COPENHAGEN – How It Works: Root Server Operations
Monday, March 13, 2017 – 15:15 to 16:45 CET
ICANN58 | Copenhagen, Denmark

STEVE CONTE: We're going to get started in just a second, but I wanted to invite you guys in the back to come on up to the table. This is really a session for you guys and about you guys, so we have power, we have microphones. You don't have to use either one but we do have a table, too, and please come on up and join us up front here.

All right, so we are going to go ahead and start. Throughout the whole day, I've been saying go ahead and ask questions as you think of them. I'm going to change that for this one. Throughout the session, we're going to have more representatives of various root operators coming in and joining us, too, so we're asking that you hold any questions until the end and then they can make choices on which root op or caucus maybe even we'll answer those questions. Same for online. We'll collect them during the session and then we'll mix between online and in the room.

So thank you for joining. This is tutorial on the root server system and today, we have Wes Hardaker from B Root and we

will have Daniel Migault, who's the IAB liaison to RSSAC, as well, joining us. So I'm going to pass it on to Wes. We only have the one floor mic, so as we do take questions at the end, if we have any raised hands in the back, you get to be Phil Donahue.

WES HARDAKER: That's okay. We actually hope to run this sort of as a town hall. So if you have questions and you want to ask at the end of the presentation about any of the slides, please do so. We'll be around to. There's going to be plenty of time at the end to ask questions, and I'll be happy to run a mic to you.

So as he said, I'm Wes Hardaker. I'm with the University of Southern California. And you're here for the tutorial on the root server system, which sits sort of at the height of the tree of DNS. And so I'll go over a quick overview reminding you what the Domain Name System is or DNS and what the root server system is and how it integrates with the DNS and what it looks like today and its features.

And then we'll go over an explanation of Anycast because that's a technology that the root server system relies upon greatly and that causes some confusion, so we have a good number of slides demonstrating how that works. And then we'll talk about RSSAC, which is the Root Server System Advisory Committee, and its

Advisory Committee to ICANN, and the recent activities that have been undertaken in that space.

So, first off, an overview of the Domain Name System and of the root servers. One thing to remember is that the computers on the Internet communicate entirely over IP addresses, and that includes both IPv4 addresses and IPv6 addresses. But humans are pretty bad at trying to remember numbers. We remember names much better. It sticks with us much better. So the DNS is really the identification system on the Internet that matches names that humans can remember to numbers, like IP addresses.

So the original problem was just that. That's sort of where DNS started. It was entirely a name to IP address mapping system, and it has grown over time to adopt on some other things. The nice thing is that sometimes IP addresses change. The servers for your favorite website might actually change, but that name to number mapping hides all that complexity and that change from you.

Today, IP addresses can also be shared. One box can actually contain multiple names as well as one name can contain multiple addresses in the mapping. And multiple IP addresses may serve as entry points to a single service, and which one do

you use? And so the DNS talks about the whole protocol for how to make that choice and that decision.

So the Domain Name System is really sort of an inverted tree. And when you have a resolver that is trying to answer a question for you, like if you want to go to www.example.com, the resolver has to start somewhere and it starts at the root, which is what we're talking about today as the root of the DNS. And that's top-level box.

So it starts at the root and it says, "Okay, if I'm looking up www.example.com" – I'll use one off this chart, www.cmu.edu – it asks the root, "Where is that?" And the root just says, "You know what? I don't know exactly where it is but I can tell you how to get to .edu." And then the resolver goes off and it asks .edu next and then it asks cmu next and, finally, it gets the answer it's looking for through a long chain of questions. It's actually much more complex than most people realize. This is happening underneath the system.

But there are many other types of mappings, too. Mail servers have their own mapping. So when you want to send mail to cmu, it actually is likely going to a completely different box because the resolver says, "Oh, I need to send mail. Where should mail go?" So there are different lookups of things like that.

So the domain resolution process is, as I said, quite complex, and so this chart shows all of the interactions and the queries that actually take place, and I'm not going to go in depth into it. Hopefully, you went to the "How the DNS Works" earlier. If you went to the beginnings of "How DNSSEC Works" last night, we went into it in more detail.

But basically, a user submits a question to the recursive name server in their ISP, and then that name server starts talking to the root and gets an answer back and then talks to com and gets an answer back and then talks to example.com and gets an answer back before handing it all the way back to the user. So the user only sees one query going out, but there were actually six or more packets that were traversed around the network.

And again, the root servers are just the entry point to the whole system. And the one important thing to realize, and we'll talk about this a few times today, is that caching is the art of memorizing a particular answer so that you don't have to ask again. So, for example, if the next time the user actually asks for something else instead of example.com, say, example2.com, it's actually not going to go back to the root because it knows where .com is already. So the root actually doesn't get a whole lot of traffic because of that because .com is cached by the resolving name server. So .com gets a lot more queries than the root, and we'll go into more about that in the future, as well.

So the DNS resolution process perceives the actual transaction that the users want to do. So after the user's application, say a Web browser, gets the answer, then it's actually going to start http and some of the other protocols they need to communicate. The DNS is just looking up where the right place is that you want to begin that final conversation, be it sending mail or looking at a Webpage or whatever.

So, again, the root servers really only know and the only thing they have is where to go next. They actually have no information themselves other than about themselves and about where to go talk to next. So they know where .com is and they have a list of all the servers for .com. And then they know where .net is and they have a list of all the .net servers. And it's basically just an answer saying, "I don't know." That's pretty much all the root ever says. "I don't know. Go talk to these people next."

And as I said before, caching of previous answers means that there's actually less need to query the root servers because there's not a whole lot of information in the root, .com has a lot more information in it. So most people cache, most of the root records are a day long in caching, so you actually don't need to go back and check again for a day, although a lot of software turns out queries once an hour at worst.

So there have been some modern refinements to the DNS. The DNS has not been a static technology over the years. We have thrown more things into it as the time has gone by. It's not just name to IP address mappings anymore. The latest big ones are we've added DNSSEC. I'm sure you've heard that thrown around ICANN quite a bit if you've been around for a long time, and that's adding cryptographic signatures to the DNS.

