

SAN JUAN – Cómo funciona: aspectos básicos del DNS

Lunes, 12 de marzo de 2018 - 17:00 a 18:30 AST

ICANN61 | San Juan, Puerto Rico

**CATHY PETERSEN:** Buenas tardes. Bienvenidos a todos a la sesión de cómo funciona, aspectos básicos del DNS. Tenemos al señor Larson, que es el vicepresidente, la persona encargada de investigación en la oficina del CO de la ICANN. Como pueden ver hay pocas personas aquí físicamente presentes, así que si no les molesta por favor vengan hacia delante y vamos a empezar con la sesión.

**MATT LARSON:** Buenas tardes a todos. Bienvenidos a esta sesión de cómo funciona, aspectos básicos del DNS. Somos muy pocos aquí así que por favor levantan la mano si tienen alguna pregunta. Tenemos mucho tiempo para cubrir el material, así que podemos parar en cualquier momento y responder preguntas.

Las direcciones de IP son fáciles de entender para las máquinas, pero difíciles de entender por parte de las personas. Por eso vamos a hablar de DNS. Cuando solo teníamos direcciones IPv4 había esperanza de que los seres humanos pudieran recordar un par de direcciones IPv4, pero es un ejemplo de una dirección

---

***Nota: El contenido de este documento es producto resultante de la transcripción de un archivo de audio a un archivo de texto. Si bien la transcripción es fiel al audio en su mayor proporción, en algunos casos puede hallarse incompleta o inexacta por falta de fidelidad del audio, como también puede haber sido corregida gramaticalmente para mejorar la calidad y comprensión del texto. Esta transcripción es proporcionada como material adicional al archivo, pero no debe ser considerada como registro autoritativo.***

---

IPv6, y la que vemos aquí tiene pocos caracteres. Puede haber direcciones más largas y es prácticamente imposible recordar las direcciones IPv6.

El tema es que las personas necesitan usar nombres. Las computadoras y los routers utilizan números. Las personas, nombres. Cuando empezó, los nombres eran simples. Teníamos etiquetados únicos, nombres que no tenían datos. Nombres de dominio porque los nombres de dominio todavía no se habían inventado. Entonces todos los nombres de la Internet tenían que entrar en un espacio de nombres de 24 caracteres. Se los llamaban nombres del host. Host es una palabra que quiere decir computadora, y al mapear estos nombres con direcciones de IP para que los seres humanos pudieran usar los nombres, un programa, una computadora, una enrutadora usarán números, esto se llama resolución de nombres. Y la resolución de nombres, antes de que sea el DNS utilizaba un archivo host. Esto se llamaba host.txt. Tiene la misma función que un formato ligeramente diferente a los archivos modernos de host que se utilizan en Linux. Es un archivo de texto con nombres y direcciones. Nombres de host y su correspondiente dirección de IP. Esto lo mantiene en forma central NIC, centro de información de redes, que tenía un contrato otorgado por el gobierno de los Estados Unidos para que realizara ciertas tareas de

---

administración de la red. Una de esas tareas era mantener el archivo host.

En esa época Internet era mucho más pequeña. Ni siquiera se llamaba Internet. Se llamaba Arpanet. Era un experimento del Departamento de Defensa de los Estados Unidos. En esa época había muchos menos host que hoy en día. Hoy en día hay decenas de miles. En ese momento era razonable mantenerlo todo centralizado. La centralización se hacía por correo electrónico. Un administrador de red agregaba una computadora a la red o la sacaba o le cambiaba el nombre o la dirección de IP, mandaba un correo electrónico a NIC y le decía: Hice esto. Por favor cambien el nombre o la dirección de IP de esta dirección y pongan esta otra. Entonces NIC mantenía la copia maestra de ese archivo host.txt y elaboraba una nueva copia una vez por semana.

Los administradores de red cuando decían que su archivo estaba desactualizado descargaba en un nuevo archivo host.txt. Las cosas no cambiaban tan rápidamente así que no había que actualizar con muchísima frecuencia. Todo se descargaba utilizando FTP.

Es una solución que utilizaba pocos recursos tecnológicos, pero tenía algunos problemas que podían predecirse si uno pensaba en esto por un ratito. Uno de los problemas era la controversia

---

por nombres. Si solo tenemos 24 caracteres para definir el nombre de una computadora y el tamaño de la red crece, cuantos más dispositivos hay en la red, más controversia hay en cuanto los nombres y más difícil es evitar la duplicación. Y para empeorar las cosas, este archivo se mantenía de manera muy simple. En NIC se editaba a través de un editor de textos. No había una base de datos. Simplemente era un archivo de texto, así que no había un buen método para evitar las duplicaciones. De vez en cuando aparecían duplicados. Eso es un problema.

