BUENOS AIRES – How It Works: Domain Name Registry Protocols
Sunday, June 21, 2015 – 15:30 to 17:30
ICANN – Buenos Aires, Argentina

DAVID CONRAD: Welcome, everyone. This is the second of the How It Works tutorials. This is a introduction to addressing and routing. [inaudible] provided by Alain Durand, who is a recent joinee of ICANN, actually started in March of this year. Alain is our principal technologist, if I remember correctly – yeah. So I will hand it over to him and hope you enjoy this session on the networking technologies.

ALAIN DURAND: Thank you, David. Good afternoon. I'm going to talk about some of the fundamental technologies to have an underpinning of the Internet, how things work.

I would like to make this a little bit interactive, so if you have any questions, please feel free to interrupt me, and then we can have a discussion.

I have four agenda items. First one is networking by number. You may remember coloring by number. When I was a kid, there was this grid [inaudible] numbers. If it was 17, it had to be yellow. It was 4, it had to be green, and things like that. I'm going to try to [inaudible] some of that.

That will be my introduction, but the real meat of this presentation is about naming, addressing and routing. Those are notions that we all

use a lot in this community, but it would be interesting to have a little bit more in-depth look at what it really means.

Essentially, if you are from a community from which IP means intellectual property, well, I hope that maybe today you will learn a thing or two, but it may also mean something else. If you already know what it means – IP protocol – well, you know everything and I have nothing to tell you.

So, networking by number [inaudible] I'm going to take some examples, actually from layer zero to layer nine. Traditionally people talk about layer one to seven. We're going to expand this a little bit because this is the 21st century. We can expand things more.

I don't want this to be too academic or too technical, so I've tried to use it for examples and try to show you what it looks like.

Layer zero is a physical one. That's the only thing that we cannot really virtualize. If we have to put a fiber somewhere in the ground, somebody has to go and dig up a trench and run some cable for it. If you have to put an antenna somewhere to do wireless, you may have to go to the rooftop of an historical building, like the one in the picture on the right-hand side, and put this antenna into [inaudible]. There's really nothing you can do about that. This is the only part of the Internet that cannot be virtualized. This is purely physical.

Values for wireless, values for wired: essentially at the end of the day, it's a usage question and a speed question. Typically wired has more infrastructure but has higher bandwidth, higher speed. Wireless? Well,

it's easier to deploy, less than 500 pieces to put in place, but typically you will get less overall bandwidth.

Let's move up a layer. Now that you have a fiber in the ground or you have a wireless antenna somewhere, you want to send information. So in the fiber, what you're going to do is to [inaudible] the fiber. You're going to send some light at one end and receive a light at the other end. Your fiber can be really short. It can be cap and fit. Or it can be really long. It can be thousands of miles, if you go underneath the ocean.

You're going to send light of a certain color, a certain wavelength. You can send green and Steve there can decide to send red. So on the other end, if I have receiver that only receives green, well, it will receive my traffic. If I have a receiver that receives red, it will receive Steve's traffic. That way you can share infrastructure.

The thing to remember – this is really point to point – is that one end of a fiber to the other end of a fiber. [inaudible] in the middle you see nothing. So that's really the most elemental part of networking that you can do, simply send light from one place to the other.

I'm going to move on a little bit on the layer stack. Now, let's say I want to have a fiber that goes from my office in D.C. all the way to here in Buenos Aires in this hotel. It's unlikely that I'm going to dig a trench, that we'll go from the hotel all the way through Argentina and the other countries and maybe underwater all the way to Philadelphia, and have only one fiber. And if I wanted to have another one that goes to Chicago, I would have to do the same. It makes absolutely no sense.

So instead what is happening is people just lay lots and lots of fibers in different places, and then you patch them together. So you may have some active equipment that simply connects those fibers together for you, and you reconstruct a fiber path going through there.

Maybe you're using this green color that I was talking about on the first leg. On the second leg, maybe the signal will be regenerated and maybe you will use another color, maybe a blue or red or whatever it is. Somehow like that, the fiber operator will reconstruct an end-to-end path by stitching together those different fiber elements.

What kind of speed can you get with fibers? Well, speed today in fibers is measured in gigabits per second. GigE means gigabits per second, essentially gigabit Ethernet.

The lowest speed you could get today is essentially one gigabit per second (1 GigE). Typically you would get more than that, so people will lay out ten gigabit per second (10 GigE). That's what you get traditionally in fibers that you will see in the ground.

The newest [inaudible] speeds now try to go higher. The best you can get today is about 100 gigabit Ethernet. It requires some special equipment, but this is something that can be done.

In data centers, people have said, "We have a need for some intermediary speeds." So, typically you will see servers that are connected either with 40 gigabit Ethernet or maybe 25 gigabit Ethernet. So, 40 is relatively easy. [inaudible] four times ten, so you take the same infrastructure that you would have with a 10 gigabit

Ethernet and you multiplex them so you have four colors, essentially, and that gives you 40.