And we're going to go over that for quite a bit today because it turns out that IANA, which controls the root data, signs it. And so you can get the data from anywhere. It doesn't matter whether you get it from a root server or from anywhere else. You can verify with absolute certainty that nobody has modified it. So that's going to be a key talk point for today.

And it reduces the risk of spoofing. Nobody else can fake the answers or change the answers. And what we need to do, and it's starting to happen, but we need to get all the resolvers on the planet to actually start validating it. So even though DNSSEC is around, not every ISP has turned on validation to actually check those answers. It's not just for the root. It's for the root plus .com plus zones underneath.

Anycast, which is the bottom one, is something that has been deployed a lot in the last decade. It's the technology for scaling out the system to greater and greater use. It really means that

multiple servers are sharing a single IP addresses. We're going to go into great detail on that in a little bit, so I won't talk too much about Anycast now because we're going to hit it later.

But it does a few things. It improves latency, so you get your answers faster because of Anycast, and we'll show why in a bit. And it improves resilience, so it actually protects you against DDoS and other related attacks so that it's much harder to take out components of the DNS network.

And finally, the middle one is where we are today in terms of rolling out the latest changes that we want to do, which is privacy-related enhancements. And one of the reasons why is that when you're asking these name servers for questions, you're disclosing the whole name to them when you go talk to them most of the time. So if you go look up www.example.com, you're sending that question to everybody, even though the root never knows that answer or .com actually doesn't know where www is. So there's work underway in the IETF to create some standards to reduce your privacy footprint.

So, next, let's talk about the difference between what the root zone is versus the servers that actually serve the data. The root zone is the starting point. It's just a list of all the TLDs, the top-level domains, and their name servers. And the root servers are just responding with that data from the root zone. So they're

actually just responsible for returning the answers, but the data exists in the root zone.

And the root zone is managed by ICANN per community policy. That's actually a large part of what ICANN actually does is managing the root zone itself. And it's compiled and distributed by the root zone maintainer to all the various root server operators. We'll get into that in the future, but basically, as changes happen to the root zone, all the servers around the planet, and there's actually over 600 as we'll see in a picture in a bit, all respond with the exact same data.

So the database is just the content of the root servers. That's really all the root zone contains and references to everywhere else. So, currently, everything in the root servers, they're distributed from 13 identities from over 600 instances. And where I was talking before of how we use Anycast these days to expand the system, there's over 600 Anycast instances of physical boxes and physical machines and things responding to the address at various places around the world.

And the root servers are also purely technical in role. The only thing that they do is serve the root zone. That's it. They're a technical service on the Internet. And they're the responsibility of the root server operators, which we'll get into in greater detail in a later slide.

So the root server operators are 12 different professional engineering groups and they're focused on the reliability and the stability of the service, making sure that it runs smoothly, making sure that the entire root system is accessible for all Internet users. They do technical cooperation to make sure that they're all operating on the same service, and then there's a degree of professionalism. As I said, they're 12 different professional engineering groups that have been around for a long time, and they're very diverse in their organizations and their operations. They're diverse technically, they're diverse organizationally, and they're diverse geographically. They're international in general.

The operators themselves are not involved in policymaking. They just serve the data, they're responsible for running the root server system, but they're not responsible for policy. That's sort of what ICANN is in charge of, and we'll get into that more in a bit.

They're not involved with data modification. They don't actually change the data. That's ICANN and IANA's job. In fact, that's one of the things that DNSSEC, that I talked about a few slides ago, provides you is the ability to check that any data that you get from any root server instance of all 600 is returning the authentic data to you that has not been modified.

Operators are involved in the careful operational evolution of the service. As things change, they are responsible for rolling out those changes. So if major DNS protocol changes happen in the future, they're responsible for making sure that all of the root server instances around the world keep up with that change then.

And they're also responsible for continuing evaluations and deployments of suggested technical modifications and for making every effort to ensure that everything is absolutely stable so that the entire system runs without interference for every user around the Internet.

So a bit about the root server system today and its features. It has grown over time. The DNS was originally stood up back in 1983 and due to changes in technical demands and it's had to grow over time, so there are now 13 identifiers that actually correspond with 13 IPv4 and IPv6 addresses both. IPv6 was actually added in 2008. And scaling issues now are solved with Anycast, which is why even though there are only 13 identifiers, we actually have over 600 instances around the world. That's the current solution for dealing with what used to be the need for adding addresses. That's it.

And the root server system is sort of founded on a couple of principles, some key principles. First off, it needs to be a stable,

reliable, and resilient platform for the entire DNS. Because it's where everything starts, it has to be critically of importance that it's always there and available to everybody.

And it operates for the common good of the Internet. It doesn't play favorites. Everybody is treated equally. Everybody in the root zone is served equally around the planet. And the IANA is the source for all of the DNS root data, which is important. So IANA controls the contents of the zone and the 600 root servers around the planet all serve the exact same copy and must because, otherwise, DNSSEC would actually check, would fail and would show you a modification had occurred. So IANA is the source for all of the DNS root data.

And then there are architectural changes that need to be made, have always been a result of technical evaluation and demonstrated technical need. So if changes have to be made to the root server system as a whole, it's because somebody has found a technical reason why that change has been needed. And then the technical operation and the expectation of DNS is actually defined by the IETF itself.

Note that RSSAC 024, if you want to look into the history of the root server system, it contains significantly more detail on the entire history. So it goes into much greater detail how that change occurred from 1983 all the way out to the present.

If you squint your eyes, you can see the details of the root system and where all the IPv4 and IPv6 addresses are allocated and the various identifiers that are used to look them up. We won't go into detail about them because they're technical geeky stuff.

But the important thing that I've said all along is that they're announced by over 600 instances around the world. So the whole earth has tried to be covered in terms of availability so that no matter where you are, there should be a root server system near you that can give you low-latent answers and is highly redundant, too, in the face of attacks, and we'll get more into that later.

So, as you know, we've recently transferred to the post-IANA realm and this is now what the current root server system looks like. There's actually sort of separation here between where the root zone is maintained and operated from the data point of view to where it's distributed. So the root servers are over here on the right and then the clients are making queries and getting responses from the root service.

The TLD operators make changes, so when a TLD needs to make an address change or add new information, that happens over here. The new gTLD program inputted a lot of the changes into this function, and they went to IANA, and IANA was responsible

for maintaining the database that the entire root zone system is created out of. And then, finally, it's distributed by the root zone maintainer out to the various instances around the planet through the root server operators.