Otro problema obvio era la sincronización. Nunca todas las personas tenían la misma versión de un archivo. Siempre había alguien retrasado. Después el tráfico la carga también era un problema. El archivo empezó a crecer tanto que requería mucho ancho de banda para bajar el archivo. En ese momento las conexiones 64 kb por segundo eran las más rápidas. Esto es antes de que yo creciera, pero me dijeron que en esa época llevaba más tiempo descargar el archivo que lo que llevaba actualizarlo. Uno nunca podía obtener la última versión porque cuando uno había terminado de descargar el archivo anterior, ya había un nuevo archivo que debía descargarse. Entonces claramente mantener esos archivos host a nivel central no era escalable. Había que hacer algo.

Se empezó a hablar a principios de los 80 sobre cómo reemplazar host.txt con dos objetivos. Número 1: Resolver los

---

temas de escalamiento. En segundo lugar, teníamos el tema del routing por correo electrónico. Cuando pensamos en DNS recordamos el primer tema, el tema del escalamiento. Pero no sabemos o no tenemos en cuenta los temas de enrutamiento de correos electrónicos. El resultado es el sistema de nombres de dominio DNS. Este es el resumen del DNS. Sí tenemos una sola diapositiva para resumir el DNS, es esta. Básicamente DNS es una base de datos distribuida. En esta base de datos, los datos se mantienen a nivel local. Entonces todo el mundo tiene su propia parte de la base de datos y mantiene sus propios datos, pero estos datos están disponibles a nivel global porque la base de datos está distribuida a todo el mundo, en toda la Internet. Entonces uno mantiene sus propios datos a nivel local, pero puede ver los datos de todos los demás. Los archivos del DNS tienen un modelo de clientes de servidor. Los resolutores son del lado del cliente. Los resolutores envían consultas. Los servidores de nombres son del lado de los servidores, y lo que hemos hablado aquí es que los servidores responden consultas. Así los resolutores manda las preguntas y los servidores de nombre las responden.

Aquí hay algunas optimizaciones. DNS utiliza memoria caché para mejorar la performance. Esto significa que como estamos hablando de una base de datos distribuida en todo el mundo a la velocidad de la luz es una definida. Entonces cuando hacemos

---

una búsqueda en nuestra base de datos quizás haya que hacer varias consultas cubriendo a todo el mundo ida y vuelta. Esto lleva tiempo. Entonces es muy útil. De hecho, es vital recordar no solamente el último resultado de la búsqueda sino todos los resultados intermedios de esa búsqueda. Eso pasa a la memoria caché y acelera el proceso para la vez siguiente.

DNS también utiliza replicación para ofrecer redundancia y distribución de cargas. ¿Qué quiere decir esto? Como dije antes, cada uno mantiene su copia local de los datos, pero la replicación también es importante. No hay una sola copia. Hay múltiples copias de los datos. De esta manera hay redundancia. Es decir, si solo tuviéramos una copia de los datos y algo les pasara esos datos nadie podría buscar nada. Pero si tenemos múltiples copias, tenemos redundancia. Esto también distribuye la carga. Si muchas personas hacen búsquedas y hay muchas copias, la carga de búsqueda se distribuye entre todas las copias.

Aquí tenemos los diferentes componentes del DNS en una sola diapositiva. Vamos hablar de todos estos elementos en detalle más adelante, pero sirve verlos todos juntos para entender a donde apuntamos. Empezamos en el extremo inferior izquierdo y después vamos avanzando. En el extremo inferior izquierdo tenemos un dispositivo conectado a Internet, un teléfono, pero todos los dispositivos conectados a Internet deben transformar

---

los nombres en direcciones. Todos los nombres o todos los dispositivos que utilizan DNS tienen un cliente llamado resolutor stub. El resolutor stub puente entre una aplicación, como un navegador, y todo el resto del DNS.

Entonces el resolutor stub toma el pedido de una aplicación, por ejemplo de transformar un nombre en una dirección, y después lo transforma en una consulta al DNS que envía hacia delante y lo manda al resolutor recursivo. Este resolutor acepta un pedido de una aplicación, lo transforma en una consulta del DNS y envía esa consulta y espera una respuesta. El resolutor, del otro lado, tiene una función más compleja. Sabe cómo ponerse en contacto con diferentes servidores de nombre autoritativos, que es donde están los datos del DNS, los datos que se buscan, que se consultan. El resolutor recursivo quizás tenga que ponerse en contacto con muchos servidores para encontrar respuesta. Quizá se ponga en contacto con un servidor y este servidor autoritativo diga: Yo no tengo la respuesta que usted busca, pero puedo referir a otro servidor. Y este segundo servidor dice: Yo tampoco tengo la respuesta, pero lo puedo derivar a un tercero que está más cerca. Entonces el resolutor recursivo es inteligente y va navegando por todos estos servidores hasta encontrar respuesta. Si buscamos dentro del resolutor recursivo, vemos que tiene un servidor de nombre y un resolutor. Los servidores de nombre responden consultas.

