Next slide. And this just is saying the same thing in words. We're currently in a closed beta. We're still in development. So there's still work being done to actually make this work, but progress is being made. And I've got a few more details that show a little bit more granularity in terms of how it's being done as well as where it is in the stages.

Next slide. And, again, this is just the table of under development and testing for the status with us polling anybody who's not us as a DNS provider.

Next slide. So right now, we're focusing on making sure that, as we do the development, it's always going to be scalable, performant, and focusing on implementation methodology ensures that that scalability is entirely going to be feasible. The goal we have and that we're continuing to include and meet is we're presuming that everybody uses us the registrar, even if they're not using as the DNS provider, and that we can scan all the zones if everybody signs their zones and do that fast enough to be able to at least one poll of every zone every day, which we think is the "you have to be this tall to ride the ride." We expect to be able to expand this towards the end of summer. And the barebones component of using CDS to submit records via EPP has been tested quite a long time ago.

Next slide. And this is just giving a little bit more granularity into the implementation methodology. We're just doing this as a multi-stage funnel, which is kind of like a pipeline, except that you expect that there's going to be fewer elements being polled as you get further along based on how many zones are actually signed, whether there has been a CDS record, whether it's changed, whether it compares to the DS record, if it's the same as the current one or not, and then doing the validation of signatures, and then finally submitting a validated differentiated set of DS records for zones that are managed for zones that are managed by third parties and where GoDaddy is the registrar. And that's all looking like it's going to be very performant. So

we've got to the point of at least doing the existence polling and being able to do the CDS validation and submission. And we're just refining the stages as we go.

I think that might be the last slide. Yeah. I'm not sure if there's any questions about that.

STEVE CROCKER: Let me ask, though we do questions at the end. I know it's a bit of a burden for the audience to hold the questions, but we'll move this along quickly. So thank you, Brian.

And now we'll move on to Kim Davies.

KIM DAVIES: Thanks, Dave. Hi, everyone. So this is going to be a high-level review of how we do delegation management in the root zone for secure delegations. And then I'll talk a little bit about what we're actively working on right now, followed by future plans.

Next slide, please. So when it comes to DS record management in the root zone, the way we administer those kinds of changes is modeled on the same workflow that we use for all other kinds of root zone management changes. So that includes NS record updates, WHOIS records, and so forth. Maybe this is [easy to hear]. So the way that a TLD manages and administers their delegations and their secure delegations is they have a self-

service portal: the root zone management system or RZMS. And they can log into that portal and submit changes that way. But there are other ways that we facilitate root zone management changes, not just via the portal.

One key thing to note here is the trust model for the root zone does not depend on the TLD manager themselves submitting a change request. Anyone can actually submit a change request for the root zone, but in the course of processing a root zone change, we have a validation process that includes obtaining consent from authorizing parties at the TLD manager that will be used to verify that the request can proceed.

As part of processing root zone changes, we perform a variety of technical checks. The one most pertinent to DS record changes is we look for a matching DNSKEY at the zone apex for each DS record that is submitted to us. Part of the general model for root zone changes is it provides evidence that the change request is on behalf on someone that has possession of the TLD zone. So by having some artifact of the DS record present in the child zone, that gives us confidence that the request is on behalf of the part operating the zone.

This does present some challenges, though. Particularly we have some TLD operators that seek to bypass or skip this check. In some instances, some argue that it's a standby key. They don't desire to a DNSKEY in their zone at the time. Also, when there is a

change of registry operator vendor, sometimes there's complications in fulfilling this requirement as well. So it's not without difficulty in some instances, but in most instances, it works quite well.

We also look to validate the SOA record of the child zone via at least one of the DS records that is provided. We'll also note that we don't support the entire set of algorithms. We only support a subset of algorithms for listing in the root zone.

Next slide, please. So active work we're working on right now. We've been working for a number of years on our next-generation root zone management system. The current system we have in place is actually almost basically 20 years old now. We've had some difficulty growing it beyond what it does now. So we undertook a full rewrite of our system. And that system is due for launch this year.