So the root server operators, as I mentioned before, they're highly diverse. There's a diversity in organizational structure and operational history. They use different components of hardware and software, trying to make sure that no one vulnerability in any particular system will affect everybody. There's a diversity of funding models and where they actually get their funds to run the service.

But they all have a shared set of best practices, which include physical system security. Everything has to be securely maintained. They all overprovision for capacity so that their bandwidth and computing requirements are significantly higher than the traffic that they see on a normal day-to-day basis. And they're all professional and trusted staff.

They work in cooperation through a number of different Internet-related meetings. That includes ICANN itself and the IETF and operational communities like RIPE and NANOG. Research communities like DNS-OARC and APNIC kind of falls in both. ARIN and AFNOG, as well. They use Internet-based collaboration tools, so they use open standards in order to

communicate between them. And they try and be as transparent as possible, and we'll get into transparency efforts a little bit later.

And they coordinate through permanent infrastructure that's used to respond in case of emergencies when there are things that affect the entire root system as a whole, and that includes phone bridges and mailing lists and secure credentials and things like that.

And there are periodic activities to support emergency response capabilities, and there are also established Internet bodies that have been around to help out the root from the beginning. That includes both RSSAC, which we'll get into in a minute, which is part of ICANN, as well as the IETF, which is where the DNS is founded and changes happen to the protocol are made. And then DNS-OARC, as I said, is a research facility that often drives some of those changes when new research studies have found that changes are needed.

So as the Internet has evolved, new requirements have been put into the DNS system, like I hinted at before. Most recently, the root zone operators have analyzed and adopted a few new extensions as time has come on. One of the most recent ones is DNSSEC. I'm sure you've all heard this week and actually the last few ICANNs that the key signing key that actually signs the root

server data today is going to be changing here in the coming year.

IPv6 is actually fairly new. It's actually only fairy recently that all of the 13 identifiers actually have both a v4 and a v6 address pair. And then IDNs, internationalized domain names, is continually being rolled out technology for how to put in non-U.S. ASCII-based identifiers into the root zone and other DNS services around the planet.

And then, of course, there's the increasing robustness, responsiveness, and resilience. So the number of root instances around the planet has been continually going up, just like the number of instances of .com or any of the major zones or even the Alexa top 500 zones, for example, all are likely using Anycast or a good percentage of them.

A couple of myths to be corrected. There's a bit of misunderstanding that happens over the years, and it's just from nonunderstanding how the root server system works. So we're going to run through a few of those to dispel those really quick.

The myth is that the root servers control where the Internet traffic goes. That is not true. The root servers tell you the address that you want to talk to. It's actually the routers that control where the Internet traffic actually goes. They're the ones that make sure that it gets to the right place.

Most DNS queries are handled by a root server. As I hinted before, most DNS queries are actually not handled by a root server. And in fact, the root servers get significantly less traffic than .com and other stuff because, as I mentioned before, caching actually alleviates the need for most resolvers to talk to the root servers on a regular basis.

Administration of the root zone and service provisioning are the same thing. No, they're not. Administration of the root zone is completely separate from the provisioning of the service. So as I've said, the root servers just serve the data. Where they get the data from, I said it all along, IANA and ICANN actually control the data itself. So administration of the data happens within IANA, and the root server system is actually provisioned technically independently of that.

The root server identities have a special meaning. Some believe that some are more special than another. And no, they're all identical. So, no matter which of the 600 instances that you go talk to, you're going to get the exact same answer. You can go try it. I dare you to go find one that's answering differently. Right? It won't happen.

There are only 13 root servers. No. As I just said, there's over 600 that answer from all around the planet. There's only 13 technical

identities. There's a technical limitation but, nowadays, Anycast allows us to hand out answers from over 600 different instances.

The root server operators conduct operations independently. No. They're very much in interaction with each other and act in a technical collaborative way. They coordinate the root service system in a sort of cloudlike operation and act as a whole.

The root server operators only receive the TLD portion of the query. And as I said on the privacy slide a little while ago, that's also not true. The root server operators receive the entire query most of the time. There are some new technologies coming out that do privacy-related enhancements. One of the ways that happens is it's expected that if the IETF actually passes some of these upcoming protocol modifications, nobody will actually ask the root for www.example.com. They'll only ask for .com. And they'll say, "Where's .com?" And so the root servers will actually stop seeing the entire queries.

So now we're going to dive into a quick explanation of Anycast, after I take a quick drink of water. All right, so I talked about Anycast all day and it's the key for why all of the DNS top-level services have scaled so well. With the growth of the Internet, you would think that something would break as we start asking more and more questions. Anycast has been one of the key technologies for why that has not been the case and why we've

been able to grow to 600 different instances around the planet for the root system.

In Unicast, the packets all go to the same destination. So you have an IP addresses and wherever you send it, it's going to go to that same box or collection of boxes in a single physical location. With Anycast, there's actually multiple instances that all serve the same data, and we'll talk in the next couple of slides about how that works.

The problem with Unicast is that DDoS attacks, distributed denial of service attacks – or any type of denial of service attack, to be honest, distributed or not – can take out that one box pretty easily or that one set of boxes because it's very easy to overwhelm a small number of machines.

With Anycast, it's much harder. The DDoS traffic is typically going to the nearest sources, nearest destinations – we'll see in a minute – and thus, DDoS attacks don't affect the whole system. They only affect a part of a system.

The other advantage of Anycast, as I have mentioned before, is that sources tend to get the data faster because there's a limitation to the speed of light. We can't change that. So if you're closer to the box that's answering you electrically, then you will get that answer faster because it doesn't have to travel as far around the planet.

As I mentioned before in Unicast, when you send out a packet and it's going to a destination, it has to follow from the source all the way to that destination. It might take a different path. There could be multiple paths to get there, but it's always ending up at the same place.

With Anycast, however, traffic takes the shortest route to the closest destination. So what's actually happening is that each of these blue dots here would actually be advertising the same address. They're all saying, "I'm serving this address. If you have traffic that's close to me, send it my way and I'll deal with it." The advantage of that is that the sources are often much closer to the destination, so the responses come quite a bit quicker. So the path is shortened and the data is delivered more quickly.