---

Entonces responde las consultas que les presenta el resolutor stub, pero después, del lado del resolutor, los resolutores envían consultas para los servidores de nombre autoritativos.

También ya vimos la parte de caché. Todo lo que recibe el resolutor, todas las respuestas que recibe del servidor autoritativo, las pone en la memoria caché para poder utilizar esa información más adelante para responder consultas. Eso es el ecosistema de DNS descrito a nivel muy general.

Ahora voy a definir algunos términos importantes de DNS. En primer lugar, tenemos lo que llamamos el espacio de nombres. Ya dije que el DNS es una base de datos distribuida y la estructura de esa base de datos es lo que llamamos el espacio de nombres. Quizás ustedes conozcan esto cuando hablo de base de datos. Quizás ustedes piensen una base de datos relacional. Y la estructura de esta base de datos es que incluyen tablas, y las tablas tienen hileras, y dentro de cada hilera tenemos columnas de información. Así es como están estructuradas las bases de datos relacionales. Las bases de datos de DNS son totalmente diferentes. Es lo que llamamos un árbol invertido. Aquí tengo un ejemplo de una parte muy pequeña del espacio de nombres DNS. En un árbol invertido la raíz está arriba y las ramas crecen hacia abajo. Este es un árbol informático. No su sorprendente que los árboles informáticos estén al revés. Raíz arriba, ramas abajo. Todos los nodos de este

---

árbol, todos los cuadrados, tienen un hombre, una etiqueta. El nodo raíz es muy especial. El nodo raíz que está arriba no tiene una etiqueta o su etiqueta es la etiqueta vacía. No tienen nada. A veces es algo representado como un punto entre comillas. Simplemente para mostrar que no hay nada allí, que está vacío. Ese es el espacio de nombres.

Muchas veces llamamos a estos nodos según su posición de relación en la raíz. La raíz está arriba y debajo de la raíz a la izquierda tenemos los nodos de alto nivel. Después tenemos los nodos de segundo nivel, etc. Vamos descendiendo en el espacio de nombres. A veces a los nodos los llamamos utilizando nombres de familia: Padre e hijos. Aquí la raíz sería el padre de .com, y .com sería el padre de ejemplo, pero ejemplo el hijo de .com. Hablamos de relaciones padres-hijos. Cada una de estas etiquetas tiene una serie de caracteres que pueden utilizarse. Los llamamos LDH: Letras, dígitos y guion. Son los únicos caracteres que se pueden utilizar en los nombres del DNS. La longitud máxima de una etiqueta es 63 caracteres. Además, los nombres de las etiquetas no importa si se escriben con mayúscula o minúscula. Podemos mezclarlas y son equivalentes.

Todos estos nodos tienen un nombre de dominio, y el objetivo del nombre de dominio es saber dónde está el nodo en el espacio de nombres. La definición de nombres de dominio es

---

muy clara. Empezamos con un nodo y vamos avanzando hacia la raíz y vamos escribiendo los nombres de las etiquetas y en medio ponemos el punto. Aquí vemos el nodo que tenemos arriba. Escribimos www punto y después vamos a su padre, que es ejemplo. Escribimos ejemplo y ponemos punto, después com y otra vez punto, y llegamos a la raíz.

Hay algunos nombres de dominio especiales, que llamamos nombres de dominio totalmente calificados. Los nombres de dominio inequívocos nos dicen exactamente dónde está el nodo en el espacio de nombres. Un nombre de dominio inequívoco siempre termina con un punto, y el punto es lo que separa el TLD, en este caso .com, y la raíz. Llegamos al dominio de alto nivel, ponemos un punto y después ponemos la etiqueta de la raíz, que no existe en realidad. Entonces esta etiqueta también era un punto. Esto seguramente le sonará conocido. Seguramente conocerán archivos Windows o Unix. Es un ejemplo de una estructura de datos que también se puede representar como árbol invertido. En el sistema de archivos los nodos representan archivos o directorios. En lugar de un nombre de dominio tenemos un nombre de vía o trayectoria que nos dice exactamente dónde está el archivo o directorio en el sistema de archivos, así como el nombre de dominio nos dice dónde está un nodo en el espacio de nombres.

---

Tenemos otro importante también aquí. Dominios. La definición de dominio es muy simple. Es un nodo en un espacio de nombres y todo lo que está por debajo. Por ejemplo, tengo aquí el dominio .com. Punto com sería el nodo y todo lo que está por debajo. Aquí les estoy mostrando tres nombres .com. En realidad, hay 131 millones. O sea que hay unos cuantos que no están en la diapositiva, pero el dominio .com es muy grande. Es todo lo que está por debajo de .com. Todos los nombres de dominio que terminan con .com están en el espacio de nombres .com.

