ICANN71 | Virtual Policy Forum – DNSSEC and Security Workshop (1 of 2)
Monday, June 14, 2021 – 09:00 to 10:00 CEST

KATHY SCHNITT: Hello, hello, hello, and welcome to the DNSSEC and Security Workshop. My name is Kathy and I'm joined with my colleagues, Kimberly, Danielle, and Andrew. We are remote participant managers for this session.

Please note the session is being recorded and follows the ICANN Expected Standards of Behavior. During this session, questions or comments will only be read aloud if submitted with the Q&A pod. We will read them aloud during the time set by the chair or moderator of the session.

If you would like to ask your question or make your comment verbally, please raise your hand. When called upon, you will be given permission to unmute your microphone. Kindly unmute your microphone and speak at that time. And with that, I will go ahead and turn it over to Dan. Dan, go ahead.

DAN YORK: Good morning, everyone. Thank you for joining us here. Or good afternoon, good evening, wherever you may be coming in from. This is our DNSSEC and Security Virtual Workshop. We have been doing this for many years now as part of the ICANN conferences and events. And today we're going to talk all about DNSSEC and a bit about security of other kinds of protocols.

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

I'm going to begin by just saying this is a result of the work of a Program Committee. You can see the names and the people that are there who are part of this. These are the folks who meet weekly and come up with the program that you're watching today, that you're seeing, you're listening to.

We do put out a call for proposals after each session. For the next one, you are welcome to submit something to speak at the event that will be at ICANN72. So when you're done with this, if you've listened to this, if you think you have something to contribute on the topics that are here, please know that we'll be putting out a call for proposals and we definitely welcome any additional ideas, talks, things around DNS, DNS security, routing security. Pieces like these are all part of this panel that we have today. So, with many thanks to the people who are here, many of whom are online right now on whatever time zone they're in and are going to be part of today's program.

This workshop and the activities are an organized activity of the ICANN Security and Stability Advisory Committee or SSAC, with some additional assistance from the Internet Society and that's who's bringing you this today.

I'm going to talk briefly just about what we're seeing in terms of deployment of DNSSEC around the world. If you recall, of course, there are two sides to DNSSEC. There is the validation, the checking of signatures, and there is also the signing of zones and the creation of the signatures. I'm going to talk about statistics from both because they're very different.

On the validation side, the checking of signatures, is this a correctly signed DNS record? Geoff Huston's team at APNIC Labs have been running measurements for quite some time now using a method that uses ads in browsers to be able to go and get data across the globe to do this. As you can see on the chart, the measurements on checking are running about. They're pretty steady right now. We're holding it around 25% globally of all DNS queries are being validated, are being checked for signatures. And as you can see, it's been fairly stable for a while now. We'd like to see that grow, of course. But it is of course lumpy because if you look at some of the chart, some of it, if you look at the central column here, it says, "DNSSEC Validates," you can see in some places, in Oceana but also in Western Europe, if you look at that number, around 48% of all DNS queries are being validated. So there's a high deployment among the ISPs and the others that are there. Similarly, Southern Asia. You can see some of these here as the numbers go on. Western Asia at 39%, basically, 40%; North America, 37%; South America, 33%; Southern Africa. You can see the numbers here and you can see that in many parts of the world, we're seeing 40% to 50% of all queries being validated by DNSSEC. And then in other parts of the world, we're not. So it's a disparity that's happening in that space. But this is the validation side.

These slides are available for you to look at. And in those slides, you can see the URLs we're using if you want to check this area and dive into more information about your area specifically.

Now, the good news we're seeing is that on the signing side, the statistics that we're seeing continue to grow, and you can see how

nicely it's growing up there and getting up at this height that we're having. So this is great to see this growth. This is of the overall signatures and we're gathering that at the DNSSEC Tools project. They're gathering the statistics through the number of DS records that are out there. So we've climbed a good bit on the signing side, which is great to see and the pieces that are part of that.