The other thing that happens is that under a DDoS attack, where an attacker might be trying to attack an address and overwhelm it, all of that traffic is also going to go the closest instance near that attacker, and it doesn't affect, say, the rest of the Internet, where all of their traffic is going to their nearby destinations that aren't overwhelmed by the attack traffic.

Let's talk about the root server system and what you can do on your networks. How can you best optimize it for your local network? First off, you want three to four nearby instances. You want to increase your peering activities and you want to

possibly host a root server instance so that you actually are getting the benefit of the slides that I just talked about, where by having one near you and by hosting one near you, you can get that data.

There is also recent technologies. RFC 7706, which is a technology to configuration of your resolvers in your network, so they increase their caching. They sort of pre-cache everything. And then they actually stop sending questions to the root at all because you have an early copy of all the data that you might need.

As I mentioned all day, if you can convince your local resolvers to turn on DNSSEC validation, then you'll know that nobody on the planet has altered that data since IANA originally wrote it. So, be it [inaudible] in the middle, be it at the root servers themselves, you'll know that any of those instances that you get information from are returning perfectly cryptographically proven unmodified data.

And then, also, you can participate and contribute to the RSSAC Caucus, which is where all the technical decisions are made about the root server system and advice is crafted to be sent to ICANN members that have asked for it. We'll get into the Caucus in a little bit.

So, first off, what is RSSAC and then what have the recent RSSAC activities been? This is the final section, and we'll take questions after this section.

So, first off, what is RSSAC? RSSAC is the Root Server System Advisory Committee and it's to advise the ICANN community and the Board on matters related to the operation, administration, security, and integrity of the Internet's root server system.

It's a very narrow scope. Their only responsibility is to advise the ICANN community and the Board related to the root server system. There's not a whole lot of extra wiggle room in there.

RSSAC is a committee that produces advice primarily to the Board, but also to other ICANN bodies, as I mentioned. And the root server operators are represented inside in RSSAC, but RSSAC does not involve itself in day-to-day operations. RSSAC has nothing to do with the technical service of actually standing up the root server system.

In the ICANN organizational tree – which if you've ever looked at it, it's quite large – but we sit there. I'll let you pull it up on your own sites. You can dive down into it in greater detail because those words are hard to read, I'm sure.

The RSSAC organization is composed of representatives of the root server operators and an alternate to those, as well as

liaisons to a bunch of other Internet-related boards, which I'll get to in a later slide.

The RSSAC Caucus is a body of volunteer subject matter experts, and the members are confirmed by RSSAC based on statements of interest. These are the two RSSAC co-chairs, Brad Verd and Tripti Sinha. Can you raise your hand, Tripti? I don't think Brad's here today, but there's Tripti.

Then the liaisons that I talked about, there's a whole bunch of liaisons that all communicate with RSSAC about questions coming to RSSAC and about things that RSSAC are doing and bringing back to the rest of the communities. That includes a representative from the IANA Functions Operator (now PTI); the Root Zone Maintainer; the Internet Architecture Board (IAB), which oversees the architecture of the Internet along with the IETF; the Security and Stability Advisory Committee; the ICANN Board itself; the ICANN Nominating Committee; the Customer Standing Committee; and then the Root Zone Evolution and Review Committee.

The RSSAC Caucus is composed of 85. It's actually, I think, 89. I thought we changed that. So, anyway, 85-plus current members. They submit a public statement of interest about why they're interested in helping determine the future of the root DNS service, and then they get public credit for their individual work.

The documents that are coming out of RSSAC are typically those produced by the Caucus and the names of the authors that contributed to it are in the bottom of that documentation.

The purpose of the Caucus is to house DNS experts who bring a diverse expertise to these publications and who all have different viewpoints and can contribute toward a common solution. And then the purpose of the caucus is also to promote transparency, so who does the work. The Caucus is a body that anybody can come contribute to if they have a public statement of interest that indicates they have the background to do the work.

And then it's a framework for really getting work done, so the RSSAC Caucus work parties are where all of the new technical evaluations happen, and I'll get to a few of the examples of that here in a minute.

Recently, the RSSAC has published four most recent documents, starting with the bottom one. RSSAC023 is the History of the Root Server System. I talked about that before. That's a document that came out of the Caucus that defines in great detail. We wanted to capture the history of the root server system that's evolved over time before that information was lost into minds that had them written down.

RSSAC024 is the Key Technical Elements of Potential Root Operators. It's sort of the minimum requirements for what it takes to actually be a participant in the root server system.

In October of 2016, the RSSAC body held a workshop and RSSAC025 is the report from that workshop and what happened there.

And then RSSAC026 is the Lexicon. We went through with great detail and described a few terms to make sure that we had common agreement. In the same way that the myth slides earlier indicated there's some misconception that happens over time, sometimes certain terminology are used inconsistently around not just ICANN, but around all of the Internet bodies. So we produced a lexicon of the most common terms and make sure that there's a common definition behind them all.

Please do come and attend the RSSAC public meeting to hear the additional details of these. We'll actually dive into a greater detail in the public meeting. Tripti, can you remind me when it is? This time tomorrow in which room? Okay. We'll get that, but please do come to the public RSSAC meeting where we'll talk in greater about it.

STEVE CONTE: It says 3:15, Hall A3.

WES HARDAKER: Okay, so 3:15 at Hall A3.

STEVE CONTE: Yes.

WES HARDAKER: Which is around the corner that way.

STEVE CONTE: The other side. This is the B section.

WES HARDAKER: Okay. If you want to hear more about the administrative and policy side of RSSAC, that's the place to go.

Currently, we have two outstanding work parties that are doing work right now. One is on the Technical Analysis of the Naming Scheme Used for Individual Root Servers. One of the questions that has been outstanding lately is a time to change the identifiers either to house them directly in the root. Right now, they're in a separate subzone. There's a whole lot of technical details about why we may or may not want to do that, and a document is just being wrapped up on that subject.

Also, work has been recently started on the Best Practices for the Distribution of Anycast Instances of the Root Name Service, so all of those little dots around the map. One of the things we want to study is what's the most appropriate and ideal placement. What's the best way that we can deploy new nodes to make sure that everybody is covered really well by Anycast instances?