Ahora voy a comparar esto con el término zona. Es un término muy importante porque se menciona muchas veces y debemos entender qué es. Recuerden que llegamos aquí y tenemos el DNS porque necesitamos una administración distribuida. El mantenimiento central de la información con respecto a los nombres de host y de direcciones no funcionaba realmente. Tuvimos que buscar un sistema distribuido para que todo el mundo mantuviera su información sobre sus direcciones y de nombres. Por lo tanto, el espacio de nombres está dividido para que pueda ver esta administración distribuida. Estas divisiones administrativas se llaman zonas. Hay diferentes zonas y cada una puede jugar dentro de su zona. Dentro de su zona se pueden hacer todos los cambios posibles sin afectar a nadie. Cada uno maneja su propia zona, la mantiene. Y las zonas son creadas por

---

delegación. Se delega desde el punto más en el espacio de nombres hasta el más bajo. Entonces a la zona de delegación le llamamos padre y la zona que se crea es el hijo. Este proceso empieza de la raíz y avanza hacia abajo.

Les voy a dar un ejemplo. Aquí tenemos el espacio de nombres otra vez. Si miran y lo ven así, no tenemos suficiente información para ver dónde están los límites de las zonas. Entonces el espacio de nombres, si simplemente lo miramos, no sabemos cómo está dividido a los fines administrativos. Les voy a dar un ejemplo. Imaginen que están mirando desde un satélite o una estación espacial y ven América del Norte. Si ven América del Norte desde arriba, no saben exactamente que en realidad hay tres países allí: Canadá, Estados Unidos y México. No lo saben desde el satélite. Pueden ver América del Norte todo el día y nunca van a saber cuáles son los límites administrativos y políticos porque no se ven. Necesitaríamos información adicional. Lo mismo pasa con las zonas. Vemos el espacio de nombres, pero a menos que sepamos dónde se da la delegación no sabemos dónde están los límites de las zonas.

Les voy a marcar algunos límites de zonas. Yo sé dónde están las delegaciones en el espacio de nombres, por eso puedo mostrarles los límites de las zonas. Arriba del todo tenemos la zona raíz. La zona raíz delega en las zonas que están en primer nivel. Decimos que una zona tiene información delegación que

---

apunta a la zona en la que delega. Este proceso de delegación continúa hacia abajo a lo largo del espacio de nombres. La zona raíz delega en .com en este ejemplo. Después .com delega en este caso en ejemplo, example.com, bar.com y food.com. Entonces si pensamos en el tamaño de estas zonas... Bueno, empecemos con la zona raíz.

La última vez que busqué había 1543 zonas de dominio de alto nivel. Entonces la zona raíz es bastante pequeña. Solo necesita tener información de delegación sobre 1543 zonas que están por debajo de ella. Pero pasemos a .com. Punto com, por otro lado, si vamos el material que incluyó VeriSign en los bolsas que recibieron cuando se registraron, el último informe dice que hay 131 millones de nombres en .com. La zona .com es la más grande que existe. Entonces debe tener información que va delegando a los 153 millones de zonas que están por debajo de .com. La delegación puede seguir por debajo del segundo nivel. No lo vemos a la diapositiva, pero puede haber una delegación de segundo al tercer nivel avanzando en el espacio de nombres.

Recordarán que hace un rato dije que DNS utiliza la replicación para la redundancia y mejorar el desempeño. Esa replicación es de lo que estamos hablando ahora. Recuerden que los servidores de nombre responden consultas. Decimos que un servidor de nombre es autorizado para una zona si tiene el conocimiento completo sobre esa zona, y un servidor de

---

nombres autoritativo puede responder consultas definitivamente si alguien las hace. Pues decir: Me preguntaron algo acerca de la zona, acá está la información. O puede decir: Usted pidió algo, hice una consulta sobre un nombre que no existe, así que yo le puedo decir que ese nombre no existe en esta zona las zonas deberían tener múltiples servidores autoritativos. Como dije, brinda redundancia y distribuye la carga. Entonces cada zona tiene que tener por lo menos un servidor de nombres autoritativo, y en la realidad va a tener más de uno. Es una mejor práctica tener por lo menos dos servidores autoritativos porque con uno solo si hay un problema nadie puede seguir haciendo consultas en la zona. Si vamos a tener múltiples servidores autoritativos es necesario mantener la información sincronizada. La idea es que la información acerca de una zona deberá ser la misma en todos los servidores autoritativos. ¿Cómo lo hacemos? La buena idea es que el protocolo tiene una forma incorporada. La sincronización de los datos de zona en los servidores autoritativos está incorporada. Hay un proceso que se llama transferencia de zona que permite moverse en la zona. Entonces designa servidor autoritativo y esto nos permite hacer cambios en la zona.