25 is said to be more difficult, but this is a new standard that is coming up now.

You can have multiple of the above, as I was mentioning. If you have two colors or three colors or four colors that you send through those fibers, you can recombine them together and have multiples of that.

If you really want more than 100 gigabits per second, well, you take two of them, or four of them. In a sense, you spread your traffic on those different paths, and then you have a multiple of a speed.

Moving up the stack, we're into networking. Now you have a bunch of fibers and fiber paths everywhere. You want to connect everybody together. Well, the world is not flat. What this means is that when you are on the same network as your neighbor, you can see each other's traffic. So imagine a network that will cover the entire planet where everybody could see every single bit on the packet, every single packet that's being sent. Especially when you have a storm of packets, everybody will see the same storm, and that will essentially take the network down.

To avoid that, we have introduced the notion of topology isolation. I'm going to create a local network, and, yes, all the hosts, all the machines in my local network can see each other, but they cannot see the machines that are on the other side of the planet. So it's isolated,

and it's going to be reconnected at layer three, for the protocol called IP, which is the Internet Protocol.

Once again, this is not intellectual property. This is Internet Protocol.

Historically, there have been different protocols that were layer three. But today there's really only one – IP – and we'll talk about it later. There are two versions of it. IP Version 4 and IP Version 6.

Moving up the stack, this one's called transport. You want to send packets, and now I have a path to go from my local network to some other network, and then in the end to Steve's network. All this is all connected at layer three. They go through a bunch of different fiber paths.

Now I want to send data. When I want to send data, I want to make sure that Steve gets what I'm saying, and everything of what I'm saying, but nothing more.

So what we're going to do is to send bits at a time, chunk the message into smaller bits that you would call packets, and we send one packet – that's what's in the diagram. I'm going to send Message Number 1, one packet, and see if he's going to get back to me, saying, "I have received your message."

Great. I know he has my first packet. I'm going to send the second one. Steve says, "Yeah. Perfect." He received the second one.

But now I'm going to send the third packet, and that packet is going to be lost for a reason or another. Steve cannot tell me, "I've lost your packet." He doesn't know he hasn't even received a packet.

So what I'm going to do is I'm going to wait for him to tell me something, but he cannot tell me anything, so I'm going to wait and wait until the timer expires. When the timer expires, I'll say, "Okay, most probably something went wrong. Either my packet was lost to Steve or Steve's acknowledgement of my packet was lost to me. So I better retransmit."

That's what is really happening here. I'm going to retransmit the packet, and this time around, Steve gets it, sends me the acknowledgement. Great. I have sent all my traffic.

Now, Steve knows that I've sent my three packets. I know that he has received my three packets. We're all fine.

This is what this TCP protocol is all about. TCP for Transmission Control Protocol. It's a way to make sure that every single packet arrives. If they need to be retransmitted because there has been some packet loss in the middle, we will do that for you.

There's another protocol called UDP. That is a lighter version of TCP that doesn't deal with handshakes and acknowledgements. It just sends packets, meaning that the application at a higher level would have to do it.

Most applications today on the Internet use TCP. Every time I've seen an application claiming that it can do better, it has not necessarily

been a very good success story in the end. But there are sometimes occasions where it actually makes sense to do so.

Moving up a stack: sessions. Well, streaming is a big thing today on the Internet, when you stream movies or you stream audio – like this Adobe Connect is being streamed, actually. What streaming means is that you are not simply downloading a big file and you want to have a full file. You have something that is real-time, meaning that there is no point in re-transmitting what I said 15 minutes ago or five minutes ago or 30 seconds ago. If I'm losing some packets in the middle, that's fine.

What we should do is to simply degrade the quality of a transmission. Maybe if I had an HD movie, and I'm sending packets to stream this HD movie, and at some point there's some congestions or packet loss, I have to go from HD (high definition) to SD (standard definition). That will take up less bandwidth and maybe have a better chance of this going through. So maybe for 30 seconds we'll have a lower-quality picture in your movie, but at least you can follow the movie. It's not [inaudible] stop.

So that's what streaming protocols are all about. It's adapting really the bandwidth that you're using to the quality of the experience that you have and to the congestion or packet loss that can happen.

If I move up a stack: presentation. Now, I'm sending some data, and right now I've been just talking, talking, talking. It's not exactly structured. That's fine because we are human. We can understand that.

A machine can't, so if you send just a bunch of data to another machine, it's just garbage, essentially. So the data, especially if you send things like a database, or if you send configuration information, all of this has to be structured.

Historically, it was simply some fixed text, and you have to go and parse this text to understand it. There was some algorithm to go and figure out how to do that.

Then some of a more complex representation came, like ASN.1. It was a binary form. XML, I'm sure you all have heard about XML. It's one of the things from about 20 or 30 years ago that has been used a lot. XML enables you to express different types and different categories in a document.