Some of the process improvements that relate to this of note. … We're implementing a whole new model for authorizing root zone changes. Today, we have a public administrative contact and technical contact listed in the public WHOIS for every TLD. They also have the responsibility of cross-authorizing every change to their TLD in the root zone. We're actually separating those two areas of responsibility. So now TLD managers will be able to maintain their own list of what we're calling authorizing contacts. These are private contacts that TLD managers administer. They

can have as many or as few of them as they like. And they can also grant different authorizers with different levels of permission. So you can have an authorizing contact on behalf of the TLD. It can only administer the DS record, for example, or other contacts with different areas of responsibility. We think that providing this additional flexibility will cater for a lot more use cases—use cases, for example, where the TLD manager has outsourced some part of their area of responsibility to someone else.

Some of the other areas of active work that won't be in the initial release of this system but will be in subsequent releases … We're looking to adapt out technical check waiver process when we do the technical checks on the delegation. Currently, it's a pass/fail system. Either you pass all the tests and it goes through without being blocked, or it's a fail. And a fail will necessarily require a conversation with us to move forward. We're moving to a pass/fail/warn system. So a lot of the technical check issues we might identify will be now classified as warnings. If they're just warnings, then we expect that TLD managers will be able to self-dismiss those issues and move forward without any further interventions.

Also part of this model might be the introduction of some kind of permanent waiver if there's an existing issue that's known and it has been discussed that it can be passed for future change request.

Another area is multi-factorial authentication. This is an area of active discussion. I know that one of the outcomes of the recent SSR2 recommendations pertains to this. We also have a draft root zone study that has been conducted recently that talks about this topic as well.

Lastly, in the root zone management system, we've implemented an API that is targeted at high-volume requirements. And what I mean by "high-volume requirements" is that, when this system was built originally, there was about 300 TLDs, and each one was essentially operated by a separate organization. The operating model today is that there's now 1,500 or so TLDs, but many of those TLDs are operated by the same party. We have a few organizations that have tens if not hundreds of TLDs that they manage. And our system frankly isn't well-suited to bulk operations on TLDs. And in the dialogue we've had with those customers over the last few years, implementing an API seemed like a best way to deal with those, at least for now. So that's been an area of focus to optimize the interactions we have with some of our customers that have high-volume requirements.

Next slide, please. So future ideas that we have in the back of our mind but there's no active working happening on right now … One is I just mentioned there's a root zone update study that has been drafted. There was a public comment period very recently. We're looking forward to the final version of that study being released. And that will inform our future planning. Another is that

the technical checks themselves that we do is the result of a public consultation I think we did around 2007 or 2008. A lot has changed in the last 15 years. So we think it's beyond time for us to reevaluate that whole set of technical checks that we conduct. Some of the areas that I already talked about that we think should be reviewed is our policy towards supported algorithms and also whether to deprecate algorithms over time, whether there's a role that we have to play there. And then I mentioned the DNSKEY match already. On that, incidentally, we think we'll pick up that work towards Q4 of this year. So we expect to spin up engagement on this topic towards the end of this year.

And then, lastly—and this is the topic today—is monitoring signals from the child zone and how that might play into root zone management. As of right now, we've looked at things like CDS/CDNSKEY, but I don't think any TLD managers actually asked us to implement. And obviously, demand from customers … I see a hand up in the room. So maybe that's an ask. But without demand, obviously that's a key driver for the work that we do.

More generally, we think that there's a role for us to play in monitoring not just those potential signals but other monitoring that we might want to do as a cohesive set. And a set evolution could be one. If we, for example, poll on our suite of our technical checks more regularly, if we identify a regression. That's something that we could provide a courtesy notice to a TLD manager (that we've noticed something seems to be amiss or

something has changed in their configuration that they might want to take a look at).

So I think, as we explore the potential for monitoring when it comes to the root zone, it'll probably be part of more of a holistic package of things that we want to monitor, and then, based upon the outcome of those, notifying the TLD manager and potentially triggering change requests.

So that's just a quick high-level view of root zone management. Hopefully, that's useful. Thank you.

PRINCESS ARIANE: Steve, are you there?

Peter, you might as well just go right ahead.

PETER THOMASSEN: Okay.

STEVE CROCKER: Sorry. I was muted. I apologize. I just wanted to pick up on the point that Kim covered at the end and the question that Jacques has asked about using the same mechanism for updating a TLD registry versus updating the root zone when there's a change. And that came up in discussions, and so that was a primary reason for

reaching out to Kim and saying, “Why don’t you come and talk about what’s happening at the IANA at the root level?”