Después tenemos otros servidores autoritativos que se llaman secundarios o esclavos. Se refieren a lo mismo. Estos servidores secundarios carga los datos de las del primario. Se contactan

---

con el primario y hacen lo que se llama transferencia de zona y copia en la zona del primario al secundario. Entonces es importante señalar que todos los servidores autoritativos son iguales. Tiene los mismos datos. La única diferencia entre el primario y el secundario es de donde obtienen los datos. El primario carga los datos de la zona desde el disco, y el secundario carga los datos de la zona desde el primario. Pero los datos que están en el primario y secundario son los mismos. Por supuesto puede ser que por un momento no estén sincronizados porque si se hace un cambio en el primario, hay una demora, pero luego se cambió se propaga el secundario y así se entroniza la información. Eso está muy bien y está incorporado en el protocolo del DNS. Por lo tanto, ustedes no tienen que preocuparse por mantener los servidores de nombres sincronizados porque esto simplemente ocurre. No hace falta que lo hagan ustedes.

Ahora vamos a pasar al próximo nivel. Hablamos de las zonas ahora. Hablemos de lo que ocurre adentro de la zona, acerca de los datos adentro de la zona. Recuerden que todos los nodos en el espacio de nombres, cada una de esas casillas, tiene un nombre de dominio. La forma de verlo es la siguiente. Un nombre de dominio determinado puede tener diferentes clases de información, diferentes clases de datos asociados. La clase más común es una dirección IP. Pues entonces vincular una

---

dirección IP con un nombre de dominio. Esta clase de información que acompaña a un nombre de dominio se llama registro de recursos. Hay diferentes tipos de registros de recursos que almacenan diferentes clases de datos. Los registros de recursos más comunes son los que almacenan direcciones IP, IPv4 IPv6. Hay diferentes tipos de registros.

Pero estos tipos de registros de recursos también almacenan otro tipo de datos. Entonces una zona consta simplemente de una serie de registros de recursos, y todos los registros de recursos de una zona se ponen en un archivo y ese archivo se llama archivo de zona. Hay un archivo de zona para cada zona y nunca se mezclan los registros de múltiples zonas en un archivo. Entonces una zona no es más que un conjunto de todos los registros de recursos. Les voy a mostrar cómo son estos registros de recursos. De hecho, hay una forma estándar de escribirlos con texto. Los registros de recursos tienen 5 campos. No hace falta que veamos todos. El mensaje importante acá es que los registros de recursos pueden ser diferentes y transfieren o transportan datos diferentes. Estos son los tipos de recursos más comunes. Ya dije que tenemos un tipo de registro de recurso para IPv4 e IPv6. Se llama registro de dirección. Después tenemos un tipo de registro que almacena direcciones IPv6. A este lo llamamos cuádruple A o 4 A.

---

Y después tenemos otros tipos que se mencionan aquí. Esta lista es la que incluye los tipos de registros de recursos más comunes. Pero hay muchas otras clases de registros de recursos. La última vez que me fijé había 84 clases diferentes. Hay un registro de la IANA, que tiene el nombre que ustedes ven en pantalla. No lo voy a decir. Pero si entran en ese sitio web se van a encontrar con esto. Entonces sobre la base del tamaño del campo de registro de recurso podemos llegar hasta 65.000. Ahora si pensamos en algo nuevo que queremos almacenar en el DNS, podemos entrar en ITF, convencer a la gente de que nuestra idea tiene que convertirse en un nuevo tipo de DNS. Y en ese caso, se lo va a crear, se le va a asignar una clase en este registro, íbamos a poder poner cosas nuevas en el DNS. La gente continuamente hace estas cosas. Bueno, no con tanta frecuencia. Solo hay 84. Pero la gente va creando cosas nuevas que se ponen en el DNS y así se almacenan nuevas clases de datos. Pero de lejos la clase de datos más comunes en el DNS son las direcciones. El uso más común de DNS es para mapear nombres de dominios con direcciones. Acá tenemos estos dos tipos de registro que les mostré antes.

Acá tenemos la representación en texto de lo que es un registro de dirección y un registro cuádruple A. Tenemos ejemplo.com. Después tenemos el tipo, que en el primer caso es A por dirección, y después tenemos la dirección propiamente dicha.

---

Este es un ejemplo de un registro de recursos que va a estar en el archivo zona para el caso de ejemplo.com. Acá tenemos que ejemplo.com tiene esta dirección IP. Y abajo tenemos un registro cuádruple. Tenemos registro.com, cuádruple A, y esta es la dirección de IPv6. La mayor parte de DNS tiene registros A y registros de 4 A porque esta es la forma más común de mapear los nombres con direcciones.