The latest one that is the fashion now, all the rage, is called JSON. I could look at this as a simplified version of XML that enables to do simpler things without having a syntax which is overly complex.

On the right-hand side, I have put a JSON example on how to describe a menu. It's a menu with a couple items in it, and it's a pop-up menu that has three different actions possible: New, Open, and Close. There is a primitive that's associated to each of those actions. So if you click on Open, for example, then it will start the Open Documentation procedure.

This is how people use code to actually embed some definition and content that is transferred from one machine to another. It could be used in a user interface, in a graphic user interface. It could be used in

between two machines that are exchanging data or configuration information.

So that's a way essentially for the computers to structure this information.

Now that all this information is exchanged, at the end of the day, computers and networking, this is not for machines. This is for humans. That's what we all want to do. We all want to be using the Internet whichever way we'd like. If we're in our bedroom, if we are in Phoenix and ready to go the swimming pool, or in an airplane, or in a conference – what we really want is to use the Internet.

All this technology that I have explained underneath is there for only for one reason. It's for us to actually go and use it.

So there is a major, major protocol that is useful [inaudible] called HTTP (for Hypertext Transfer Protocol), or the secure version of it, HTTPS now. If you look at repartitions of different protocols on the Internet, this is really what is being used the most – that and [inaudible], both of them.

Now, those are the traditional seven layers in the OSI model that everybody learned at school. Actually, there are more and more layers on top of that. The next one is the financial layer.

When you build a network, you have to make sure that it makes financial sense. If you build over capacity, then you spend a lot of money to do that and you will never recoup your money. So at the

end, you will go bankrupt, and it's not exactly a good service to a customer if you go bankrupt.

Or conversely, if you don't put enough infrastructure in place, then there's not enough bandwidth and your users are not happy, and if they're not happy, they may simply leave you to go somewhere else. So finding the right balance between how much investment you have to put in place to make the network happy and what is the return on this investment, is one of the key things that happen in the Internet.

There's a layer above that, above this financial layer, that we usually call the political layer. That's where we all are today, to take discussions and talk about Internet governance and about all these things taking place and how the committees organize themselves around all this Internet. That's where we all are.

That's essentially my introduction, like a real foundation about some of the technology we are going to use.

If there's there any questions that I can take now before I move onto the next part? Okay.

I'm going to move onto the next part then. This is about naming, addressing, and routing. I'm going to use an example to drive us through all of this. This is not a real example that is happening today, but it could have. Think about somebody who is traveling to an ICANN meeting, for example, and all of a sudden has a toothache. It's really, really hurting, especially after a long flight. I'm not going to go home any time soon. It's just I need a dentist, okay?

So I arrive here. I need a dentist. I need to ask Steve. "Steve, who's your dentist? What's his name? Carlos? Okay. I need to find a way to go to Carlos to get my teeth examined."

This is what we're going to do for this tutorial now: naming. What's a name? Carlos. Well, a name, if I look up a definition in the Webster dictionary, is "a word or a set of words by which a person, an animal, a place, or thing is known, addressed or referred to." Example: m y name is [inaudible], or my name is Carlos.

So the definition is a famous person. Same thing. Carlos is probably a very famous dentist around here. So if I know your name, I know who you are. I know that Carlos is a dentist here in Buenos Aires. He's very good. He's very famous. But that's all I know. It doesn't tell me in any way, shape or form to go and get my tooth being addressed, my problem being addressed.

So I need more information than that. But let's dig further into this name thing. Let's look at it from the other side. This dentist, Carlos, has a name. Okay. He knows his name, but if nobody else on the planet or in the city knows about Carlos, he won't see that many customers. So making sure that other people know about you and know your name is really important. That's how you get known. That's how people will hear about you.

So knowing the name of somebody or knowing the name of something, it can enable us to talk to something, to talk to a person, or to talk about a person.

Now, we have been talking about Carlos so far, but we haven't really talked to Carlos. That's the next thing that we're going to do.

So Steve told me, "The name of my dentist is Carlos." Okay, let's move on. Names have scopes, so within a scope, you need to have unique names. For example, my name is Alain, or Alain if you speak French. I was the only Alain in my family. So whenever there was a family gathering, and somebody says, "Alain, go do something," it was me. Nobody else. I had to go do the dishes. Fine.

But when I was in school, Alain was a fairly popular name back then, so there was always four of five Alains in the class. So when the teacher sats, "Oh, Alain, go to the blackboard, we were all looking at ourselves and say, "Okay, which one?" Well, the teacher was getting upset because Alain was not going to the blackboard, and she had to say, "Alain Durand, go to the blackboard." Okay. Now I could go there.

This is an example to show you that you need to find a way to disambiguify names through their context and really find the scopes for which they are really unique.

But again, I've heard about this Carlos guy, but I have no idea where he is. So I need to somehow map the name of this Carlos dentist into his address, and that's what we call name resolution. That's what the DNS is all about.

