HYDERABAD – Tech Day (Part 3)
Saturday, November 05, 2016 – 15:15 to 16:45 IST
ICANN57 | Hyderabad, India

UNKNOWN SPEAKER: …our thoughts [inaudible] registrar, today support DNSSEC. So we have about 200 signed domains, and those were done manually, but there is no way we could do automation today, because our China partners, they don't support DNSSEC.

So, that's why we build this. In addition to not having any registrar supporting DNSSEC at [inaudible], we looked at, for the last two years, the DNS operators. So second problem is, DNS operators, sometimes they're far removed from the registry. And the DNS operator can be the registrant, the person running, owning the domain.

They can be the registrar. Or they could be a third party organization that is hosting provider, a content provider, that doesn't have a relationship direct with the registrant or a registrar.

But the challenge for the DNS operator is, when they sign a domain, they need to get a DS record, from the DNS operator to the registry, because the registry is going to take that DS record and put it in a zone and sign it. So, getting the DNS from a DNS

operator, when they have no relationship direct with the registrant, or a registrar, is virtually impossible in the world that we live in.

And expecting a large scale DNS operator to these guys that might deal with hundreds of different registrar, and it…

…so we need a solution that works more on a more global basis. So in the last couple of years, there has been talks about doing DNS automation to support this. So, CDS records were created, and basically, it's the first solution in the DNS that enables for parent child synchronization, for automation. So that a parent can synch off a child information to be up to date.

So there is two draft that were written so far. So the third party DNS operator to RRR model. So this is a draft on how two external parties can talk to a registry. And then the other draft is managing DS records from parent. So this is a draft about DS automation using SDS.

So, to do maintenance of domains. So, to do key rollover and all of that. So, basically it's the first time that a protocol allows a child to communicate to the parent through the DNS. And that's the automation that we need to leverage, to make DNSSEC more usable. And for dot CA, it's pretty much the only way we can implement the DNSSEC, us using this form of automation. So

that the registry, dot CA, we can grab CDS and do DS automation, and I'll cover that.

…a registrar could leverage the CDS to create DS record. So this picture, you have a child domain, example dot CA, in its zone, there is a CDS record. So to do this, you need to use [inaudible] dot 11, like [inaudible] talked about, or open DNSSEC, or whatever. Create a manual CDS record.

So, the child has a CDS record with a key, a DS record that they want their parent, dot CA, to put in a zone. So the idea is that you need some sort of automated provisioning for DS, so [DS SAP?] is the DS automated provisioning. And the idea is that this thing grabs the CDS and it validates that the domain is properly signed and it has got good domain hygiene. It converts the CDS into a DS. And then it generates the appropriate APP code, so the registrar can take that and submit to the registry.

So the registrar could poll the domain that they have, that they signed, and some of that to the registry using the RFC 59 10 EPP stuff. So, to create a DS, to delete the DS, and all of that. So you sign the child as a CDS, and then you grab it, you process it, and then you submit it to the parent through standard PP process.

So this [inaudible], this is for a registrar. So all of our registrar could run this piece of code. But for dot CA, I don't think it's a

good solution because every time I talk DNSSEC to them, they turn around and run away, so it's not going to work there.

So some registrar express interest in using this for their own domain, instead of writing code from their hosting business to generate DS record [inaudible]. So, there is some value for that.

So, the [inaudible] which applies to [inaudible], is for the [D SAP] and for the provisioning engine to be connected to our registry, which means, we were doing a piece of code that would grab a CDS from a signed domain, but and then, we've validated the same code. And then we generate the EPP code, and created DS directly in our registry. And then in turn, we generate a zone file, with the signed DS record, [inaudible].

So the issue with this is that the registry, we need to create an actor, an EPP actor that can write, do DNSSEC functions, so create keys, delete keys, across all the registrar, across all domains, using one account. So that's something we have to create for [inaudible]. And that's something that we need to put the right security controls to ensure, we need the right security controls so that certain domains don't get deleted keys when they're not supposed to.