I think what is unfolding in front of everyone is an exploration of different mechanisms for doing these. And then that will provide an opportunity over time for comparing and contrasting these approaches. And think there’ll be some mutual learning in all directions. I don’t know that there’s a single right answer. And rather than try to force it or make noise about it, the right approach is to simply bring all of this out where everybody can see the different ways of doing things and the see whether it’s appropriate to have multiple solutions that fit different circumstances.

And so thank you, Kim, for describing all of that. I think that you guys obviously have been in the business for quite a long time of carefully managing the root zone and that there’s probably some important lessons to learn and pay attention to across the TLD space. Thank you.

So let’s move on. The next presentation is Peter Thomassen’s on the bootstrapping problem: how do you initiative these relationships? Peter, are you there?

PETER THOMASSEN: Yes, I am.

STEVE CROCKER: Thank you.

PETER THOMASSEN: So actually I haven't considered the bootstrapping for TLDs yet, so I don't know if IANA would want that. But for all of us on the chat, there's at least two people who have expressed interest.

Okay. I will talk about automatic authenticated DNSSEC bootstrapping, and I'll speed it up a little bit. I think we're a little bit behind schedule. So this is an update. I've talked about this at ICANN72. So I'll go over the intro quickly.

Next slide, please. So Steve already has explained that DS records need to get from the DNS provider or assigner to the registry. There is currently one reliable authenticated way where the registrant logs in TLS portal at the provider and then forwards stuff to the registrar through that party's web interface, and then EPP-over-TLS. That's authenticated, but many, many steps. And things tend to go wrong. The registrants don't know that this is even possible. And currently the other options, especially the CDS pull, is not authenticated. So it's not secure for bootstrapping—only for rollovers.

Next slide, please. That's not a good stage of things, and it's a little bit too hard for people to turn DNSSEC on. We need to change this, I think.

Next slide. The goal is to build on the CDS scanning where the registry or the registrar looks at the [time] zone and retrieves the CDS or CDSKEY information from the apex. But we would like to add authentication to that. And I'll quickly give you a heads up on how that is intended to work and then what has changed since the last time I talked about it and then what the current status of the implementation is.

Next slide, please. So we start out with the root zone, which is signed, and two TLDs, which are signed. And we are going to delegate the domain example.com and try to do DNSSEC bootstrapping with it. And then the nameserver is going to be under the .net TLD.

Next slide, please. So we have provided .net, which is the DNS provider for the exmaple.com domain. And they have NS1.provider.net. And all of this on the left-hand side already does have DNSSEC. So the NS hostname does have DNSSEC already. And that's a prerequisite for the protocol.

Next slide, please. So the customer registers example.com. It's in the .com zone but not yet securely delegated.

Next slide. Now the DNS operator puts CDS records into the child zone at the apex and co-publishes the identical thing at a subdomain of its own nameserver host name. So it would be something like example.com.ns1.provider.net. And in fact, there's an underscore labeling between … I'll show you that a few

slides later, but there's an exact copy of these customer CDS records at the nameserver hostname subdomain, and that is already signed.

Next slide, please. So the scanning party, the registry or the registrar, can look at the child and discover these CDS records—next slide—and then cross-check them against what's already signed under the nameserver host name. If that matches—next slide—it can go ahead and provision the DS records at the parent.

Next slide. So what this does is that we use an established chain of trust through the nameserver hostname to take a detour. And with that, we can compare the identically published CDS records at the apex and under the nameserver subdomain and authenticate what's there and use that immediately without any waiting times. And we can exclude the possibility of unaware attackers to mess with this, unlike the previous DS initialization of RFC 8078, which doesn't have authentication. So we're adding authentication to what's already there.

Next slide. The status of the draft is that it has been adopted by the DNSSEC Working Group in April of this year. And we wrote a blogpost about this at APNIC. So Nils—in fact, my colleague at deSEC … So if you want to have a more wordy explanation of the whole thing, you can take a look at the blogpost. Implementations are underway currently. So there is a tool we published in GitHub which is essentially a scanning tool. So it's

the parental-signed. You can give a list of delegation names, like child registrations and their NS record sets. And it'll do the CDS scanning and the authentication and spit out the DS records if all the checks succeed.

CoCCA is working on implementation for the 59 ccTLDs for which they provide the software. GoDaddy—I think Brian has talked about this—is also interested in adding this after they have finished working on the regular CDS scanning for rollovers. Chile is working on it, and some other registries and DNS operators, which aren't ready yet to say that in public, are also working on that. But I prefer not to name them right now.