If I want to go to see Dr. Carlos, what am I going to do? Well, probably I will go to the concierge at the hotel and ask him, "Do you know the address of this Dr. Carlos that is very famous around here?" And what

will the concierge do? He will take up his Rolodex or his directory and he's going to go through all the different letters and say, "Oh. C. Carlos. Here it is. 125 Root Canal Road. Here you go." I have the address of this dentist.

What the concierge just did is exactly the same thing as what a DNS does. It takes a name – dot dot dot dot – ICANN.org – and transforms it into an address, which is an IP address, like, I don't know, 128.something – whatever the current IP address is. This is exactly the same process: finding the name, transferring it into an address.

If I look at the DNS now from a more technical perspective, this is something that was done very early on in the Internet. You may have seen some of the presentation, those who were earlier this morning: the RFC's document from the IETF that described the DNS of 935, if I remember correctly. They were done more than 20 years ago.

But the DNS has continued evolving, and some of the more recent issues around DNS were about internationalization. How do you represent a name in different languages? It could be in Chinese. It could be in Arabic. It could be in French, in English. It's not just one language and one character set. So there have been a lot of discussions about that.

Authentication, as we know, is also a big issue DNS security, DNSSEC. How do you make sure that, when I go to 125 Root Canal Road, that's really going to see Dr. Carlos, this is not a bogus address that I have been given? So this is all about DNSSEC.

Expansion of a root zone. Well, you have all heard about the gTLD program in ICANN, so I'm not going to talk much about that.

Those are three examples of more recent issues in the last, let's say, five, ten years, that have increased the power of a DNS.

Now I know 125 Root Canal Road in Buenos Aires. I have this address, so let's talk a little bit more about addresses. Again, the definition from the Webster: "An address is the particulars of a place where someone lives or where an organization is situated." An example that they give is they exchange addresses and agree to keep in touch. Again, that's this notion, if I know my address and nobody else does, that's nothing good to me. I need to somehow disseminate this information.

So if I know the address, I know where you are. The dentist, Dr. Carlos, 125 Root Canal Road. I know exactly where it is.

Let's take a little bit of a detour now. I'm going to make a parenthesis about addresses. Then we'll go back to Dr. Carlos. I live in D.C., and the most famous address in D.C. is 1600 Pennsylvania Avenue, Northwest Washington, D.C., zip code 20500-0003, U.S.A. It's a small house, painted white. There's a field around it. It's really nice.

If you look at this, when I read the address like this, there's actually a structure in it. There's a hierarchical structure. First you have to read backward from the end to the beginning. So U.S.A., that's the country. D.C., the District of Columbia, that's essentially the [inaudible] or the state – except that D.C. is not a state. That's just a small detail.

Northwest identified which part of D.C., which quadrant of D.C. Pennsylvania Avenue identifies the road in there, and 1600 is the house number on that road. So there's a very nice, clean hierarchy.

But not everything is organized geographically like this. For example, in the U.S., we have this 1-800 telephone number system, which is a toll-free number. You can dial the number and you don't have to pay for the communication.

Well, this is a completely flat space. You have no idea if the person on the other side is going to be in the same city, the same state, even the same country. Sometimes the 1-800 number is rerouted to India or to China or to wherever it can be.

It's the same thing if you have cell phone numbers. I had a cell phone number when I lived in California. When I moved to Pennsylvania, I kept it for a while, so people were a little bit confused. They were saying, "You have a California number and you live in Philadelphia?" That was a little weird. So I changed that. But I didn't have to.

Similarly, IP addresses – because that's what we really care about here – are not organized along geographical boundaries. It's not like you have Network 5, which is in Argentina, Network 6, which is in Brazil, and Network 7, which is in France. This is nothing like that. It's all organized by service provider, not by geography.

Still, there is some hierarchy. We will talk about that later. But this hierarchy is not necessarily a geographic hierarchy.

Similarly, if we make a parallel with the DNS and the namespace, you have some of it that has a geographical organization. For example, the ccTLD, you may have a .fr for France. You may have a .ar for Argentina [inaudible] geographical organization, but if you look at things like .com, it can be anywhere. So you can have a mix of the two.

As I mentioned earlier, names have scopes, but addresses also have scopes. If I am in D.C., when I'm in D.C, and somebody tells me "1600 Pennsylvania Avenue, Northwest" – they don't specify D.C., they don't specify the zip code, they don't say U.S.A. – I know it's in D.C. It's a very famous address. No problem. If I'm on Google and I just type this address, it will find it immediately in D.C.

Or another example: when I'm in Europe and somebody tells me, "Oh, I'm from Paris," I know exactly what it is. It's Paris, France. A big city. Eiffel Tower and all of this. Good restaurants.

But if I live in the U.S., there are 29 cities called Paris in the U.S. So if I have someone telling me, "I'm from Paris" in the U.S., and I know he's not French, then I have absolutely no idea where he is.

So same idea. I need to have scope. So if he says it's in Paris, Texas – okay, now I know which Paris you're talking about. Not just Paris.