We're also seeing a continued growth in the use of DANE records with e-mail servers. These are the e-mail domains with signed MX and DANE records. We continue to see a very nice healthy growth up toward the right in this chart and we look forward to seeing that continue to grow in the time ahead here.

Now, another technology we talk about in this workshop is routing security and how to be sure that the integrity of the routing information is accurate. DNSSEC is all about ensuring the integrity of the DNS info. Is this the correct info that was put into DNS by the people originating it? RPKI does something similar with routing records, BGP records, as far as information about routing paths, where to go. What this shows is the Route Origin Validation in terms of the percentages being observed in this case by the statistics at NIST in the United States, but they're showing the percentage that they're seeing in terms of routing with what are called the generation of ROAs which is Route Origin Authorization, which basically says, "Yes, I am the one who can start originate routes for this routing path. And it's cryptographically secured so that you can be able to check that and know that this is the route that you are supposed to be seeing."

But if you look at this nice graph, you can see that the growth was very slow and flat for you know 2014 on up through was 2018. And 2019 things started to pick up, 2020 things have picked up even more, and on into 2021. We're now up at around 29% of the routes are having some kind of origin validation along with them in terms of ROAs, so it's great to see there.

And partly you could see why, if you look at this chart which shows the growth in the signed ROAs that are being put out there by people in the different RIRs, Regional Internet Registries. You can see there the growth with RIPE NCC in the yellow line having gone very steady over a while and continuing to grow. The light blue line being ARIN in North America, who has had a very large jump in the number of signed ROAs. And you can see some of the other regions as well too. Again, this is why you're seeing on that previous slide the chart of the increased growth in the percentages of routes that you could validate.

We have been maintaining now a list of ccTLDs that have deployed DNSSEC in some way. Our current charts show this number of pieces. One of the things that you'll see here is that in the time over the past years, if you haven't looked at this, we've added a sixth stage last time and this time which is for DS automation, and we have a whole panel around that today, but basically our domains doing something to ensure that they're getting updates for DS records. The typical mechanism is to use the CDS/CDNSKEY records to be able to go in and get that information updated, basically solving the issue of key expiration and winding up with some automated way to get the DS records up into the registry.

At the current time, we're recording three ccTLDs doing this and they're all in Europe at the current time. But we're looking forward to more TLDs continuing to deploy these automation technologies so that we can get to a much more automated space where we don't have breakdowns because somebody didn't update their registry with the DS record and they changed their keys, etc. That is the one manual and challenging part right now within the DNSSEC infrastructure.

So that's a bit about what we've seen. We have a variety of resources for tools, dnssec-tools site, and also stats.dnssec-tools.

The Internet Society had a project in 2020 called Open Standards Everywhere that created a number of resources around how to go and configure various technologies, including DNSSEC.

There is dnssec-deployment.org. It has some historical information around what we've done over these years to work with this. And then again the APNIC stats, as we mentioned before.

For RPKI, again there's a number here that you can take a look at and see more information around that.

So again, thank you for being part of this session. As you can see our agenda today, I've just begun with this part here talking about our accounts and a piece around this. We're next going to go into a session around identity extensions in DNSSEC.

Daniel Migault and Jacques Latour are both going to provide presentations around how to go and use DNSSEC for identity purposes in different ways. We'll take a short break, if I have my timing right on

this, and then at 10:30 Central European Time, Steve Crocker will be here with a panel on again the provisioning automation, the piece I just said before. Can we automate the provision of the key exchange so we can be able to work with this? And you can see a number of speakers from a variety of different spaces who will be there. And then, finally, we will wrap it up with a couple of other presentations around other topics again within the DNS space here today.

So that will be it. We do have time for some Q&A built into the panels and to other places. As Kathy mentioned, we are asking people to use the Q&A pod and we will take the questions that way. So with that, I am going to stop sharing here and I will turn it back over to Russ Mundy to begin the next session.