Next slide, please. So what's changed since ICANN72? I mentioned that there is a child-zone-specific name subdomain under the nameserver host name, and the specific format of that has changed. So the new format is that you write _DSboot.thecustomer domain (example.com here)._scoresignal.thenameserverhostname. So that's just a new format. Everything else is conceptually the same as before. The idea here is that, by adding the prefix, we solve some ambiguity problem and some edge case that I'm not going to go into details about. And we also at the same time find that there's generalized … the signaling mechanism. For example, if you use another prefix, you could be doing some other kinds of announcements from the DNS provider, which are then authenticated and that, for example, could be used in the future for multi-signer key

exchange. I'm not saying it must be this mechanism, but at least it could be extended this way.

Next slide. So this is already the final slide. The authors of the draft considered the protocol to be rather mature. So in fact I think we resolved all the open questions at the DNSOp Working Group interim meeting a few weeks back. And this week, we are going to submit an updated draft that reflects all the issues that have been closed. The only thing that's needed, I guess, is final feedback from the working group and from everybody who's here who's interested in providing feedback, perhaps some document polishing, and then we will go ask for the working group last call and hope that everybody who's here is going to implement it.

Thanks. Next slide.

STEVE CROCKER: Thank you, Peter. Excellent. A really major contribution of identifying and filling in a substantial piece of the puzzle here.

Let's move on to the next presentation, which I think is … So we've got to flip through the backup slides here quickly. So this is a related activity recognizing the technical work that's going on. We've now spun up within SSAC a new work party leading to what we expect will be recommendations for implementers and operators and software developers to include the necessary functionality for DS automation.

Next slide. So as I said, the goal is to develop recommendations, and the intended audience all of the different parts of the ecosystems—the operators and the software developers, etc.

Next slide. So the work party is underway. We have initiated some surveys. The operator survey is still underway, being formulated, gathering the data, and we expect to have a draft report of … Well, we expect to complete this survey process within the next couple months and then, with respect to the overall effort, we're expecting to develop a draft report by the end of this calendar year and then to conclude … There'll be a comment process and revision process and so forth. And we hope to complete this work party activity roughly a year from now—maybe a little less than that.

Next slide. That was intended to be a super-short presentation just on basically the fact that the work party exists and a quick summary of the work that has taken place so far. And we will of course be reporting on this incrementally in future sessions.

So we move on to now focus on the multi-signer protocol and the implementation of all of that. Back up on slide, I think. And so let me turn this over to Johan to talk about the work that's being spearheaded within the Swedish Internet Foundation.

JOHAN STENTSAM: Thanks, Steve. So in the essence of time, let's go immediately to the next slide. So we've been working on an implementation of … Let's call it the multi-signer algorithms for dealing with having multiple signers either as a transition mechanism or in a more continuous, ongoing fashion. And we call that [DNSSEC provisioning] music for multi-signer controller. MUSIC essentially works. Of course, there's always things to polish and things to improve, but it does what it should do. And as we keep polishing the rough corners and try to sort those out, we have discovered a couple of corners that we needed to think more about and needed to work more on.

So next side, please. The first issue that we discovered was that, in the initial design of MUSIC and also in the way that we've read the documents, we basically synchronized the zone-signing keys but not the key-signing keys because we simply don't need to synchronize the key-signing keys among signers. And that worked fine until we realized that, well, guess what? The are also certain zones that are using so-called combined-signing keys. And then suddenly our logic wasn't sufficiently complete. So we needed to do something more to correctly detect and treat this case.

Next slide, please. There are obviously different alternatives here to deal with these combined-signing keys. We can either figure out per key whether it's used for signing the zone or whether it's used to generate the DS in the parent, etc. And that would allow

us to construct the so-called … Let's call it the optimal DNSKEY RRset. However, this is a bit complex.

Next slide, please. The alternative is to keep stuff simple and just stop only synchronizing zone-signing keys but rather synchronize all of the keys—basically just incorporate every key in all the DNSKEY RRsets and just construct a union. This is simple and this is actually what we do right now.

Next slide, please. So this works. However, it also leads to a slightly larger DNSKEY RRset than theoretically necessary.

Next slide, please. So the issue of the size of the DNSKEY RRset has been something that has … I won't say haunted DNSSEC for a long time, but it's certainly been a topic of discussion for many years. And in our case, obviously, what we're doing here by creating a union at present—a complete union or some sort of cast of the needed keys—leads to larger DNSKEY RRsets.