So, if we have a registry lock on a domain, you can't use this stool to use DNSSEC automation. So, that's the idea around this. So, if we look at the DNSSEC, or [D SAP] provisioning

engine, there is an input and there is an output. So once a domain has been signed, like a new child is signed, and they have CDS record available, then the idea is that a DNS operator, can use a [inaudible] API interface to tell this tool, this domain is ready to be, have a DNSSEC operation to be done.

So either to bootstrap the domain to do maintenance, or to remove. So the API is the piece that the internet draft focuses on. How to, what are all of the transaction? So the concept we have is that, Joe User, somebody that has a single website, a domain, they can go to the web interface, they login, and they say, I just got this domain, and I want to sign this domain, and click go.

And the automated provisioning engine would automatically bootstrap the domain. So we have an interface for large scale DNS operator within API, and a web interface for regular users. And then an EPP code could go either to the registrar or the registry, but it's the same process there.

So, for this project, we build a prototype, it's [D SAP], and there is… You can connect there, right now, if you want. [D SAP] dot [inaudible] dot CA. And you can also download the code for the [D SAP] interface, and it's on Get Hub, I know the code is there. And there is also a good documentation.

So the other thing I note, the thing I think we had to do, is we created five domains to test with, so you can play. And so there are [D SAP] one all the way to five. And they're all in different stages of DNSSEC validation. So some are ready to be bootstrapped.

Some are misconfigured. One is for remove the secure delegation. And then [D SAP] four and five, those are key rollover. So DNSSEC four is, actually the root zone. But five is add a new DS record, and then four is remove a DS. So that's a key rollover. And that's how it would look like.

So you can [inaudible] that's [inaudible] [D SAP] five dot CN, and you can see what a DS record is. It's exactly like a DS except there is a C in front. That's C-D-S. Pretty high tech. I'm not going to try the demo, because I'm not going to try the demo. But you can all go and try it on your own, all at the same time, and crash our prototype, you're welcome to do that.

So if you go to the website, you'll log in, and then you put zero [D SAP] one, and you click on, you click on the preview bar. So, you click secure domain, that means create the initial train of trust. So, zero [D SAP] one is signed, it's a domain that's signed. It adds a CDS, it has a DNS key that corresponds to the CDS, and when you click secure, it goes, it validates that the domain is

properly provisioned. That means it's not [a lame?] delegation, name server…

And then it recursively goes to all the name servers and query, all TCP, to make sure that everything is in synch, that CDS is the same everywhere. And the output of a secure domain is that the yellow box, it's the EPP command to add a DS, and that's it. So, a user… So, technically, if I'm Joe User, I put my domain name there, I click secure domain, and then this would provision the DS record in the registry for that domain, knowing that it's reachable to TCP, it's very, it's good hygiene domain.

So remove a secure delegation, delete, so this is a domain, if you do a dig on DSAP three, you'll see that there is a null CDS record for that domain. So the reason I'm not going in detail at this session, on the DNSSEC stuff, is that tech day has a session, DNSSEC workshop. I've got another 20 minutes to talk about this in detail, more on the CDS side of this.

So if you want to remove a secure delegation for this domain, you click the domain name, remove secure delegation. In this case, in the registry, there were two DS, with different digest algorithm, so both of these, DS would get deleted from the registry. So it's simple. Put a domain name, validation, it's all good, remove the DS.

And then in this, so you'll notice that on top, there it says, zero one and the post, so the EPI is a post to the rest of the interface, this is a delete to the rest interface, you remove the interface. And here it's a put to the rest interface. And in this case, this is new, so this is a secure domain maintenance.

So, once a domain is signed, and then you want to do a key roll over, that means automatically, [inaudible] 9 11 would add a second DNS key to sign the zone. And they would publish a second CDS record. And this would grab, it would know that there is a new CDS record. And it would add in the registry the corresponding DS record for that.

So this is how we do DNS, the DS automation. Automatically, we [inaudible], create DS, delete DS. So registrants don't have to manually contact the registrar every time to add DS or delete DS, it's all automated from here.

So, give it a try. It works. Question?

Do you want me to answer before you ask the question?

ROBERT: That would be great. That would save me some time. This is Robert from [Packet?] Clearing House, PCH. As I understand it, the DNS operator will tell the registrant, registrar to pull the CDS, and if possible, push it to the registry. Is that correct?