RUSS MUNDY: Thank you, Dan. As Dan said, we've got a program we're rolling on with here. Our first presenter in this panel session is Daniel Migault and he will be talking about some new activity on support for individual identity that is making use of DNSSEC and DNSSEC-related capabilities. I think Daniel has been promoted to panelist, though I haven't actually seen it on the list. So let's just make sure Daniel can speak and get underway with his presentation.

KATHY SCHNITT: Well, he was there. Now I do not see him.

RUSS MUNDY: Oh.

KATHY SCHNITT: Let's see. He was promoted a bit ago but now I do not see them. With that, we might just have to go to Jacques.

RUSS MUNDY: Yes. Let's go ahead and make that switch then.

KATHY SCHNITT: All right. Jacques, let me put your slides up here.

JACQUES LATOUR: What happened to Daniel?

KATHY SCHNITT: I don't know. He was on and now he's gone. Jacques, can you see your slides?

JACQUES LATOUR: Not yet. I used to.

KATHY SCHNITT: Can anybody see Jacques slides?

FRED BAKER: I don't see them. No.

KATHY SCHNITT: Okay. Let me try again.

FRED BAKER: Okay. There we go, Kathy.

KATHY SCHNITT: Close screen, make sure it works. All right. How's that?

JACQUES LATOUR: Yes.

KATHY SCHNITT: Okay, beautiful. It's all yours.

JACQUES LATOUR: Thank you. Good morning, good evening, wherever you are. Today I'm going to talk about IoT Device Identity Management. I think it's a really interesting DNSSEC evolution happening here. Next slide.

I'm with CIRA. CIRA we run .ca. A couple of years ago, we started building an IoT registry, and we did it because we saw an opportunity. One of my main visions for the IoT registry is to have DNSSEC integral in the entire solution. I saw an opportunity to leverage DNSSEC to make a really cool solution out of the IoT registry, and I think we're getting close to having that vision established. I'm not going to talk

about the IoT registry today. You can go to the URL below and you'll find out information about our project. Next slide.

eSIMs. This is the key thing that we started to work on. eSIMs are secure element. They're like HSM, the base or identity for phones. They're secure element, they're GSMA compliant, and we're leveraging our entire solution on eSIMs. Next slide.

And then inside the eSIM, GSMA they recently developed a framework that on top of an eSIM, you have an IoT SAFE applet that can be managed by third parties and mobile network operators. And in there, inside the IoT SAFE applet, that's where we store IoT device identity. It's a pretty robust solution. On an eSIM, you can verify signature, you can create the public/private key pair, and you can have the eSIM manage a TLS session, for example, by computing a signature to prove that it owns the private key. So our entire solution is based on IoT SAFE eSIMs and other secure element. So next slide.

The IoT registry framework is all about IoT device identity management. The goal of the IoT registry is to manage IoT device identity on behalf of the cloud provider on top, and then to interface with mobile network operator to do the provisioning of identities on the eSIM. But the ultimate goal of all of this was to have DNSSEC at the heart of the solution. So if you did a mutual TLS connection from an IoT device to a cloud provider, that with DNSSEC DNS queries, you could validate the identity of the IoT device with a TLSA query to prove that it is with this, to prove that the IoT, the application service provider is with this, with a TLS query. And also to go in the registry,

it's got a certificate authority, and there's a bunch of keys here, a way to validate that the certificate have been signed by the proper certificate authority.

So that was the original goal of DNSSEC is to ensure that we could have the proper resource in the DNS to facilitate this entire process. In the beginning, we did. None of the things we wanted were there, except for the TLSA for the cloud provider. That's been there since 2012 or something like that. Next slide.

So what do, the IoT SAFE applet, this is the component on the eSIM that is provisioned by the IoT registry. And there's a couple of things that we do. On the eSIM, we have a service profile and the IoT SAFE applet, and we have the ability to create a public/private key pair on the eSIM from the IoT registry. So we create a new identity remotely from the IoT registry. And now we can sign that public key and write a signed CERT on the IoT device.