Exactly the same as with names. I can use an address as a handle. I can use the address to go there. This is what we are going to try to do: go over to 125 Root Canal Road. Or I can simply use that as a reference, like the concierge told me that Dr. Carlos is at 125 Root

Canal Road. Same thing. Either use it directly to communicate or to pass it on as a reference.

An address is still not enough to achieve communication. Let's say that I'm going to send a postcard to this very well-known address, 1600 Pennsylvania Avenue, Northwest Washington, D.C., 20500, U.S.A. I can send a postcard there from anywhere in the world. It will arrive.

Why? It's not magic. It's because there's a system, called the post office system, that interconnects with different countries. It's going to make sure that they will take the postcard from one place to another. They will route it all the way to Washington, D.C., and then it will go in the end to the White House. [inaudible]

In the Internet, there's similar system, similar construct, that will enable my data, my packets, to go from one place to another, and that's what we call really forwarding or routing.

Now, going back to my example, my teeth are hurting. I know the address of Dr. Carlos – 125 Root Canal Road in D.C. Let's go there.

First off, I need to use the Internet Protocol (IP). I was mentioning earlier that there are two versions of the Internet Protocol. There's IPv4 and IPv6. IPv4 was designed in the early '80s and is still in use now. IPv6 was designed in the early '90s and is starting to be in use now.

The difference between the two is mostly about the format of the addresses. In IPv4, this is 32-bit to describe an address. In IPv6, it's 128-bit. Think about how many digits are in your phone number. It

could be four digits, six digits, or ten digits. The more digits you have, the more telephone numbers you can have.

Same thing with addresses. In IPv4, 32-bit, you could have 4.9 billion addresses. Out of them, about 3.2 billion are useable, and most of them are allocated.

In IPv6 – [inaudible] maybe read this number – I've been told it's 340 in the zillions. I guess that's true. I tried to convert it into billions, so it's 340 billions of billions of billions of billions. The expectation is that we will not run out of space there any time soon.

So [inaudible] 3.2 billion of IPv4 addresses have all been allocated, and we have I'm all sure heard about this thing called the IPv4 address exhaustion that really started to happen in 2011. When we talk about exhaustion, it's not that the address gets tired. They are not exhausted that way. But it's really that all of them but a few have been allocated. So there are none left unallocated for new usage.

But the Internet is still growing, right? The Internet does not stop growing in 2011. [inaudible] much, much bigger than it was four years ago. So something had to happen.

Initially, IPv6 was designed in order to address this problem, provide more address space. But the issue is that they are not compatible. If you have an IPv6 machine and an IPv4 machine, they cannot talk to each other.

An example, it's the same as a power supply for my Mac. It's a U.S. power supply and I cannot plug it into the sockets in the wall here. I

need to have some kind of an adaptor here. So v4 and v6 is the same thing, cannot communicate. It's just a technical thing, and there was absolutely nothing that could have been done to make it different.

So what do people do? Well, first they try to say, "Okay, let's abandon IPv4 and just do IPv6." The problem is it doesn't really work from a financial perspective. If you remember the slides I had in my first part on layer eight (financial), and I said it has to make financial sense to do something.

Today, if we look at the equipment on the Internet, not all of them support both IPv4 and IPv6. Many do. My cellphone does. But some don't. The TV that I have at home is a smart TV that can download movies straight from Netflix or whatever streaming service. This TV doesn't do IPv6, only IPv4.

So, there are some large service providers in the U.S. – for example, Comcast – that have done a really, really big effort to deploy IPv6. They even went to all the way to help people get a new home router, a new home gateway. But as IPv6, I can request IPv6.

They claimed at the last NANOG meeting, just a month ago, that they are at about 60% of their user base that is configured with IPv6. You would expect that we get a lot of v6 traffic. They said at the same meeting that we only get 10% of traffic in visits. 60% of the user base has v6, only 10% of the traffic.

You can say that's because of the content that's not ready. No. The content is things like Netflix and YouTube. All this is v6-capable. So it

has to be the device that is in front of the eyeballs. It has to be the TV or it has to be any kind of a smart box that is plugged directly to the TV that is not v6.

The point I am trying to make here is you can build all the infrastructure, put it in place, but until every single element there that matters can do both IPv4 and IPv6, you cannot abandon IPv4, and it's going to be like that for a very, very, very long time. It may go at different speeds in different parts of the world, but this is not a transition that's going to happen in two years. It's going to take maybe 10, 20. I don't know.

So if you still need to maintain IPv4, at least as a service, there are some technologies there that can help to simplify it. But you need IPv4 addresses, at least some of them.

So two points that I wanted to make. I said that all the addresses have been allocated. That's true. It doesn't mean that all the addresses are in use. There are some places that have a number of IPv4 resources and they don't really use them and they are willing to let them go and transfer them to somebody who needs them. Usually those transfers involve some kind of money being exchanged. This happens quite a lot. We're going to see some statistics later on.