UNKNOWN SPEAKER: The registrar or the registry, yeah. It doesn't matter because it's submitting an EPP command.

ROBERT: Okay. And is there some way in this case, in the TLD, where the registry will tell the registrars how to connect to the registry? Look use EPP on this port, or is there some way of signaling that?

UNKNOWN SPEAKER: The registrar to the registry?

ROBERT: Yeah, because each TLD is different, right?

UNKNOWN SPEAKER: But each TLD has a set of registrar they deal with. So technically, all domains are managed by one or more registrar. So they already have the EPP, if they have EPP interface.

ROBERT: Okay, so and the only one that can send to the registry in this scenario, is the registrar, the user himself cannot go with EPP and talk directly to the registry.

UNKNOWN SPEAKER: No, no, a user doesn't talk.

ROBERT: So this is only valuable for the three R model.

UNKNOWN SPEAKER: This is… No. So, this is a piece of code so dot CA, we would have this app, dot [inaudible], or dot CA, or whatever, and anybody that wants to sign a domain for dot CA would connect to that web interface, our service, and they say, I want to sign my domain, and then we would validate and create the DS in our registry.

ROBERT: Right. So…

EBERHARD LISSE: I think you can take this offline.

ROBERT: We'll take this in your other talk.

EBERHARD LISEE: Okay, thank you very much. Let's give him a big hand.

When is the DNSSEC talk?

UNKNOWN SPEAKER: It's supposed to be Wednesday, but mine is Sunday.

EBERHARD LISEE: It's tomorrow.

UNKNOWN SPEAKER: Monday, it's Monday.

EBERHARD LISEE: Monday. So, for the ones of you who don't know, on Monday there is a whole morning or something setup for DNSSEC, much more deeper topics than what we do here, usually, but focuses only on one thing. So feel free to go there.

Dave Connor was supposed to give us an overview of the SK, sorry, the KS key roll over. He has a scheduling conflict that occurred recently, so Rick Lamb has been quickly roped in to do the presentation. You've got 10 minutes.

RICK LAMB: Okay. I know you guys are sick and tired of hearing me talk, so I'll make this kind of quick. So, it was in 2010 that we first signed the root and generated a root KSK with much fanfare. It has

been six years since then, and so it's time to change that. The purpose of this talk is to just continue to beat the drums and make sure people know that this change is going to happen, because if you're not with the change, your DNSSEC will stop working, or your DNS will stop working, and people will start getting calls. All right?

So, I just said that. Yes, this is important to ISPs, and DNS resolver operators out there. I've already had conversations, of course, with people at the large operators like Google and they're aware of it. Don't need to do that.

Okay, 2010, we did generate a key back then. We're very transparent, open about it. 21 people from different parts of the world, mostly not American, that was on purpose, are involved in this process. They hold physical keys, smart cards, etc. I helped create this process, because of course, no one trusts ICANN, but this has been a process that has been working since 2010.

It has been working very well. We have key ceremonies four times a year. Two at one end, which is LAX in Los Angeles, the other one is in Culpepper, Virginia. And we had one recently. We actually have some people that were there in the room as well. Yeah, you know, much fanfare.

You guys have all seen these pictures, but you know, Dan [inaudible] was there, he's actually one of those 21 people still. Vince [inaudible], who is one of those people as well. And you know, we had close cooperation with VeriSign as well to do this, and there is some nice pictures of all of the facilities.

It's all very open. Anyone has any questions about this, absolutely come up to contact me. None of this is secret. We'll tell you… Well, we won't tell you the combinations of the safes, but we'll tell you what the PIN numbers of the smart cards are, and you know, what kind of control systems we use, all that. All that is very public.

Okay. Why are we changing it? It's working. It ain't broke, why fix it? Well, it's good cryptographic hygiene to do this. Right now, it's a 204 8 bit, RSA key. That should be good for probably another 20 years, but it depends on who you talk to. And so no one, you know, it's very hard to get a straight answer, even from the cryptographic gray beards that are out there.