So we can have either a public/private key pair. We can have a signed CERT by the IoT registry. But the unique thing is we have the ability to put a unique identifier on the certificate that can be connected back to the DNS. And that's the secret sauce that we have in here is that every certificate that we provision has a unique DNS identity. Next slide.

So a little bit about TLSA, on the cloud provider side. I'm not PKI expert and the certificate expert, by all means, but I know that this all makes sense to me now. We have a cloud provider, and then TLSA is there to express the values of the certificate that is used by the cloud provider. So there's a TLSA record and there's four fields in there.

There's certificate usage, there's a selector, matching field, and the actual certificate data, either a hash of the certificate or the whole thing. So if you have a cloud provider for an IoT device and they run MQTT Service, so if you do a query for that service, the TLSA record would look something like this. You have Port 8883, on TCP, domain issue certificate, entire certificate you want to check.

So when you do an MQTT connection to the cloud, the IoT device connects to the cloud provider. They can compare the CERT they get from the cloud provider to the TLSA record here and make sure they match. Either in whole or as a hash, it depends on the compute power that the IoT device can do. So whatever is simpler.

TLSA actually works for that and its service dot the protocol. But for to express the identity of an IoT device, the TLSA record didn't work because when an identity is not a port or a service, it's the actual identity of the device. So before we used to use the CERT record but it wasn't the proper usage of this to express the identity of the IoT device. Next slide.

So client side, TLSA for client identity. There's a couple of drafts that were written in the last six months, which actually makes the whole solution now potentially functional with the DNS and DNSSEC. So it's a DANE client CERT 6 and TLSA DANE client. These are the two drafts here. The first one on top is a TLS extension to support DANE client identity. So let's start with the DANE client identity.

What DANE client identity is it's proposing to use a underscore device label in the TLC record to represent the identity of the IoT device. So, if

we look at the usage of that TLS record label, there's two ways you can have it. Well, you can use it multiple ways, but for us, for the IoT registry, we can have for every IoT device in the IoT registry that we manage, we could have a certificate [inaudible] of the unique identifier that we put for our CERT dot _device.iotregistry.ca. So that's for a very specific IoT device identity. We could have a signed CERT with the parameters 3 0 0 for that IoT device. Or if it doesn't have a signed CERT, we could just add a hash of the public key for that as TLSA record for that IoT device.

But this is pretty cool. This is actually keeping with TLSA for the device identity. We can have a record that would be a standard base. But the most important thing is, okay, it's nice to have a TLSA record, but once you do your virtual TLS connection, the TLS needs to note that it can either use the public key or the signed CERT information. You can use the name information from the signed CERT or the public key. So the extension that they're proposing in TLS is to enable the passing of that information that unique identifier to the server in the TLS connection so that the server can do a query to validate with TLS the identity of the client. So this is pretty cool. So we're going to modify our IoT registry to support this. Next slide.

And then the next part I think is cool, on the certificate side is to find the root or the subordinate key discovery. So, once you have a signed CERT, you need to find the root key to validate that CERT, and then you need to find that key with trust. So there's another Internet draft which is PKI based certificate discovery. And what it does is based on the DANE client ID or the TLSA record, you can build a URL from that

device name, where you can go and find the PEM file or the AKI, Authority Key ID. This is the structure for the domain name. So you can find the certificate to validate the root CERT or get the root CERT key to validate the signature.

So that's the proposal. So what I'm saying here is it'd be nice to have another label maybe in TLSA with root CERT or something like that based on the TLSA client maybe or you can actually do transform that to find the actual root key or the sub or intermediate CERT relevant for that DANE client ID. So that's an option that is useful. Can you go back another slide again?