Ahora hoy otro tipo de registros y lo interesante es que la mayoría de estas clases son utilizadas por gente que consume información del DNS, qué busca información en el DNS porque necesita hacer cosas cómo conectarse con un servidor web, el navegador necesita buscar un nombre y una dirección. Pero a veces hay algunas clases que son utilizadas por el DNS propiamente dicho, y un ejemplo de esto es el registro del DNS y el registro SOA. Y éstas son cosas que a nadie le importan salvo al DNS propiamente dicho. Lo interesante de esto es que estas son clases igual que las otras clases. A mí me gusta ver esto y usar la analogía de un depósito. Digamos que alquilamos un depósito porque queremos almacenar un montón de cosas ahí. No traemos el camión y empezamos a tirar las cosas en el depósito. Para poder utilizar el depósito, necesitamos tener estanterías. Una vez que tenemos estanterías podemos sacar las cajas del camión, si es que realmente nos importa lo que hay en las cajas. Y ponemos las cajas en las estanterías. Entonces el

---

depósito no sirve de nada si no tiene estanterías. El DNS es similar. Podemos pensar en los registros NS y SOA como las estanterías. Necesitamos eso para que el DNS funcione. Ahora nadie fuera del DNS le importa esto. A los que están fuera del DNS les importan las otras clases de registro, como A o cuádruple A.

Hablemos acerca de estos registros NS. Así es como decimos cuáles son los servidores de nombre autoritativos para una zona. Este ejemplo acá muestra dos registros NS. Estos registros NS nos dice lo siguiente. Nos dicen que ejemplo.com, esa zona, tiene dos servidores de nombres alternativos. Uno es NS1 y el otro, NS2. A la izquierda tenemos el nombre de la zona y a la derecha, el nombre del servidor de nombres. Ahora, los registros NS son un poco complejos porque aparecen en dos lugares en realidad. Aparecen en la zona padre y en la zona hijo. En este casillero tengo toda la lista de registros NS para la zona .com. Hay 13 servidores de nombres autoritativo para esta zona y se llaman, como ustedes podrán ver, A.gTLD-SERVERS.NET, B, etc. Esta lista de registros, estos 13 registros aparecen en 2 lugares. Vamos a ir acá. Aparecen en la zona raíz. Este caso la zona raíz es la zona padre y estos registros NS en la zona padre son los que hacen la delegación. Esto es lo que le dice al resto de DNS que por abajo de la zona raíz tenemos la zona .com. Y la lista de registros NS vuelve aparecer en la zona que en este caso se

---

llama .com. Los registros NS.com aparecen en la raíz, en el padre y luego aparecen en .com nuevamente. Cuando hablamos acerca de cómo buscar datos en DNS, verán cuán importante es que estos registros DNS estén en la zona padre.

Les voy a decir ahora que la forma en que funciona la resolución en DNS es comenzando a la raíz y siguiendo estos punteros de delegación, buscando los registros NS. Entonces si estamos buscando algo que está por debajo de .com, empezamos a la zona raíz. En la zona raíz vemos la delegación a .com y vemos al servidor de nombres de .com y buscamos la delegación que está por debajo, y así continuamos. Después vamos a volver a este tema.

Los registros NS incluyen solo nombres. Entonces si buscamos el ejemplo que yo les di antes, vemos que ejemplo.com, uno de los servidores de nombres es NS1.EJEMPLO.COM. Ahora, eso solo no alcanza porque necesitamos luego la dirección IP de NS1.EJEMPLO.COM para poder contactarlos. Entonces la información de delegación también debe incluir registros de direcciones en algunos casos. A esos los llamamos registros glue o de adherencia. Es el registro de dirección para un servidor de nombre.

Hay otro registro que está en todas las zonas, que se llama registro SOA. No voy a hablar mucho acerca de este registro.

---

Solo quiero señalar que existe y que hay un registro SOA por zona. Está en la parte superior en lo que llamamos el ápice de la zona. La mayor parte de los valores están relacionados con esa transferencia de la que hablé antes. Les dicen a los servidores autoritativos cómo sincronizar, con qué frecuencia sincronizar la zona.

Volvamos ahora al segundo objetivo del DNS. El segundo problema que debe resolver el DNS: El enrutamiento del correo. El problema que debía resolver el DNS era como entregar correo sobre la base de una dirección de mail. En el pasado, antes de que se hiciera el DNS teníamos direcciones de email que era un usuario, arroba y el nombre del host. Era uno de esos nombres que tenía como máximo 24 caracteres. Eso era una dirección de email. Antes de que se hiciera el DNS, eso significaba que mi email iba a ir al nombre del host, que estaba a la derecha de la dirección de email. No había forma de decir mi dirección de email es una, pero el mensaje que enviarlo a otra máquina. No. Si mi dirección era MATT@FOO, entonces tenía que llegar a ese servidor. Uno de los objetivos del DNS era separar eso. Poder decir esta es mi dirección de email y esta es la dirección a la que debe ir el mail, pero tiene que ir a otro lugar. Entonces el DNS ofrece la flexibilidad necesaria. Tenemos un registro que se llama el registro de intercambio de email, que nos dice adónde de ver el mail. Este es un ejemplo. Entonces estos registros MX,

---

así se llaman, para el caso de ejemplo.com dicen a dónde debe ir el mail. Entonces en el caso de cualquier nombre de usuario@ejemplo.com, el registro MX dice que debe ir a una máquina que se llama mail.ejemplo.com. Hay un valor de preferencia que es el número 10 y número 20. Esto va un poco en contra de la intuición porque hay uno que es más deseable que otro. Entonces estos dos registros lo que dicen es que en el caso de cualquier dirección mail@ejemplo.com, hay que enviarlo a mail.ejemplo.com. Pero si no se puede hacer eso por algún motivo, entonces hay que tratar de enviarlo a mail-backup.ejemplo.com.