I've heard six months for a 1024. Who knows? All right. But nonetheless, it's good to go through these processes. There are various discoveries that happen over time as well, that make certain algorithms not so strong, or vulnerable. The other one is good operational hygiene, and I think this is the most important thing.

If we don't ever roll the key, and as some may have suggested, you know, just leave it for our children to worry about, because it won't be our problem, you know, we will not know how to do this when we have to do this. And so, that to me, is one of the main reasons. But the last reason is we promised to do this, okay? To the public.

That we will roll this key. So, that's why we're doing it. Does anyone care? I don't know. Jeff Houston has given many presentations in this venue, and it has always been… He has some really good plots and descriptions of what percentage of the world uses DNSSEC. Well, in that, 15% of the world actually sits behind resolvers that do validation.

So those people would be affected. Of course, if you ask me, what percentage of the domain names out there have DNSSEC deployed? I'd say 3%, maybe. Okay? So total, in the world. So, maybe this is a good time to do the roll. Okay? Anyway, so you know, it's actually, there is a positive side, for the lack of complete, the deployment of DNSSEC.

So, but nonetheless, we're being very careful about this. One of the reasons we've taken so long to do this is, we want to make absolutely sure that when we roll this key, we get no problems, no issues. So, because if there is an issue with DNSSEC, you know, what's the first thing that's going to happen?

People are just going to turn it off. When I'm at home, and I run the network at home, like many of us do, my wife starts having a problem going to websites, I hear screaming and yelling, first thing I do is I go, IPv6 off, DNSSEC off. We don't want that to happen here. All right? Well, I fear my wife, sorry.

All right. So, all the documents are up for review. You know, you can look at them, please look at them. If you have any comments, please look through these things. We have fall back plans, we have various contingencies described in this as well. We just generated the new key, okay?

So it's not in the root zone yet. It's not going to be for a while, but we're doing this in steps. First we have to generate the new candidate key that we're going to use, and we did that on the 27th of October. Patrick Jones here was there, as one of the people making us work.

He actually was the internal witness, the two people that tend to guide and run the ceremonies. He was one of them. There is a picture of them all. There is a sheet of paper, just like when we first generated the key, where we printed the hash, the DS record, with various people's signatures on it. And that's not the official signature, of course. The digital signatures is what's important, and we have various ways to distribute the key, but there they are.

Important dates to remember. At some point, we will see this new key show up in the root zone. September 19th, the size of the packets are going to increase. One of our biggest concerns was, is, but some research has indicated maybe we don't have to worry so much, is that when we add the new key, the size of the DNS key are [inaudible] is going to increase, and so September 19th is going to be one of those days, and we have various monitoring systems in place to keep track of this.

October 11th is drop dead, that's when the key gets swapped. This is far off, okay? 2017, but we're going to keep beating these drums until then, because we want to make sure everyone knows, because there is going to be somebody on the edges of the internet that does not change the key.

And January 11th, after that, 2018, I'll be dead by then. We're going to actually issue a, something called a revoke packet. It's actually part of RFC 5011. That actually will say, okay, the old key is no longer valid. The packet size goes up to 1425 bytes, and may or may not be an issue.

I know you can't read this from the back, but this is the detail of everything. The fallback as well as the rollover plan, and it's… I encourage you to take a look at that closely because it tells you what exactly is going to be happening in the DNS, in the root.

How do we configure this stuff? Well, it depends on who you are. I've spoken to some that are simply saying, well, what's the new key? When it shows up, we'll validate via our own mechanisms, using maybe pulling it down via dig, pulling it down via a website, where we're going to publish it.

And then entering into our system manually. Some of the very large resolvers will work that way, 8.8.8.8, you know, four eights. 8.8.8.8. Very, very popular. That 15% figure, a very large percentage of that is thank you to Google. Three guys in Manhattan that work in this thing, I ran into. I was really impressed at how capable they are.

I'm not so worried about them. They'll do the right thing. I'm not so worried about the very large ISPs that are out there. They'll do the right thing, but you know, there will always be somebody out there that maybe has some problems. This is all going to be done also via RFC 5011, this is an automated update. We've tested this, but we want continued testing on bind, unbound, not, and Microsoft DNS resolver, which actually a lot of people use.