So in TLSA there's the certificate usage. There's three modes of usage in here. And I was thinking for an IoT device, there should be a fourth mode. So if you look it up on the wiki or whatever, there's different certificate usage mode. What I was proposing is another certificate usage which would be type zero or something. The status would be that the certificate has been revoked. So that you wouldn't know by doing a TLS query for IoT device identity that it is revoked right now. Something happened to it, it's been compromised, and we could use that to express the manner in which the certificate is valid or not valid. That's something I need to work on. So, Kathy, next slide. You can go to the end.

So here to wrap it up, DNSSEC is getting there. I think with these two drafts, were making progress to do client identity management. But we have a whole lot of gaps. We've been working on DNSSEC for a long time but TLS sessions on client and server, there's no hooks, open

source. By default, there need to be TLSA validation. There's a whole lot of open-source software that us as a community, we should get together and fund, maybe create a nonprofit organization or something. And try to once and for all get all the pieces that could really leverage DNSSEC to have the code tested, level up, and supported by an entity, to ensure if we want to go the road ahead with TLSA to do client server full stack validation with DNSSEC that we need to invest in the development of all these technologies. I think I'm about done for time, Russ, right?

RUSS MUNDY: You have three minutes and you have a question in the Q&A pod. Can you answer that live for us?

JACQUES LATOUR: Oh, Q&A.

KATHY SCHNITT: Jacques, what is the business case for the registry here?

JACQUES LATOUR: Well, the presentation was not really about the IoT registry, but the business case for the IoT registry is IoT SAFE in the GSMA IoT SAFE standard, whatever. Part of the architecture is a middleware platform that does IoT registration. So there's a registration entity in the IoT SAFE framework, and that entity has the ability to use or be a certificate authority.

And we discovered that throughout this process that an IoT device is very similar to a domain name and that you can register an IoT device, you can activate, you can transfer. There's attribute that needs to be managed like the identity of an IoT device, and we decided to build an IoT registry solution around that. As far as we know, the vision for this is that every ccTLD operates their own IoT registry platform and in parallel to running a TLD operator framework. So mobile IoT device in the future will all have eSIMs and identity management is key to all of this. And it'd be really nice to have DNSSEC 100% aligned with the client server full stack identity management solution.

RUSS MUNDY: Okay. Thank you very much, Jacques. I think this does give a good illustration of some of the excellent uses that DNSSEC as a foundation technology facilitates and supports. So we do at the end of our session have some Q&A time yet, but I see Daniel has been able to get reconnected again. Sorry, we asked you before Daniel but I'm glad you persevered and got back so we just swapped around the order and Jacques did his first. We can go to your presentation now and I see it on the screen. Excellent, Kathy, thank you, who's running our screen. So, Daniel, over to you, please.

DANIEL MIGAULT: Okay. Thank you, Russ. So today I'm going to talk about TLS Identity Pinning extension, which is not about DNSSEC but is basically about making a TLS session more authenticated so that you can provide

your password and login to your registry or registrar in a more secure way. Next slide.

So what do we use TLS for? TLS is being used to establish a confidential channel with an authenticated peer. What we really want to make sure is that when we connect to a TLS server, we're actually really connected to that entity we think we want to be connected. In some cases, you can also authenticate the client but that doesn't change the need for a strong authentication with the TLS server. Once you have that session, you can start a communication and provide your login, your password, and/or send some sensitive information like an EPP streams. Next slide.

So how do we guarantee that you're actually connected to the entity you believe? Usually we use a certificate. So it's a certificate that asserts you that, yes, this name you try to connect is actually associated to the key, to that cryptographic key, which you're establishing an authentication.

The trust we have, the binding between the name and the key is actually deferred to the trust we have in one certificate authority. But how can we trust that certificate authority? I mean, if you take a standard browser, it has around 75 certificate authorities. Basically, anyone that get control of a domain name at some point can issue with a trusted certificate. And also certificate can be breached at some point. So this is why we should not only rely on certificate authentication but we should have something like a second factor

authentication to confirm that the certificate authentication is actually authenticating the entity you believe so. Next slide.