The Internet addresses are handled to their customers and subscribers and service providers by the IR, the original Internet registries. There are five of them. Historically the first one was RIPE in Europe. It was APNIC in Asia-Pacific. It was ARIN in North America. It was LACNIC in Latin America. I think the last one that was created was

AFRINIC. So that's five IR. All have policies now that enable those transfers, and they register the transfer from one block of addresses from a previous owner to the new owner.

The terms and conditions on how you can do those transfers vary a lot from one registry to another. For example, in the RIPE region in Europe, they have decided that you can transfer essentially whatever you want.

In the ARIN region, they have decided something different. It was a need-based policy that was in place over the last 20 years that essentially guaranteed that addresses go to people who need them, and they have decided for now to keep this policy in place. So if you want some addresses, before you can actually do the transfer, you have to qualify for that with ARIN. Then you get your addresses from whoever is willing to give them to you in exchange of maybe money.

So all of those policies are in place with more or less hurdles to get through them, but you can transfer addresses and then keep growing this way.

But now that you have addresses, what you really, really want is to make sure that you get the best possible use of those addresses. Then if you can leverage them and maybe share them, it's a much better usage. So there are technologies like address sharing, also called NAT for Network Address Translation, which is the same thing that you have in the little box that is connecting your home network to your provider. You have one address, usually, given by a provider, and then you share it among maybe five, ten, or 20 devices in the home.

There are bigger boxes on the exact same model that are called Carrier-Grade NATs that have been deployed by large service providers to share one IP address among maybe 10 or 50 or 100, sometimes 1000, subscribers. So even if you have a small pool of addresses – let's say you get /16, which is 65,000 addresses – and you multiply and leverage with using NAT in, let's say, with a ratio of 1:100, now you have six million addresses, essentially, or equivalent of six million subscribers that can use that.

That may be enough to run a fairly decent-sized service provider. Using this combination of those two things – like sharing addresses and then transferring addresses from wherever you need them – you can still hold your IPv4 domain.

It's not ideal. It would be better if everybody could go to v6, but as I mentioned earlier, this is not happening yet.

What is the reality of those transfers? Because it's something fairly recent. I wanted to show you some statistics. This chart goes from essentially April 2012 to May 2015, so the last three years. Here we see that essentially until 2014 there was very, very, very little transfer. So this is something new. This is something that is really happening in the last twelve months.

There's quite a significant ramp-up. We are seeing globally, worldwide – actually, as a caveat to this, I'm only looking at RIPE, APNIC, and ARIN, which are the three biggest IRs – we are seeing about 300-400 now transfers a month. It's not huge, but it's getting significant.

You will see that RIPE, in blue on this diagram here, is doing the most of the transfers, like almost all of them. But transfers of blocks of addresses are not identical. Some blocks are small. Some blocks are large. So we need to essentially go a little bit further into this analysis and say, "How big were the blocks that were transferred?" and try to multiple the number of transfers and average out by the size of the blocks.

If I do that and essentially use /24 equivalent, which is essentially the smallest block that we can transfer today, a /24 network means that you can have 256 addresses. [inaudible]

If you like at this, now the diagram is a very different color. It's all red. So in volume of transfer, volume of addresses being moved around – not simply the number of transfers themselves – this is really transferring from the ARIN region in North America.

So the transfer can happen within the region, or they can go out of region. There are some policies in ARIN that enable the transfer of addresses out of ARIN to other regions if they have compatible policies on need-base. For example, the APNIC region has a need-based policy. Some are compatible with ARIN, and so you can transfer addresses out of North America to go to Asia. In Europe, RIPE has a very different policy, so you cannot transfer addresses from ARIN to RIPE.

Okay, so it was apparent on addresses. Now, remember, I still have my toothache, and I need to go to see this Dr. Carlos. I know his address – 125 Root Canal Road. Okay, how do I go there?

I need a route to go there. Well, I said "route" but according to the dictionary, I should say "route." So I guess it depends where you're from in the world if you say "route" or "route." The route is a way, a course taken, in getting from a starting point to a destination. I want to go from here to Dr. Carlos.

Now, remember, if I have a name, I know who you are. If I have an address, I know where you are. If I have a route, I know how to get to you.

So how does this work in practice? Before I can send my data to Steve or if I can go to Carlos, routes have to be built. There's a routing system the same way there was a postal system that sends the postcards. All this is built before you actually send a postcard, put your stamp on the postcard, and put it in the mailbox.

Similarly, those routes are being built by the service providers. How do they do that? Well, they do that going reverse from where the traffic will flow. I have this diagram. I'm here at the source and I want to go to Dr. Carlos, which is my destination. So it's going to go the other way around. Essentially, the destination is going to announce to whatever it's connected to, "I'm here. I have this address, this address block in IPv4, IPv6. I'm here. If you want to join me, that's where I am."

This node here is receiving this announcement and says, "Okay, now I can reach this address through that link." Now I'm going to propagate this information to my neighbors. But what I have in green here I'm going to tell my two neighbors that I know how to reach Dr. Carlos. Okay.