On the other hand, we do have two other things that are arguing in favor of this not being a problem, the first thing being the ongoing migration using more and more elliptic curve keys, which are significantly smaller. The second thing is that the reason why we initially devised the key-signing keys/zone-signing keys split was the deal with the, at the time, rather cumbersome interaction with the parent. So we wanted to be able to roll the zone keys more often in spite of not basically having the operational capacity of interacting with the parent every time. But what we're

doing right now with CDS and, to some extent, with CSYNC and also the multi-signer stuff itself is that we are getting close to a point where we fully automate the interaction with the parent. And if we have a fully automated interaction with the parent, we are sort of losing the need for the key-signing key/zone-signing key split. So my hope is that, in the fullness of time, when we are fully automated also across zone cuts, we will perhaps, in a larger scale, move towards combined-signing keys. And then, again, this extra size of the DNSKEY RRset becomes not an issue.

Next slide, please. So the other issue that we discovered working on this since the last meeting was that we have to ponder the digest algorithms used for the CDS records. The problem here is that, in essence, it's the parent who decides what digest algorithm that will be used for the published DS. So from that point of view, our initial approximation was that it really didn't matter what the different signers did. So the MUSIC software sort of just decides on what digest algorithms to use and makes sure that those are used across all the signers, and then the parent will pick up the CDS, and the parent will make some sort of ultimate decision on what digest algorithm to use for the published DS. There is no problem here.

However, there is a problem. And the problem is that the CDS scanners—and in our case, obviously we are primarily or initially looking at the CDS scanner that the Swedish registry uses—has additional requirements. And one of the requirements of the CDS

scanner is that it makes multiple checks from multiple points on the Internet to all the different nameservers for a zone. And it wants the CDS RRset to be consistent across all these testing points. If it's not consistent, an update will not be performed. And that means, regardless of what digest algorithm is actually being used in the parent, we must make sure that the published CDS RRsets are consistent across all signers. And if we have signers that use different algorithms, well, MUSIC will basically stop working as far as it pertains to initiating a DS update.

So what we have realized that we need to do is to not only decide on what keys to generate CDS for but we also need to look at all the signers to find all the digest algorithms that are in use, strange and old and whatever they may be—just make sure that we can create a fully consistent, complete CDS RRset across all the signers.

Next slide, please. This, however, is not yet implemented. We haven't had time. There is no difficulty here. It's just that it requires some peace and quiet and a couple of days, and then it's done. But we haven't done it yet.

Next slide, please. We actually discovered the combined-signing-key issue when we were finally implementing support for the deSEC API, which is something that we have talked about and promised to do for quite a long time. And that is now finally working, which is a good thing.

Next slide, please. And that's it. Thank you.

STEVE CROCKER: Thank you, Johan.

And now we have, I think, the last presentation from Christian Elmerot.

CHRISTIAN ELMEROT: Yeah.

STEVE CROCKER: And you're ready? That's great. So it's on experience with the multi-signer protocol and operation in Cloudflare. Take it away.

CHRISTIAN ELMEROT: Thanks, Steve.

Next slide, please. Just briefly about the DNSSEC platform we have at Cloudflare. We've been doing DNSSEC live signing at scale using the ECDSA256 keys since we started providing DNSSEC support to our customers. We provide some privacy through "minimal lies" NSEC. We do pre-signed DNSSEC with Cloudflare as a secondary provider, but this is for NSEC only. So NSEC-3 is currently not working as intended there. We do support DNSSEC live signing on secondary zones with use of a hidden primary. And throughout all this, we wanted to make enabling DNSSEC very

easy through a single API call or through the button of a click in the UI. However, enabling DNSSEC is not the same as securing the zone, which requires DS automation.

Next slide. So first I want to take about our multi-signer DNSSEC implementation just shortly and finish off with some of the work that we're doing on DS automation.

Next slide, please. The multi-signer DNSSEC we have today supports both multi-signer models 1 and 2. It's ready in beta today. The common characteristics among the various configuration mixes and matches of zones, however you put them together, means that we currently only support external ZSKs. We do not add CSKs or KSKs. It's possible to add them to the zone, though they will not be signed or published in the final DNSKEY sets for some of the reasons that Johan just went into.