So identity pinning is about defining a server-side second factor authentication method. So just to start with, we had an alternate solution that consisted in pinning the certificate itself, and this is not what pinning identity is doing. But let me start to describe a little bit what is certificate pinning.

With certificate pinning, basically you had one first TLS session you trust, and then you say, "Okay, I'm going to keep that certificate in mind for the future sessions." So any time you connected to your favorite website, you're just checking using the same certificate that you stored. But wait, certificate are not long-term secrets, long-term data. So what will happen if my certificate is being re-issued? Okay. We don't do that only with a certificate. We basically keep in mind or pin the certificate authority. So that one is not supposed to change. Okay, fine. So you lose a little bit. The checks you're doing, you relax that. But what happened if you're changing your certificate authority? Well, it's a nightmare. So it's operationally too complex and we don't do that anymore. We don't do certificate pinning. So for all these reasons, we don't do those and what we really need is an identity that is independent from the certificate. How identity pinning is working? Next slide.

So your first attempt is a TLS session. So you have to trust that one. But the next sessions, you're checking that you were actually making a TLS session with exactly the same entity you did with the previous

session, which means that you start a session, you generate a secret, and then in the next TLS sessions, you're going to check that the other party is aware of that secret, and so and so you're basically checking that you were making a constant TLS session with the same identity. So I'm going to explain that in more detail in the next slide. Next slide.

So, in the initial exchange, you have a TLS client that is requesting a pinning ticket. And on both sides, they do compute a secret. So the secret is associated to that TLS session, you generate the tickets and back a ticket, and the TLS client store the secret as well as the ticket and it keeps that for a defined lifetime. So that's the first session. Next slide.

The second session, you want to make sure that you are establishing that session with the same entity as the previous one. So you take the ticket, you send a ticket to the legitimate server. This server generated proof. So usually it's decrypting the ticket and sending a proof that he knows the secret you have generated as well. The knowledge of the secret proved that you're establishing a session to the same entity as before. So he sends the proof, you verify the proof, and then you can be sure that you can trust that session to be established with the server you've requested. Next slide.

What are the advantages of this solution? First, it's integrated into TLS. It works with every protocols that are based on TLS. It doesn't rely on anything other than TLS. This is important because certificate pinning, for example, was only working with HTTP. There is actually no management to be done on the TLS client side and it's really

orthogonal to the TLS certificates, which means we are not trying to replace that. We're just defining a complementary method that provides more guarantees. And it's very good for business-to-business secure communications. Because then if you notice something is wrong, this business-to-business communication, you know how to react. Next slide.

What are the different resources we have? We have a blog that describes and explains how it works and provide a high level description and a bit what I did here. We have an RFC that describes the mechanisms and we do have a proof of concept on the Go implementation of TLS 1.3. Next side. And that's all for today.

KATHY SCHNITT: That's the last slide.

DANIEL MIGAULT: Okay, yeah. So that's all for today. I'm happy to take any comments, any suggestions, any questions.

RUSS MUNDY: So we have about 15 minutes before break, and I think this is a great chance to ask Daniel some questions about this, or Jacques, if you had some questions come up. Peter is really doing a great job of checking if URLs are working. Something for you to look into, Daniel. The first one doesn't seem to be working, it's on your slide here, but that's okay.

Now, when you were going through the design phase of this, I understand the desire to keep it within a single protocol. But I think you could probably strengthen it if there was some way to essentially associate it with underlying DNSSEC support for—

DANIEL MIGAULT: Yeah. I think that the use of DNSSEC makes the trust into certificates stronger. So it improves the certificate-based authentication, which is good. And this extension is more a complementary check that you do.

It's exactly what you do with your phone. You have a second factor. So it's improved the overall authentication. So I see those as very complementary. One is improving the already existing authentication, and the other one is adding another way to authenticate. But they are complementary, definitely.