Now, most of the neighbors are receiving this information, and essentially it is, “Huh. Now I know a guy who knows how to go to Dr. Carlos. Great. I have this information.” They’re going to do exactly the same thing. They are going to reannounce, propagate this information to other people they’re connected to.

This guy here at the bottom of my diagram is going to receive this information, and now he says, “Huh. I know a guy who knows a guy who knows how to reach Dr. Carlos.”

This is what the Internet is all about: “I know a guy who knows a guy.” This is a very cooperative system. In the end, that’s me. I’ll receive this information from this guy. “Yes, I know a guy who knows a guy who can go to Dr. Carlos.”

If there’s a bad actor in there, that’s a bit of a problem. If something doesn’t go [inaudible], that’s a bit of a problem. But the reality is, in the Internet, all the service providers are all cooperating to make this work.

So you see there are different ways to get to Dr. Carlos. I could have gone from the top, from the middle, from the bottom. I could have made all kinds of different paths in there. In this diagram, the path that is in yellow happened to be the shortest one. There’s only one, two, three ups. This one will be much longer. That one would be longer. Any other would be longer.

So those routing protocols usually make decisions about, of multiple paths going to Dr. Carlos, which one am I going to use? Typically, this

is the shortest path. It doesn't have to be. There could be some policies in place. There could be policies that say, "I'm here. Yeah, this guy's telling me a short path, but this service provider I don't really like. They charge me too much. I would rather like to go for longest path, but that would be a more economical path." So I'm going to say, "No, I don't like this route. I will go north."

All those are decisions that every single service provider can make. It all works in the end because they all collaborate to receive those routes.

So remember, this is all about, "I know a guy who knows a guy who knows Dr. Carlos."

So now that those routes are place, I can send my traffic. I can go to Dr. Carlos. That will be the equivalent of me going from there to Dr. Carlos, and at every street intersection, there is a sign that says, "Dr. Carlos, To The Left," "Dr. Carlos, Go To Florida Avenue," "Dr. Carlos," every time left, right or in the middle.

Now I'm going to take exactly the reverse path from the one I used, the one I've learned the route from, and simply for one of my packets. So I've learned from this guy that the new guy [inaudible] Dr. Carlos. Great. I trust him. Keyword here: I trust him. Send him my traffic.

This guy heard from the previous guy that he knew a guy who knew Dr. Carlos. Great. He's going to trust him. Then like that, for every single hop until the packet reaches its destination.

So what you remember from this: this is a cooperative system. It works on trust. It works because all the different service providers have this privileged relation between each other that enables them to say, "Yes, I know a guy," or, "I know a guy who knows a guy."

There is not a single entity there that knows everything. There is not a police that says, "You cannot turn left here." There is no such thing. It all works because everybody trusts each other.

Now, sometimes problems happen. Problems could happen when somebody is a bad actor in this system. There was a case a number of years ago where in the country they were not happy with some content coming from a big content provider, and they wanted to take it down. It created a leak of information that really [inaudible] the system.

I'm going to explain how this could work. Let's say that I have a bad guy somewhere here at the bottom of my diagram. The bad guy wants to pretend he is Dr. Carlos, so when I will be there, instead of helping me with my teeth, he will probably point a gun to my head and ask for my watch or something like that.

So what is this bad guy doing? He is going to pretend he's Dr. Carlos. He's going to announce very loud with a loud speaker that "I am Dr. Carlos." If he can convince a number of service providers upstream that he is Dr. Carlos, all those guys announce that Dr. Carlos is over here.

When we reach a point where somebody knows, "Oh, Dr. Carlos could be there or there," he has to make a choice. Who is he listening to? Okay?

So if a bad guy is speaking louder, essentially, than the good guy, well, we say, "I need to send the traffic to the bad guy." Right? So when all those announcements in the end go to the source when I'm sending the traffic, forwarding it hop by hop, it will end at the bad guy.

To prevent that, there's a system that has been created. It's called RPKI for Resource Public Key Infrastructure. Essentially what this is about is not simply listening to the guy who is talking the loudest, but when you build those routes, having some kind of a certificate of origin saying that whoever is announcing to be Dr. Carlos is really Dr. Carlos. It has been certified by the Board of Dentists in Argentina.

When you do that, this bad guy can maybe speak loud enough to corrupt some of his neighbors, but when you go into the global routing system and when we verify the signatures, then we say, "Oh, I'm receiving these announcements really loud from this bad guy, but the signature doesn't match. I'm receiving another announcement from the guy whose signature matches. Okay." I can simply drop the bad announcement and just keep sending my traffic to the real guy, to the good destination.

This system is still under deployment. It has some issues that are still being discussed. Is it going to be a centralized or decentralized system? Is doing the authentication of the source of the origin, is it

enough? Or do we also need to validate every single hop in the path? So those are discussions that are still happening.