So, it's working for all of those. We keep in contact with those vendors to make sure this works. Here is some test sites for that. One written by Warren [inaudible], key roll systems. One written

by yours truly, those are accelerated test beds, where the key is continually rolling.

And it's very good to use this to see if your systems comply with RFC 5011, and are predicted, path that we're going to take in rolling the key. There are other testbeds that will be coming online soon, that ICANN is creating, that are running in real time, as opposed to fast time.

Please feel free to use those and contact us. That's it. Okay? This is a very technical room, so I don't have to go through much of the details about DNSSEC here. You will hear from us again. Thank you.

EBERHARD LISSE: Thank you very much. Any questions? This was basically informational only, as they say. Thank you very much for doing it on such short notice.

And that means I can give the floor to Joe Walden from [inaudible]. Let me bring you the clicker.

JOE WALDEN: Thank you. So the presentation that I'm going to go through is… Sure.

So, the discussion that I'm going to go through is what we've referred to as a registry verification framework. This is something, as the guy at VeriSign who is responsible for the product management of our domain name registries, it was like starting with what problem are we trying to solve.

So the problem here was, where I have a registry registrar model, and there are verifications required on objects that are maintained within the registry, how do I do that where I'm able to accomplish several different tasks?

So, if I look at this problem, one of the things that I want to do is I want to be able to have the verifications conducted, but I only want to pass data between the registry and registrar that's required, and there may be additional verification data that the registrar has, or that's required to conduct that verification that the registry doesn't necessarily need in order to fulfill the registry functions.

So I want to maintain that relationship between the registry and registrar. I don't want to insert an additional layer between the two entities. And where there is a requirement to conduct the verification, I want to do that one time, and I want it to be auditable. I don't want to have a registrar conduct a verification, and then have the registry conduct that same verification. That's just not very efficient.

I also have to have the ability to verify any of the objects. So you can do a verification on the domain name, on a contact, on a host, whatever the requirement is. So we've tried to build a flexible framework here that can expand to meet whatever requirements may be needed for these verifications. And then we'll talk a little bit about the verification interface, and again, it's something that we wanted to make extensible, so it's not specific to one use case.

It's extensible to multiple profiles, multiple instances where a registry may have various verification requirements within the same top level domain. So, if I try to do this graphically, and I'll try to walk through this diagram fairly quickly, but we'll have what I refer to as a profile, so that big red rectangle at the top called [foo], is a, it could be something geographic.

It could be functional. So it really is flexible based on how that set of requirements is defined. So if you just think of it… The easiest way to think of it, I think, is just like a geographic region. Right? So, if I have a region of [foo], within that profile, I've got three registrars that are shown in the blue boxes, [foo] dash R1, bar dash R1, and [foo] dash R2, those three registrars all are associated with the profile, and that's what we're going to call that red box, the profile of [foo].

Now, you can also have within the, within a profile, you can have this nested profile. So we'll have bar as something that is a, that has the requirements of [foo], but also has additional requirements. So in this case, the registrar bar one, or bar R1, is only registrar that has this requirement.

And then you can have a completely separate profile that we call [baz], and that we have two registrars in there. These are profiles that are defined by the registry in order to meet these separate verification requirements.

And the way that we implement the verifications is through an entity that we've labeled a verification service provider, or a VSP. Now the VSP actually conducts the verifications. There is an interface between the registrar and the VSP, and that is… There are proposed definitions out there now, or could be defined individually by VSPs based on their needs.

But that is a mechanism for the registrar to pass the necessary data to the VSP. And then the VSP conducts the verification, and then generates a signed code, that is then returned to the registrar for them to pass on to the registry. And then the registry also establishes a relationship with the VSP by entering in a trust anchor, so that, and assigning a code to that VSP, so that when that code is generated, it's unique, and the registry

can identify codes that are sent by different registrars, that may be using different VSPs.

So it's all auditable, that's the reason behind a lot of those. So, let me just briefly describe the verification code. So this is something that Jim [Gould?] from VeriSign has an internet draft, and you can see the name of it, simply, the verification code, and it's an EPP extension.