RUSS MUNDY: Yes, I certainly agree they're complementary. I'm just looking at design because your set of slides is the first time I've had a chance to sort of look a bit at what's been included in the design. And the place where it seemed like it might be the most helpful to add strength to a certificate activity is in the very first exchange, where you could have the second independent cryptographic verified capability to make sure you got to the correct name location to begin with.

DANIEL MIGAULT: Yeah. But any TLS session, it's always better to have a stronger certificate authentication. First of all, if you cannot rely on certificate authentication, it won't work because we rely on that at least at the initial session. The identity pinning is only providing some advantage to the sessions after the initial one. You have to trust the first session. So in that sense, using TLS for the first session, yeah, definitely we need that.

The way I saw that is that when you're always connecting to the same website, for example, doing EPP or sending some information, configuration action to your registry or registrar this kind of sessions, at that point, I thought that you should be able to not rely only on the certificate authentication. But you could enable this extension so that you can share that any time you connect to that website, you're connected to the same one. So if someone is hijacking at one given time in your session and providing you a rouge certificate, then you can say, "Hey, wait. I'm not speaking to the person I used to speak." Then it raised an alarm and you take action. The good thing is that both entities will notice that. The good thing of TLS identity pinning is that the TLS client will notice that. But next time when you're doing a session—I mean, the TLS server say, "Hey, what are the secret? You've been hijacked."

RUSS MUNDY: Okay, good. Thank you, Daniel. We have a couple of questions in the Q&A pod. Kathy, could you read those please?

KATHY SCHNITT: I'd be happy to. We have one from [inaudible], "How about server-sided overhead of TLS identity pinning?"

DANIEL MIGAULT: The overhead in terms of the protocol, that's my understanding. It's an additional cryptographic. It's like an additional hash you're doing. So I would expect the overhead to be very, very low and not even noticeable in terms of load.

What I'm not sure about is how much overhead in terms of deployment on the TLS server side and how you manage that. I mean, you're using a ticket so you can basically reuse the existing infrastructure of resumption tickets you had with TLS 1.2. You reuse an existing infrastructure and there is no much more to do with that. I think it's also addressed the ticket of Ken, I hope.

JACQUES LATOUR: To Daniel for IoT device identity management, "Do you think it's adding a whole lot of resource constraints to support this or a small IoT device could support this easily?"

DANIEL MIGAULT: I am tempted to say yes. So on the client side, what we do is we had one step to the key schedule of the TLS 1.3. It doesn't add more. We only use that with ECDHE authentication. So when you're doing [certain] resumption, for example, you're doing PSK or PSK CDHE, then you don't need to generate this pinning and so on. It could also

impact in the size you need to store that ticket but I don't believe it's a huge constraint.

KATHY SCHNITT: We have a question from Steve Crocker. Steve?

STEVE CROCKER: Hi. Good presentation. Very interesting. Has there been any experience with errors? We've learned the hard way with DNSSEC that configuration errors caused a lot of problems, including service calls to the people who aren't responsible for the error but nonetheless get the call anyway. How much experience do you actually have in deployment of this protocol?

DANIEL MIGAULT: We don't have a huge experience in terms of errors and this kind of thing. This is why we envisioned that to be used maybe not for a standard user, it's not for web browsing, because the problem is, as you mentioned, when something fails, the end user don't really know what is happening and how to handle that.

Typically, the reason I was presenting it here is that the way I envision that is for more critical infrastructure, where something is happening—you have to use this mechanisms when it's more important that you don't establish the connection when something is going wrong than to establish that session. So that's the balance. I think we are in the situation with communications to registries or this

kind of things where it's better not to provide your passwords than to provide it to something that doesn't raise any alarm.

STEVE CROCKER: Thank you.

RUSS MUNDY: Daniel, I think we have another question from Ken Bernard in the Q&A pod. "What about the lifetimes of server keying materials?"