I mentioned before that one of the goals of both the root server operators as well as RSSAC is to be as transparent as possible. So we've undertaken that goal in hand and tried to do a lot of things to make sure that we are increasing our transparency as time goes on. Recently on the RSSAC, we've established the Caucus that I've talked about. All of our minutes and workshop reports are publicly published. The RSSAC and Caucus calendar is publicly available.

The RSSAC has public meetings every single ICANN where we talk about the work that we're doing as well as take questions and answers. There are meetings with other ICANN community groups. Lots of times, RSSAC is constantly meeting with other community groups within ICANN that are needing to make decisions related to root server systems policy.

We hold tutorials like this one. We're involved in some of the other tutorials, too, including How DNS Works and DNSSEC and

the rolling of the key signing key that's coming up soon. And then we have liaison relationships and our operational procedures, and how RSSAC actually operates and how it's tied ICANN is in RSSAC000.

Finally, the root server operators also publish minutes from any meetings that they hold. RSSAC 002 is a document that defines a set of statistics that every root server operator should be publishing about their system. All of those are publicly available, and you can go look at them and do your own analysis of the root server system based on that information.

They all participate in RSSAC. They have a representative that sits on the RSSAC body. There's a public Webpage, www.rootservers.org, has a whole lot of information, including that map that I was showing earlier with the 600 instances. And you can zoom in and find out exactly where they are. There's individual Webpages, which are all linked off of that Webpage, as well, so you can go find information about each instance.

And then there's public letters with some of them with IANA about their agreement about operating and serving just the IANA root. And then collaborative reports on major events, so when there's a major event that affects the whole service, the whole system as a whole, they write up sort of a triage report afterwards and what happened.

And then RSSAC can also respond to technical questions about the root server system. There's, I think, a link on the RSSAC page and there will be soon on the rootservers.org page about how to send a question about the whole technical service to RSSAC and the RSSAC chairs will take care of handling that and making sure that everybody gets an answer.

All right. Finally, here's some more information on RSSAC, but we will go on to questions next for anybody that has any. That's the RSSAC main Webpage, if you want to go look up more information, including all the documentation, and it's the publications are on the second one, clearly linked from the first, as well.

There's more information on the RSSAC Caucus on that bottom Webpage and how to join it. And actually, you join it by sending a request to RSSAC-membership@icann.org. There's a form that you'll have to fill out, but we encourage everybody to participate in the Caucus because we want the widest body of experts that we can get. And that includes both technical and policy sides and for whatever work might need to be done. We do work in both of those spaces in the Caucus.

That it is. So, with that, there's a lot of experts around the room, so I'm going to take questions but I'm going to try and defer the

questions so somebody else can talk because I've been talking for an hour straight. Does anybody have questions? Please.

SAURABH DUBEY: Hello. I'm Saurabh, first-time Fellow from India. My question is this. Few days ago, I did a test that from root zone file, I removed the entries off 10 root servers. I put the entries of only three root server [which] we host [as a Nixi]. Then what my finding is this, the query resolution which took earlier three millisecond/two millisecond, it resolved within one millisecond. So why we are putting the entries of 13 root servers? Suppose any country having four to five mirror instances in that country, so we can't put the entries of only five. Why we are putting 13?

If there is a chance of redundancy that for redundancy purpose we are putting the entries of 13, then why we cannot increase the number of technical incidents that 13 to 15, 16, and more?

WES HARDAKER: Does somebody want to answer that? Lars?

LARS-JOHAN LIMAN: Hello. I'm Lars Liman. I work for Netnod, who operates one of the root name service installations. There are several reasons for having 13. To begin with, it could be that the three operated by

you are mishandled somewhere. Someone fiddles with the content, then there will still be 10 outside of India who work. Sorry for the nefarious here, but the idea is to have a wide cloud and the client should be able to select any one they like.

Now, in addition, the set of root name servers is – I should back up. The resolver on the client side is normally configured to use the hints file, what you have configured, only during the very first query and it then queries for a complete list of root name servers. And when it has the complete list, it normally walks through them all, one by one.

As it needs new information, it doesn't do it a specific way but, "Okay, yeah, I need to talk to root name server." So it talks to one and it will go through all the 13 addresses, and it will measure the time it takes to talk to every root server. And when it's done with all these 13 identities, it looks at the list. "Oh, that was the nearest one. That was the quickest one. I'll use that one."

So you don't have to do that. You will just put a lot of energy into doing something that the client already does automatically. And if all the three ones that you have disappear, it still has 10 on the list to say, "Okay, another one in" – what do I know – "Colombo in Sri Lanka is the nearest one and it was on the list."

It keeps and maintains that list, so I see no reason, really, to cut down on that list. And even though it may not be the detailed case right now, we're very close to looking at signing the list of root name servers. And as soon as we do that, the signatures will not work anymore if you try to modify the list. So I strongly encourage you to leave the list alone. Thanks.

WES HARDAKER: Thank you. Anybody else have a question?

ANEERAV SUKHOO: I'm Aneerav Sukhoo, first-time Fellow. My question is about the, it was mentioned that the queries can leak information. Can you give an example of a query that can leak information?

WES HARDAKER: Warren?

WARREN KUMARI: Yeah, sure. So, for example, if a user is looking up www.google.com, there is no reason for the root server to know that the user is looking up www.google.com. All the root server needs to know is that the user is trying to find .com. So basically, currently, implementations send more information than is actually needed to the root servers.

There's currently some work in the IETF, which I think is an RFC now, which is being implemented still, so that when a resolver asks the root, it will only ask what the TLD at each level, it will include the amount of information that's actually needed.

The information that's being leaked is simply what the user is looking up, and that only happens once every many number of queries because most of the time, the .com information or .in or whatever the TLD information will already be in the cache. So one every many, many thousand queries, it will have expired from the cache, the resolver will ask the root and will include a little bit more information than it actually needed to include. Not sure if that was clear.

WES HARDAKER: No. Hopefully, it was clear. Does that make sense and answer your question? So, if you want to go look up the documentation, whether it's published or not, the magic keywords are "query minimization." That's in the title of the document that defines actually why you want to do this, as well.