But this is one of the interesting things, interesting developments, in the Internet. You can look at this as there are three major things in the Internet. You have names. You have addresses. You have routes. Names? The major development was DNSSEC to add security. IP addresses? The major deployment was IPv6 to have more addresses. And routing? The major development is trying to bring security to this routing system. And with RPKI (the Resource Public Key Infrastructure), is one other major piece to do that.

So I'm able to send my packets now and I can arrive to Dr. Carlos, and Dr. Carlos can now take care of my problem. Thank you, Steve. He was a really good doctor.

That's the end of my presentation. I would be happy to entertain any questions that you may have.

Please state your name for the record.

RICK LAMB: Oh, okay. Hi, I'm Rick Lamb. I work at ICANN. But I have a question about the current state of affairs. How long typically, if the bad guy directs a route like that, how does that problem exist for us usually? Is it corrected quickly or is it days? Months?

ALAIN DURAND: There have been a number of incidents like that, like route injection in the past. Typically it's detected within hours and it is corrected within a day or two maximum.

RICK LAMB: [inaudible]

ALAIN DURAND: Yeah. And this is all because all the service providers do collaborate with each other. When they see something that is out of the ordinary, they usually take action immediately.

Another question? Please state your name again.

VICTOR MARTINEZ: Yes, thank you. My name is Victor Martinez. My question is something about the addressing. What kind of risk we can find with the multiple use of NATs around the network? This is in the process of the transition from IPv4 to IPv6. You said that there is some kind of solution. You've seen the NAT. But maybe this provokes some problems inside the routing network.

ALAIN DURAND: Okay, so the question is, "Does NAT help us or does NAT hurt essentially in the network?" I look at this after working for maybe the last 20 years on this issue as NAT is a necessary evil. You have no choice today but to use that if you are out of addresses. You have a

finite number of addresses and you want to use them in a bigger set, so you need to multiply them. There's really no other way.

On that, I've made a lot of progress in the last five, ten years. It comes with some problems. One of the bigger problems is, as people are sharing the same address, on the other side, on the server side, you only see one address maybe for ten or 100 people.

So if you're from, for example, from a law enforcement community, how to do you attribution of traffic? How do you know who was really coming there? There have been some recommendations that were published at IETF as an article, Article Number 6302, that has been published, that makes some recommendation that when you do logging on the servers, not only do you log the IP address that is coming in, but that's not enough now to do attribution. You also need to log the source port number that was coming in.

If you have those two information, IP address and source port number, you can pass this to the law enforcement community, that can then go back along the chain to a service provider that operates the NAT. By looking at those two parameters, they can do the final attribution of the traffic. So that does somewhat help a lot.

Does this answer your question?

DANIEL EBANKS: Afternoon. Daniel Ebanks, Cayman Islands. Quick question for you, or two quick questions. One, what has been the uptake with ISPs worldwide? And secondly, how do we encourage them?

ALAIN DURAND: Could you please clarify the uptake of –

DANIEL EBANKS: RPKI.

ALAIN DURAND: RPKI. Okay. It's not deployed really widely now. It's still at the very beginning. Trying to address the two issues that I mentioned earlier, trying to get resolution on how this is going to be deployed, either centralized or decentralized. Will it really help?

Also, your second question is, "Is it going to be enough or do people want to have something a little bit more comprehensive, that we do the verification of the entire path, not just the origin?"

Participating into those discussions, trying to get progress, and the resolution of those two issues will probably the most effective way to get this technology adopted.

Next question?

UNIDENTIFIED MALE: [inaudible], Cayman Islands. Is RPKI unique to IPv4 or v6, or can it work in both network types?

ALAIN DURAND: RPKI works both v4 or v6.

UNIDENTIFIED MALE: Is it up to individual ISPs to deploy it all along the route?

ALAIN DURAND: You don't deploy it all along the routes necessarily. You can deploy it only on your exchange points, for example, where you accept the routes coming in. So it doesn't have to be on every single router.

UNIDENTIFIED MALE: Okay.

ALAIN DURAND: Next question?

UNIDENTIFIED MALE: Let me check online. One second, please. Are there any questions online?

ALAIN DURAND: All right. So well then, thank you very, very much. I'm here all week. If you have any further questions or want to have any discussions on this topic, please feel free to reach out to me.

Good afternoon.

DAVID CONRAD: Is this on? There we go. I'm David Conrad, the ICANN CTI. I forgot to introduce myself at the beginning. We have one more tutorial that occurs at 3:30. If interest in DNS Registry Protocols, that session would be for you. I hope you enjoyed this session. We'll plan on doing tutorials similar to this at future ICANN meetings. We are intending to do something at Dublin. If this is of interest to you, keep your eye out. We'll probably have some sort of registration so we have an idea how many people to size the room for. But in any event, it will be sort of a similar style and content. So thank you for your attendance, and hope you found it worthwhile. Thank you.

**[END OF TRANSCRIPTION]**