Entonces un servidor de mail que va a enviar mi email tiene que poder buscar los registros MX para los nombres de dominio. Hay una unión muy estrecha entre DNS y el mail basado en SMTP. Entonces el servidor de email contiene un mensaje que entregar, busca el registro MX para la dirección de email, y así es como sabe a dónde enviar el mensaje.

Ahora estuvimos hablando de cómo mapear los nombres con las direcciones IP, que es una tarea importante, pero a veces queremos hacer lo contrario. El mapeo de nombre a IP se llama mapeo hacia delante. ¿Qué pasa si quisiéramos hacer un mapeo de IP a nombre? En algunos casos esto es necesario. Imagínense por ejemplo que hay una red que quiere detectar un problema y quiere ver cuáles son los enrutadores entre nosotros y la

---

dirección a la que estamos tratando de llegar. Cuando buscamos la dirección de IP a través de todos los routers, quizás nos interese la dirección IP, pero también nos interesa saber dónde está esa dirección IP, quién la opera, cómo se llama. Es un ejemplo de un mapeo inverso. Se trata de tomar una dirección IP y encontrar el nombre correspondiente.

Ahora piensa en una tabla de host. Si tenemos solamente un archivo con una lista de nombres y direcciones IP y queremos hacer un mapeo hacia delante, es relativamente fácil. Tenemos un nombre, lo queremos vapear con una dirección IP, buscamos en ese archivo hasta encontrar el nombre, y ahí tenemos la dirección IP. ¿Qué pasa si quisiéramos hacer el mapeo reverso? Es igual de fácil. Buscamos en el archivo el IP y al lado tenemos el nombre. Entonces eso funciona muy bien con la tabla de host. ¿Qué hacemos con DNS? Voy a retroceder para ver el ejemplo del espacio de nombres. ¿Cómo está estructurado este espacio de nombres? De una forma que hace que sea muy fácil encontrar los nombres de dominio, pero en este caso es imposible buscar direcciones IP.

Si yo quiero buscar un nombre de dominio, digamos [www.ejemplo.com](http://www.ejemplo.com), pueden ver que empiezo en la raíz y voy bajando hasta llegar a [www](http://www). ¿Qué pasa si empiezo con la dirección IP? La respuesta es que no se puede hacer esto en DNS de la forma que les mostré hasta ahora porque no podemos

---

buscar una dirección IP. Entonces lo que tenía que pasar era que había que encontrar una forma de convertir las direcciones IP en direcciones de dominio para después poder buscarlas como nombres de dominio. Eso es exactamente lo que tenemos. Entonces tenemos otro tipo de registros que se llaman PTR, por pointer o puntero. Las direcciones IP van a nombres de dominio especiales, en donde ponemos registros PTR que nos permiten buscar direcciones IP como nombres de dominios y encontrar el nombre correspondiente. Entonces las direcciones IPv4 van con un nombre de dominio que se llaman IN-ADDR punto arpa e IPv6 va con IPv6 punto arpa.

Les voy a mostrar un ejemplo. Aquí tenemos el árbol del espacio de nombres. Les estoy mostrando una parte nueva que quizás ustedes no sepan que existe. Aquí a la derecha tenemos ejemplo.com, que ya conocemos. Pero después acá tenemos el dominio arpa y el dominio IN-ADDR. Aquí es donde tendríamos el registro PTR de ejemplo.com. Entonces a la derecha tenemos un registro de una dirección que dice: La dirección IP de .com es 192.0.2.7. Entonces buscamos ese ejemplo sabemos cuál es la dirección. ¿Qué pasa si yo quisiera saber cuál es el nombre del nombre de dominio que se corresponde con esa dirección 192.0.2.7? En ese caso tengo que convertir esa dirección IP en un nombre de dominio. ¿Qué hago? Tomo esa dirección IP, la doy vuelta si le agrego IN-ADDR y arpa y así busco el registro PTR.

---

La única razón por la cual funciona es porque todos se entienden y conocen cuáles son las reglas. Todos saben o entienden lo que yo les acabo de describir. Ahora si yo tengo un espacio de nombres asignado a una persona, los RIR, los registros regionales de Internet, cooperan para administrar este dominio. Así podemos tener una zona que corresponda con una dirección de IP delegada.