The reason it was never architected that way from the beginning was that you never actually know whether one server is actually able to give you a deeper answer because it happened to be serving multiple levels, and so by asking only one additional level each time, you may actually have to ask that same server

two or three questions. So there's sort of a downside to it but, typically, the privacy sides are worth the downsides because round trip times are very small. Yes, sir.

ADEEL SADIQ: Hi. Adeel from Pakistan. I'm a first-time Fellow. My question is regarding caching. So, for example, if I open a website www.mail.google.com. So, how does the caching work in that [set? It will store] whole address in a cache or .com in a separate cache, mail a separate, or Google separate? How does it work?

WES HARDAKER: Excellent question. Somebody else want to answer? I've already got the mic. Okay. To answer your question. All of them.

What happens is that every record that you look up, regardless of whether, you'll get an answer back for .com, let's say. And you'll get a list of all of the servers for .com, of which there are quite a few. There's I don't remember how many .com servers, like 10, I think, and their addresses.

They all come with a time-to-live value associated with them. So, it'll say, "You can keep this information for up to (let's say) a day." Okay? So the .com records from the root, I believe, are a day. And then the next thing you're going to ask is, "Google." And Google will say, "Oh, well, you can keep these records. Here

it is. The answer is for Google.com. You can keep these for five minutes." And then you'll ask Google, "Okay, but what about mail.google.com?" And they'll give you a different answer, "Well, here's the answers and you can keep these for one hour."

So each of those records will actually have a different expiration time on how long you keep them in a cache, and it's up to the servers in question that dictate, I should say, the data owners for that zone, that dictate how long to keep stuff. And there's different reasons. Some people like really short ones. I actually did a recent analysis that showed that the average is like 20 minutes or less. If you own a zone, I suggest you push it up for 20 minutes. It's actually sort of a bad value, but do the proper engineering work to figure that out.

Very good question. Thank you very much, and welcome to ICANN.

STEVE CONTE: Sorry, Wes. Before that, we have a gentleman right across from you here had a question. Okay.

AFIFA ABBAS: Hello, everyone. I'm Afifa from Bangladesh. I'm a first-time fellow in ICANN. I have always wondered about one thing that why there are only 13 root servers, why not 14 and 15? But from

today's presentation, I can say that there are 600. Right? So that answered my question.

My second question is I don't know too much about DNS but as per my knowledge, all the root servers are identical. My question is if there is any change in any root server, how do you manage to propagate it on 600 instances and the other 12 root servers?

And the third question is, can anyone host the instance of the root server? I mean, me being from a general user in Bangladesh, if I want to host one instance of root server, can I do that and what is the procedure?

WES HARDAKER: Excellent question. Does anybody else want to answer? Lars. Thank you.

LARS-JOHAN LIMAN: Do you remember the question? How can we manage the – ah, right. So, the data – there are two types of changes to the root servers. You're quite right that all the instances here, regardless of which operator who operates them, they contain exactly the same data. There is this myth that there are mirrors or copies or something that should have some kind of lower class. That's not the case. They are all identical and they have identical data.

And there are two types of changes that can happen. One is a normal update of DNS records. That happens every day, and that's the normal procedure on how DNS is updated. So things will be – it goes through this entire thing here. The IANA receives a request for change. It kind of validates and checks that everything is okay, asks the root zone maintainer to put it in the DNS system, and then there is actual automatic synchronization methods for the DNS server programs itself.

The servers here, which are operated by the root zone maintainer, will send out something called a notify message, telling all the root server operators throughout the servers at the root server operators that there is new data to fetch. And these servers, which are typically normally a few servers per root server operator, they are not the root servers that you see on the network, but these are kind of intermediate servers that help distribute the data. So, we call them distribution systems.

They will receive notify messages that there is new data to fetch from root zone maintainer that will go there, pick up the data, and then in turn, they will repeat this thing, talking to its own root server instances. It will tell all of them there is new data to fetch here. So, for Netnod, all the Netnod instances will receive a message from Netnod distribution systems saying there's new data to fetch and they will go and fetch that new data.

Normally, when I talk about new data in the root zone, it's only a small number of DNS records that have been changed. The root zone is actually a very small database. We're talking about a few thousand records. That's not much. It's very small. And an update here usually pertains to only a small number of records.

So what's sent across when they actually come to fetch the data here, what's sent across is only the difference between the old database and the new database so that it will send the information will contain at least remove these records, please add these new records and that is often a very small message.

So, for Netnod, the time it takes for the Netnod system from the time we receive this update information, the notify message here, until we have fetched sent the notify and all the instances have fetched from the Netnod distribution system, that timespan is roughly seven seconds. So keeping the [data in sync] is not much of a problem under normal operations. And the world is not going to fall over if that delayed for a bit.

The other type of change that can happen, happens much, much more seldom, and that's when we have to reconfigure the system to behave in a different way. For instance, we need to add IPv6 or we need to add DNSSEC or some of these major changes. Or we need to update the software because someone found a bug somewhere.

In that case, we have to walk through all the servers here and one by one, we have to execute the instructions to change the system. It can happen that that is out of sync a bit, but it very seldom has any major impact on the queries. If we fix a bug before, it will answer DNS queries. After, it will answer DNS queries. So there is no big change there.

So it can happen that we have the systems configuration bias out of sync a bit, but no Internet user with less than several years of DNS experience will notice the difference, and typically, any client will not see any difference at all.

There was one more question. You're not in line. You have to wait. How do you go about to host a server? That varies by operator. All the root server operators have different approach how to roll out more instances, and the way is to talk to several of them and ask if it's possible if there's any way to make that happen.

We at Netnod are quite open to suggestions to put the root servers in virtually any place, but it is a balance for us because we don't want to deploy server if there's not a lot of use to the user community in that region. If there's a special need someone really thinks they want to have one in place where we don't think it's very important, then maybe we can come to a financial

agreement that if someone pays for the hardware and so on, we might be able to find a solution.

For the other operators, there are other requirements and other kind of ways to do it. But if you want to reach out to all the root server operators, come and talk to RSSAC. There is a mail address, and now I look at Tripti, ask-rssac@icann.org. That will reach RSSAC and RSSAC is willing to find out, to send that out to all the root server operators and with some politeness involved, we will definitely respond to that for you. Thanks.