DANIEL MIGAULT: There are two things. If it's the key of the TLS server that is being used to generate the tickets, that's entirely being defined by the crypto and it can be very long. But if it's how long you have to keep the tickets—I think in the draft, we mentioned seven days. There is a balance. I think that every scenario can define something else but the ticket is not expected to be very, very long. I mean, we don't expect the tickets of a lifetime ticket of years. For example, we don't expect for a few seconds. For example, certificate pinning, because it was very sensitive to any changes in the certificates, those that implemented it had the certificate pinning for a few minutes, which doesn't help because you need a second session after the initial one. At least one. This is why we find out that maybe seven days is a good balance to have.

RUSS MUNDY: Okay. Thanks, Daniel. We have another question in the Q&A pod. Is this protocol more suitable to machine-to-machine or server-to-server communications? We'd like your comments on that, please.

DANIEL MIGAULT: Any time you have to carry some sensitive information, I think it's useful. And server to server, yeah, because you don't have that interface and you don't have this sort of controls—when you're doing machine to machine, you basically don't look at the communication anymore and you don't have anyone say that might detect something suspicious. So at that point, I think it's like a heart failure and you can send a report then or something like that. At that point, I think it's very useful to be able to say, "Hey, something is wrong." You raise an alarm and you have someone digging on what's the reason or what is going wrong. Because it's completely independent from the certificate, we really narrow down the possibilities of something getting wrong.

RUSS MUNDY: Okay. Thanks, Daniel. I want to just add a little bit to what Steve raised earlier in terms of the errors and handling of errors and so forth, what you just mentioned there. That we've learned again lessons from DNSSEC and as we've gone through deployment actions, one thing that I think would be real helpful to the community is if you could, in looking at how people are going to make use of this and the test betting and so forth, if you could actually include some quantitative testing and develop some real numbers from the real impact. Because certainly, in the earlier DNSSEC deployment activities, that was one of

the big hesitations about validation, is that this was an additional cryptographic mechanism that had to be done in a relatively constrained timeframes. Once some of the testing was done and quantitative data was available to people both in terms of impact on the machinery and the time it actually impacted for the actions that were being considered, that really was helpful for people making decisions about when and how to deploy this.

DANIEL MIGAULT: Definitely.

RUSS MUNDY: We have just one more question in the Q&A pod and it is, "What is the difference between DNSSEC and TLS and which is better?" You want to take a swing at that, Daniel?

DANIEL MIGAULT: Okay. DNSSEC is basically providing some information by a binding source. What DNSSEC is doing is that it takes some DNS information and provides some trust regarding to that DNS information. Typically, when a DNS information can be the name, associate an IP address to a name or a certificate or something like that, but this is what DNSSEC is doing. And TLS is establishing a session with an entity. So you need to validate. To establish this connection, you need DNSSEC information. DNSSEC provides you the ability to make session and to establish a TLS session with this entity.

But these are completely two different protocols. TLS is more about communicating to an entity or establishing with that entity, and DNSSEC is more about the information you're asking about the entity you're willing to establish that TLS session.

RUSS MUNDY: Excellent, Daniel. Thank you. It's nice to hear someone else's answer be similar to what I've given earlier. Well, this does end the time we have allocated for the session. I want to particularly thank Daniel and Jacques for their presentations. It was most interesting. What is the duration of the break here?

KATHY SCHNITT: Thirty-minute break.

RUSS MUNDY: Thirty-minute break. It is the same Zoom room for part two, Kathy, is that right?

KATHY SCNITT: Yes. That is correct.

RUSS MUNDY: Okay. I don't want to get in the way of people's chance to get a little bit of a break from our session here. Thank you again for these great presentations and very interesting interchanges. I hope to see

everyone back after our break. And then 28 minutes from now, we will restart with the panelist, Steve Crocker, is chairing. Thanks all.

KATHY SCHNITT: Thanks, Russ. Please stop the recording.

DANIEL MIGAULT: Thank you. Thank you very much, Kathy, Russ.

KATHY SCHNITT: Thank you, Daniel. You can stop the recording.

[END OF TRANSCRIPTION]