And again, the value that is used for that code that the VSP generates, contains two key components, one is the VSP ID, and that's a unique ID assigned by the registry, and then the verification ID, which is what the VSP generates, so that is a unique identifier that the verification service provider generates for each verification that they conduct.

And then as I said, you can conduct verifications on any of the entities that are maintained within the registry, and in this case, you can have an example where the verification is required to be done on the domain name. And then once that verification has been conducted, the VSP generates the code that is assigned, and the other piece of this that I'm going to walk through next, is this concept of the registry profile.

So we saw the different regions, so we'll see how that unfolds in the definition of the registry profile. So here is the same graphic that we had before, and I'm going to add a VSP called [foo] dash

V1. So that is a verification service provider that meets the requirements of the registry for the verifications that are required to be done within that profile of [foo].

And then, two registrars are using that VSP to be able to conduct verifications. So that's [foo] dash R1 and bar dash R1. But then as I mentioned earlier, so if the profile of bar has additional requirements, then the bar one registrar would have to go to a separate VSP, if [foo] dash V1 doesn't perform that service, they go to a separate VSP, to conduct a separate verification.

So you may have a verification that's done by one VSP on the domain, and a separate one on a contact. So again, doesn't require that, but it provides that level of flexibility. Additionally, we could have a registrar, I know it would be hard to see, but there is the registrar [foo] dash R2, also has the ability to perform the verifications themselves.

So, if they elect to do that, they would obtain an identity as a VSP, they would conduct the verifications, generate the codes, and pass those using the EPP extension to the registry. And then down in the profile for [baz], we've got two registrars down there. In this example, we have one VSP that is fulfilling that profile's requirements, and both registrars in this example are using [baz] dash V1.

Now, you could have multiple VSPs and each registrar could select their own. The registry policy could be to designate one VSP that registrars would have to use. So again, there is flexibility in the framework. So, I think one of the great features of this is that we have the ability to have multiple profiles, per registry, which you've seen, and there is a one to many relationship between those profiles and the VSP.

So I can have multiple VSPs conducting the same types of verifications, and because I have a globally unique identifier for each verification code that's being passed to the registry, I have the ability to have many, many VSPs.

You could actually have any registrar conduct, be their own VSP. So that's a great feature. We also have one to many code types for the profiles, you saw that. And then we have a many to many relationship between the profile and the registrar. So if you look at the, if you apply that framework to the example that we just went through, within the profile that we had for [foo], I've got two VSPs, those VSPs, in this example, are capable of performing the registrant type of verifications, however, those are defined by the registry.

And then I have three registrars that are within that region. So that's the first profile for [foo], and then with bar, which was contained within [foo], I've got one VSP, and that VSP is able to

perform domain type of verifications, and then there is just one registrar in the example, bar dash R1.

And then down in [baz], we had, again, one VSP, that VSP was actually performing both registrant and domain verifications, which may be completely separate in terms of what those verification requirements are. So if it's the registrant, one of them may have required a passport, one of them may required, you know, a different form of identification. But within that profile, it's consistent across the information that's passed between the registrar and the VSP.

And then down there, we had those two registrars. So again, this is intended to be a flexible framework that allows registries to be able to perform those verifications. So if I go back to the original set of problems and the constraints that I was trying to solve for, this model allows us to keep the local data, local.

So the registrar only has to pass to the registry, the fact that the verification was performed, they don't need to pass the actual data that was used to conduct the verification. There is no need to insert a verification provider between the registry and the registrar, or tack it on at the back end of the registry, so that registrations are performing those verifications with the entity that has the relationship with the end user.

And again, one verification regardless of how many times it's used, with an auditable feature so that the registry has the verification code, knows which registrar it came from, obviously, and also can trace that back to the VSP that performed the verification.

Also supports any object, as we talked about. You can do domains, I mean those are the three objects within a domain name registry. You know, domains, hosts, and contacts, and any operation. So I could set as part of the registry policy, the requirement to pass a verification code at the time of create. I could make that optional. I could include it in updates. I could use it with transfers.