WES HARDAKER: And I'll put out a reminder, as well, that one of the work items that I said that the Caucus is currently working on is evaluating the right way to deploy Anycast instances in the near future to best get coverage everywhere. All right. You had a question next.

[DAN]: Yeah. My name is [Dan]. I'm from the U.S. Also, first time at ICANN.

WES HARDAKER: Welcome to both of you, actually.

[DAN]: Yeah. I wanted to return to the question of query minimization and privacy, just ask for a little bit more clarification. Do the root servers keep any data on the ISPs or the users that query? And then, two, could you give an example of either a leak that had happened or the potential for a privacy violation? I'm not really seeing how the end user could be affected.

WES HARDAKER: No. It's a very good question. So, what happens is if, let's say, you had a Web browser and you put in www.cmu.edu. And let's say nobody else in the last 24 hours had asked for anything under CMU and under .edu. Then your local resolver would go, "Wow, I know nothing." Right? "Nothing is in my cache. It has all been expired. I've got to go ask the root first." And the question that it sends to the root is, "Can you give me the address record for www.cmu.edu?" And as I said before, the root never knows. Right? That's way too detailed. The root just says, "I know where .edu is. I'm going to point you there."

So the whole issue of privacy is not related to the root. It's related to the DNS as a whole. So the reality is that every time you send a query anywhere, you're often sending the full question until query minimization actually gets implemented and deployed.

So that includes the root, includes .edu, includes CMU. Every step along the way, you're sending a bigger question than you really need the answer to. But the reality is, is that .edu is usually in everybody's resolver. It's a fairly popular TLD so that queries don't get to the root very much because most of the time, the resolvers will start here, because it already knows where .edu is. Why would it go back to the root for that?

As to which, say, all of these boxes, each one is handling traffic, which ones log their traffic and save it and which ones don't? I couldn't tell you. Everybody does different things. I'm sure that there are some countries, some military organizations, some commercial organizations that do keep that traffic for a long time because it doesn't matter what they want to do with it, but the reality is that a lot of people see that as a privacy invasion. So it would be better to not give them that, leak that information.

As to can I tell you a specific example of when that actually caused somebody an issue? I don't know of any news report that's ever happened. A lot of this has come from, I guess, the WikiLeaks posting about the [cowbell] paper was one that you could go look up and that's where government agencies seemed to be looking at data that was DNS-related, so that has some relationship and that's where a lot of the query minimization

work sort of started, as well. One way we can battle this is just stop sending the question everywhere.

So okay. Any other questions?

ABDERRAHMAN AIT ALI: Hi, everybody. My name is Abderrahman. I'm NextGen, so this is my first ICANN.

WES HARDAKER: Welcome.

ABDERRAHMAN AIT ALI: Thank you. So, my question is, well, I'm working on a blockchain technology and while I'm here, I see a lot of similarities between the blockchain technology characteristics or architecture and some of the characteristics in the root server system. Is there some kind of vision or idea that there is now evolving to use blockchain technology for root servers or DNS system in general?

WES HARDAKER: That's a really good question. Right now, the DNS system as housed on the normal DNS naming infrastructure has no plans for changing the underlying technology. There are other research that has been done, especially with blockchains.

There's something called a Namecoin you could go look up if you're unfamiliar with it that does blockchain and naming, but it's an alternate naming space.

One thing that's not well-understood is the DNS is actually one naming system. It's the most popular. It's the one that's known by most of the Internet, but there are some others. The Tor project, for example, houses stuff under .onion. That's sort of an alternate TLD that sort of pseudo exists. Namecoin actually has their own alternate space. Corporations employ their own names. They stand up fake names inside of a company that don't exist anywhere else on the Internet.

So there are other forms and the hardest part comes when you put a name into a browser. Which one do you get if there's two answers? And so that's something that each operating system can sort of pick to prioritize between them. Microsoft has their own naming architecture for their operating systems, for example, so you have to watch out for conflicts.

UNIDENTIFIED MALE: Wes, just to add to that if I could.

WES HARDAKER: Please.

UNIDENTIFIED MALE: There's a session tomorrow on emerging identifier technology and Wes mentioned Namecoin. You did, as well. We're going to have a gentleman, Jeremy Rand from Namecoin, there. We'll also have someone, Christophe Blanchi from DONA Foundation, and somebody from Frogans Technology. So that's tomorrow at 11:00 a.m. Please look it up on your schedule. If that's where your interest lies, that might be something that might be worthwhile to attend.

WES HARDAKER: Yeah. Just realize that there are parallel naming systems, so they don't necessarily operate in conjunction with. They're a completely independent naming tree most of the time. All right. Any other questions? Do you have another one?

ADEEL SADIQ: You mentioned that Anycast technology will use the shortest path to reach to the nearest. So what is the shortest? It is in terms of hop count or bandwidth or delay. What's the "shortest" terminology that you're using here?

WES HARDAKER: Right. It's the shortest hop count. Terry?

TERRY MANDERSON: Hi. I'm responsible for L Root, one of the root name servers. The Anycast technology is based on BGP, so it uses, essentially, path length, so the number of ASNs within the path. There are some other interesting aspects of Anycast via BGP, and I'm not going to describe all of the fun little tweaks you can do with BGP here and now. But, essentially, it's about the closeness of the ASN path, so the shortest ASN path between your network and the root's name server.

WES HARDAKER: Okay. So, we have time for a couple more questions. Go ahead.

UNIDENTIFIED FEMALE: Just a quick follow-up for that question. When you mentioned that this isn't about the S path and BGP, so that means if an operator, they have their own traffic engineering policy with the S path will affect the DNS resolution for the users. Right?

TERRY MANDERSON: It won't affect the DNS resolution for the users.

UNIDENTIFIED FEMALE: I mean time.

TERRY MANDERSON: Time-wise, it may. It may. It may make it faster.

UNIDENTIFIED FEMALE: I mean, faster is without doing any traffic engineering is the fastest one, right?

TERRY MANDERSON: Not necessarily. BGP is a complex little beast, and how networks are put together is quite a complex architecture. And when you have multiple networks peering with multiple other networks, and then you overlay the policy constructs of BGP on top of that, it can become either faster or slower. And you also have to remember that BGP policy is driven by businesses and business need. In some cases, what they're doing is they're changing their BGP routing policy and their traffic engineering for cost, not necessarily speed. In some cases, they are doing it for speed. It's an "it depends" answer and so there's no one true answer there, I'm sorry.