Next slide, please. With model 1 and 2 with Cloudflare as a primary, adding DNSKEYs is done through our API or through the UI, whereas if Cloudflare is the secondary provider, you manage the external DNSKEYs currently through transfer from the primary. We are looking at adding the possibility of adding external DNSKEYs through APIs on these [servers] as well.

Next slide, please. So provisioning a multi-signer on a platform is currently done today through a series of steps quite easily. I've numbered here, though 1 and 2 can be done in any order. In order to have external ZSKs published within the DNSKEY sets, you first

need to set the model on the zone. And that is sort of a requirement for even adding DNSKEYs at all. We've seen, with the number of customers that we have, that they are adding DNSKEYs all over the place, including not on the apex. Enabling DNSSEC can be easily done through the rest API or UI. And after that, you need to then add the external ZSKs either through the rest API or the UI or, in the case of secondary zones, through transfer. Then follows of course the most important step of it all: to verify that everything is working as intended and that the DNSKEYs are published and signed correctly. And once you're satisfied as a zone owner that everything works as intended, update DS if needed and finally update the NS set.

Next slide, please. So the reason why we're calling this beta is that, currently, CDS/CDNSKEY management is not quite there through the API or the UI. We also want to simplify NS RRset management. And the big one is actually the next point, and that is that we have a slight discrepancy on how zone activation and retention work when we have multiple DNS providers in detection of the nameservers that are in use. Some of the beta testers with us have discovered that we need to manually currently step in and do some steps, and we need to have this sorted before we consider this production. And finally, of course, you need all the UI and API documentation support to be there for a fully-fledged solution.

Next slide, please. So looking ahead and doing a bit on DS updates—next slide, please—we will start doing CDS scanning of delegated child zones. Currently, signed zones with delegations required manual updates by the zone owner. And very soon this can be replaced with us scanning for CDSKEY/DNSKEYs on the child zone to allow for automatic management of DS records. And of course, this is required for automating multi-signer DNSSEC for child zones, which is something that we are looking to support, including secure transfer.

Next slide, please.

UNIDENTIFIED MALE: Sorry. Just to [inaudible] quickly, with all of those mobile phones that are going off, just be aware that it is only a warning testing system. It is not an emergency. Nobody needs to panic. It's just the Netherlands government doing a test of their emergency system.

CHRISTINA ELMEROT: I'm currently in Sweden, so that doesn't affect me.

And finally, I'm fully announcing support for authenticated bootstrapping of DNSSEC delegations that Peter just talked about. We do support this to encourage more use of CDS/CDNSKEYs, but of course this enables fast and secure DS provisioning, which is something that has, as mentioned, been

missing. We currently right now support this for all signed zones using standard Cloudflare nameservers. What's in production is the current draft, which Peter just signaled an update to, which we fully intend to then implement and follow as long as it's working together with our current DNSSEC architecture within Cloudflare.

And next slide, please. With that, I'm saying thanks for me. And thank for listening. Over to you, Steve.

STEVE CROCKER: Thank you very much, Christian.

So I don't know whether we're allowed to run over, but if we are, this is the time for Q&A. And if not, my sincere apologies for not having managed the time a little more tightly. Are we in a position to take any questions, Kathy?

KATHY SCHNITT: A couple? Fine.

STEVE CROCKER: Good. Now, the next question is, do we have any questions?

KATHY SCHNITT: And as of right now, we have no one in queue for questions.

STEVE CROCKER: Well, I love it when a plan comes together.

Thank you, everybody. We will back in the next meeting and rerun this panel with updates on each of these topics and any others that emerge. And let me thank all the panelists. And let me thank my partner, Shumon. There's a lot of working putting this together. And the DNSSEC and Security Workshop Program Committee has been an absolute dream to work with, both the external people and the support staff. It's just been a real pleasure. Thank you.

KATHY SCHNITT: Thank you, Steve. And just real quick, Wes has a question.

STEVE CROCKER: Okay. What is it, Wes?

WES HARDAKER: Actually, it wasn't so much a question. It was a comment. I wrote CSYNC seven years ago, and it has received almost no implementation and deployment. And now I'm seeing four presentations all talking about it. So just thank you to everybody that eventually is implementing a specification that turns out to be useful many years later.

STEVE CROCKER: "Outstanding in your field" can mean either your role recognized or you're all alone. You've now moved from the latter to the former. Great. Thank you.

KATHY SCHNITT: And thank you, everyone. This session is now closed. You may stop the recording.

[END OF TRANSCRIPTION]