So in the gold EPP extension internet draft, each of the operations within EPP are described. We're separating the verifications from the standard registry interface. So, this is very similar to what registrars, how registrars operate with registries today over EPP. Similar to authentication codes that are part of the specification for all gTLDs.

Those info codes are passed as part of a create, they're able to be updated. So think of these verification codes in a similar manner. And then I think, as you saw, that this is extensible, so that a registry can support multiple profiles and different rules. If I've got two different regions that a registry supports, could

have different rules, could be a geographic TLD with different nexus requirements.

I can establish the verification providers that can provide that verification, and also implement those policies in the way that meets the requirements of the TLD.

So, in summary, I think that when we look at the problem that we are trying to solve for, you know, we were able to create a flexible framework that meets all of those constraints, and again, I think is auditable, keeps local data local, where it belongs. It doesn't need to be passed, especially in an environment now where we have a lot of sensitivity around passing personally identifiable information, but that data is kept only where it needs to go in order to fulfill the verifications.

So, I think that's all I have on the slides, but I'll take questions if anybody has any.

EBERHARD LISSE: No, go ahead.

NEIL: Hello, this is Neil [inaudible] from Article 19 NCUC NCSG. This system… Thank you very much for the very interesting presentation, also for the work on the internet draft. This

system seems designed in order to comply with China's internet domain name measures, and this has a very high probability of directly impacting the right to privacy and the right to freedom of association of registrants.

So, how does VeriSign see this potential impact of this technology and implementation on the rights of registrants?

JOE WALDEN: So, within the scope of what we're talking about for the internet draft, I think this is, again, flexible to provide a registry with capabilities to meet whatever those requirements are. Specific to the question that you're asking about China, VeriSign did apply for a [R SEP?] the registry service evaluation process, that is on the ICANN site that is approved, that is used specifically for China.

And I guess, if I were trying to characterize that position, it's primarily that all registries have in the RAAs, well at least we do, within our RAAs, I don't want to speak for all of the new gTLD RAAs because I haven't read them all, but we have a requirement that registrars comply with local laws.

And in the case of China, registrars have to comply with local laws, and passing those verification codes is merely a

mechanism for the registrar to serve to the registry that they've complied with those registration requirements.

EBERHARD LISSE: Thank you.

UNKNOWN SPEAKER: [Inaudible]. My question is, who will provide VSP? So, in your [inaudible] right now, you think to assume that the China government [inaudible], but I cannot imagine who, another country, so other regions.

JOE WALDEN: So that's a good question. So the VSPs are, like I said, it may be provided by the registrars themselves. It may be a commercial business that someone provides, you know, just like people become [inaudible] providers, or escrow providers, so there are service providers that choose to do those types of operations.

So, I think it's really open to anybody that meets a qualification, and has the technical ability to perform the verifications that are required.

EBERHARD LISSE: Okay. Any other questions? Thank you very much.

So, now I can see, unlucky Norm Rich in the back. He can come and close the session please.

Sorry, I'm getting ahead of myself. [Inaudible] from dot KM is going to do the final presentation before we close. Sorry.

UNKNOWN SPEAKER: Thank you. [Inaudible] from [inaudible]. Sorry.

Okay, okay.

[Inaudible] from Comoros, I'm working to [inaudible] telecom, which is a registry of dot KM. I'm looking to talk to you about the dot KM and [inaudible].

[Inaudible] before how many people in this room know where is Comoros? Raise your hand. Good. Good. Comoros is in the Indian Ocean between Madagascar and the Africa continent. It is, Comoros is four islands, but we have one [inaudible] is independent of France. The population is estimated this year, it's a small population you see. 790,400, but now general census is underway.

I promise you, next meeting, I will give you exactly the population of Comoros. The internet is began in 1998, with 64 kilobyte per second, but now we have 1622 megabyte per second. Comoros is connected to easy [inaudible] cable, and

the different island are interconnected by optic fiber. Now, we are going to connect to Maori, the airline is dependent of France, to use to [inaudible] cable, [inaudible] and [inaudible]…

This last one is connected to Maori. Now to dot KM. Dot KM is our ccTLD. It is managed by [inaudible] telecom, a national company of telecommunication, where I'm working. But the dot KM, we have 2 R model. It is now open, and if you [inaudible] a domain name of dot KM, you must be registry in our chamber of commerce. The cost is 30 Euro, but yeah.