UNIDENTIFIED FEMALE: Thank you.

WES HARDAKER: All right. Any other questions? Anyone from the back who doesn't have access to a mic? We've got allocated 10 more minutes for this session, so we don't need to use it, but this is a great chance to be with the root server operators. And so if there are questions in the back or any additional ones here, let's take advantage of it.

SATISH BABU: Hi. My name is Satish and I'm the Chair of the APRALO. I have a question on the transition to IPv6. Any of these things in the landscape of DNS, does anything change if you could transition completely into IPv6? In particular, the number 13, is it at all influenced by these IPv4 to IPv6 transition? Thank you.

WES HARDAKER: Duane?

DUANE WESSELS: If I understand your question correctly, you're talking about situation where we're only using IPv6 addresses and no IPv4. The reason that we have the 13 is because of the response size. Right? And with the inclusion of IPv6 addresses, that makes the response size a little bit bigger. Removing IPv4, although it makes the overall response size smaller, it doesn't make it so small that you would be comfortable adding more name servers

than I think. It doesn't get it under the 512 limit, which was that limit from years ago.

Is that basically what you were after? Yeah. Any other changes? Well, I mean, I think a lot of people would be nervous if the root name servers were running only on IPv6 today. I don't think we're ready yet. We do see a decent amount of queries coming in over IPv6 but certainly not enough to where you'd be comfortable running it exclusively there.

WES HARDAKER: There's been a number of interesting proposals over the years. Another one being adding more addresses to the existing identifiers, as well, but as I mentioned in one of the earlier slides, Anycast is sort of the scaling solutions these days. So identifiers are not really changing and identifiers isn't the best practice and adding Anycast instances has been solving sort of all the world's problems, but good question. Thank you. There was another one. Somebody. Yes, go ahead.

ADEEL SATIQ: So these 600-plus instances of root servers, are all of them have DNSSEC implemented on them?

WES HARDAKER: Excellent question. The interesting thing about DNSSEC is that the servers don't care. The ones actually answering the question don't care. They have to be compliant in how they return answers, but they're actually not doing any cryptography themselves. And the way that the DNS root data and most – there are some DNSSEC implementations that actually change data on the fly, but I'm not going to get into them because they're complex and usually proprietary reasons for wanting to do that.

But for the DNS root data, you can actually go get the freshest copy from IANA and you will find all the DNSSEC signatures in it. So the DNS root servers, they're just returning those answers. That includes the cryptographic signatures. They weren't created by the root server instances. They were actually created by IANA.

So they're actually not doing anything other than saying, "Oh, you want this answer." Oh, but you've marked the DO bit in the DNS protocol, which says, "I also want the DNSSEC-related answers, too." So then they're just thinking, "Okay, you want this additional stuff, too." That's all they're doing.

So they actually don't – the only thing they have to do is make sure they're returning that additional data, so the root servers don't do much. What has to be done is IANA has to sign the data

and you need your ISP's resolvers to take that data and validate it to make sure that it was true and authentic and not just for the root, but for .com and for CMU and .edu. The whole chain has to have a trusted path all the way through it with proper keys.

The root keys are the only thing that validators have to know, have preconfigured. All the rest of the keys for all the entire rest of the DNS tree actually can be queried. So, who's the key for .edu and what's the key for cmu.edu? It was actually done over the DNS protocol, so you only need to know one sort of bootstrapping key. Good question. Any other final questions? We have one more.

ADEEL SATIQ: Just to add [inaudible]. So can I find out that the query that was returned to me wasn't from DNSSEC root server or was it without it? Can I find that out?

WES HARDAKER: Yeah. Good question. So, generally, if you knew how to do it on the Command Line. If you had DNS expertise, you can definitely do that very easily. Without that, it's actually kind of difficult to know if you're in a DNSSEC-secured environment. One way to do it is if you go to DNSSEC-deployment.org, if you are in an ISP, if you're using a resolver that actually is validating, you will see a

checkbox, a green check in the title bar. If you are not in an environment where DNSSEC validation is happening, you will see as triangle with a little exclamation point in the title banner.

And there's a couple of other Webpages that do that, too, that's the one I can remember off the top of my head, but there's five or six of those you can go to, and that's the easiest way from a Web browser point of view. All right. How are we doing on time? I think.

STEVE CONTE: We've got about four minutes.

WES HARDAKER: We're about wrapped up. So, one more. Any one last question? All right. You get the last one.

MOHAMMAD ABDUL AWAL HAOLADER: Sorry. I'm Awal from Bangladesh. I'm a Fellow. My question is from that slide. To be a caucus member, do I need to be very knowledgeable about DNSSEC or DNS, or I can be a beginner technical professional and want to grow my expertise joining this team?

WES HARDAKER: That's a really good question. One of the things that you have to do, if you go this Webpage, it'll ask you to fill out a page for why you want to be a Caucus member, what your purpose is for doing it, what you can provide. So we're not just looking for technical people. We're also looking for policy-oriented policy.

There's no harm in submitting an application, and it's even likely that you'll get accepted to be a member. But actually, if you really want to dive into understanding how DNS works and stuff like that, studying up as well as going to the IETF, which is coming up in two weeks in Chicago and also is around the world every year, as well. That's actually where the protocol modifications happen and you can very quickly become an expert.

So it doesn't take a huge amount of knowledge to be a Caucus member as long as you can contribute in some positive fashion and not just to learn. We do require Caucus members to actually participate, so just going in to listen is generally not acceptable. You're expected to actually participate in the document authorship. Okay?

All right. With that, I'd like to thank you all for coming. And if you have any further questions, you can look at these resources and you can write to ask-rssac@icann.org. If you have technical questions about the whole root server system as a whole or

you're welcome to talk to any of the people around that you've heard answering today, as well, that happens to know more about the root server system. So, thank you very much for coming.

STEVE CONTE: Wes, thank you and to the other root server operators who are here. Thank you. We're going to take a short break here. The next session you have to make some choices. It's at 5:00.

[END OF TRANSCRIPTION]