That because we are social company, a national company, and we are the registry of dot KM, we did some social [inaudible], for example, [inaudible] of PC and internet access school, Latin Education Center in rural areas, we organize a conference there in school and university, for the using and internet governance. Doing the national events, or [inaudible] telecom event, we can do and often do, to students.

These students can come in our company to [inaudible] it, to [inaudible] with our technical teams. To [advise?] it where our [inaudible] etc. Look, how this [inaudible] can do to push this [inaudible]…

It is a challenge. We are a small ccTLD, but we want to go forward, and we need the help from anywhere. We have [perspective?] automation of our registry system. We will

organize a national DNS forum to increase all of the actors. And we must ensure that dot KM is using [inaudible] in Comoros, and we have intending, we are intending to open the dot KM for two registrars, registries, and we have to facilitate internet access in roll out areas in Comoros.

We have to install a copy of root server in Comoros to facilitate the connection, but the important is to have the [inaudible] with our customers. And we have to install DNSSEC in our system. Looking to end this presentation, and I would thank you for your attention. Thank you. [Inaudible]

EBERHARD LISSE: Thank you very much. Can I offer a question by abusing the chair? How many domain names have you got at the moment? And do you register only in dot KM, or do you have second level?

UNKNOWN SPEAKER: In Comoros, we don't have a registrar for this [domain?] name, because we have just a 2R. How many domain name we have? We have small domain name, about 200, 250 domain name now. We don't have many, many domain names.

EBERHARD LISSE: And just one follow-up. How expensive, in local terms, is it, as compared to the registration of a car vehicle, for example? Don't need to know the actual amount, just to compare it.

UNKNOWN SPEAKER: I don't know exactly, but I think we have to increase the different society, to use the domain name. But now, we don't have a policy for this to use this domain name. In some society, prefer now to use the domain name, another domain name, not dot KM.

EBERHARD LISSE: How much does it cost to register a domain?

UNKNOWN SPEAKER: How much? Three Euro per year, four for a domain name, now.

EBERHARD LISSE: How much?

UNKNOWN SPEAKER: Three, 30. 30.

EBERHARD LISSE: 30 Euro. That's reasonable.

UNKNOWN SPEAKER: 30 Euro. 30 Euro, yeah.

EBERHARD LISSE: Okay. Any other questions?

Okay, then I want to thank you especially for coming. I always like small ccTLDs to come and present, and let's give him a hand.

And now, finally, I can ask Rich to give us a few pointers on what happened today.

RICH: As is tradition, Eberhard asked someone to do closing remarks for these sessions. I get the honor of doing that today, since I broke the microphones, probably the last time. First of all, I'm really, every time I come to one of these, I notice the rooms are getting bigger and bigger, and they're getting more full. That's very encouraging.

You know, talking around the hallways and stuff, a lot of people come to ICANN because of Tech Day. They see these sessions getting larger and better attended, that's awesome. Very encouraging. Also, from the presentations today, they're very diverse.

Very interesting. And I like that. So, sometimes you do Tech Day presentations as, there is a theme, maybe a bit, you know, too much on one topic, but today was very good. A lot of different topics.

On the topics being presented, very encouraging to see the efforts towards DNSSEC automation. DNSSEC has been with us for a very long time, like 15 years. Its adoption hasn't been that great, struggling a lot. That is probably due to the lack of automation. Because it is more difficult to deal with.

So that's very encouraging to see. [Inaudible] was great, I love that. That was very cool to see. Someone not use this encrypted email a lot, and all of the problems that go along with it, and how difficult it was to use sometimes, that was very encouraging.

Also to Louise for [inaudible] dissection. That's something we all have to be aware of, the internet of things is going to change the patterns of abuse that we see. I don't think we know what they're going to be yet. Something we have to keep an eye on.

So, again, I think that's something that we should spend more time investigating. And, just a thank you to Eberhard, everybody that did the presentation and the tech working group for putting this together. So thank you.

I guess we're done. Beer time.

[END OF TRANSCRIPTION]