Digamos por ejemplo que a mí me asignaron 192.0.2.7/24, todo lo que empieza con 192.0.2. En ese caso yo voy a tener ese dominio de arpa 192.0.2 asignado a mí mismo. Si quiero que la gente pueda hacer el mapeo inverso va a ocurrir lo mismo. Es un poco extraño, pero funciona. Seguramente la mayoría de las personas piensan que el mapeo inverso no es tan importante como el mapeo hacia delante. El mapeo de adelante es lo que nos permite tipear un nombre de dominio en un navegador y llegar al sitio web. El mapeo inverso se hace con fines de diagnóstico o de detección de problemas. En general una persona normal no hace eso, solamente los ingenieros de sistemas y los especialistas se ocupan del mapeo inverso.

Como dije antes, hay muchas otras clases de registro de investigación. Este es un ejemplo de algunos casos más, para que sepan cuáles son los otros tipos de datos que la gente puso en el DNS. Acá tenemos un ejemplo de lo que es un archivo de zona para una zona muy chica. Es un archivo de zona para una

---

zona example.com. Y no hablé de todos estos registros en detalle, pero esta es una zona pequeña, parecida a lo que tienen casi todas las zonas en Internet porque si lo pensamos la mayoría de los nombres de dominio en Internet probablemente hay que hacer dos cosas. Tenemos un servidor web, un sitio web, ley de recibir correos electrónicos también. Obviamente hay dominios que hacen muchas otras cosas, tienen muchos nombres, pero la mayoría de los nombres y muchos nombres de dominio. Por ejemplo, mi nombre de dominio persona que utilizo para el correo electrónico, quiero recibir correo electrónico y tengo un pequeño sitio web y basta. Entonces mi zona personal, la de mi nombre de dominio, se parecerá a la que vemos acá. Es lo único que necesitamos en mi caso. Tenemos una dirección de IP ejemplo.com, que es donde está el servidor web, así que pueden ver el archivo de zona y pueden suponer que 192.0.2.7 es la dirección de ese sitio web. Después ven unos registros adicionales que dicen dónde hay que enviar el correo electrónico que queremos que llegue a ejemplo.com.

Ahora quiero hablar del proceso de resolución. Y es así como se buscan cosas en DNS. Los componentes del DNS que les mostré al principio de la sesión, los resolutores stub, los recursivos y los servidores de nombres autoritativos, todos cooperan para buscar datos en el espacio de nombres. Lo importante aquí es que una consulta del DNS siempre tiene tres parámetros: Un

---

nombre de dominio, como `www.example.com`; el tipo de datos que buscamos, en este caso A de direcciones; y también la clase de la cual no hablé. Clase es una forma que se podría utilizar para llevar el DNS a otro tipo de redes. Pero no se utilizó de hecho. Pero está en DNS así que allí quedó. En este caso, clase siempre va a ser clase internet y no hay que preocuparse por eso. En este caso es simplemente el nombre de dominio y el tipo de datos lo que son importantes. Entonces si hacemos una consulta de DNS y hacemos una pregunta al servidor de nombre, siempre hay que especificar el nombre de dominio y el tipo.

Hay dos tipos de consultas. Los resolutores stub son nuestro teléfono, la tostadora, la heladera, todo lo que se conecta Internet y debe transformar nombres en direcciones o en otro tipo de información. Todos esos dispositivos tienen un resolutor stub. Hilo resolutores stub mandan lo que se llama consultas recursivas. Estas consultas son la señal para que el resolutor recursivo escuche. Bueno, soy un resolutor stub y necesito que me dé una respuesta o un mensaje de error. No me sirve una respuesta parcial; necesito una respuesta completa. Los resolutores recursivos, por otro lado, son más inteligentes y pueden aceptar respuestas parciales que se llaman derivaciones. Así que mandan una consulta que indica que están dispuestos a aceptar respuestas parciales, derivaciones.

---

Entonces como ya dije antes, si buscamos algo en DNS empezamos en la zona raíz y vamos avanzando.

Seguimos la delegación a través de los punteros de delegación. Entonces hay servidores autoritativos para la zona raíz que tienen toda la información de la zona raíz. Y esto significa autoritativo. A esto lo llamamos servidores de nombre raíz. Cuando empezamos la resolución en la zona raíz, tenemos que poder conectar un servidor de nombres raíz. ¿Cómo encontramos cuáles son los servidores de nombre raíz? La respuesta es que es necesario configurarlos. No hay forma de descubrirlos. Hay que configurarlos en todos los servidores de nombres recursivos. Esto es diferente de otros parámetros de red. Por ejemplo, cuando mi teléfono se conectó con la red wifi de la ICANN aquí, del centro de conferencias, utilizó un protocolo especial de configuración dinámica. Mi teléfono le dijo a la red: Soy un dispositivo nuevo en esta red; necesito una dirección IP. Y la red dijo: Mi teléfono no sabía nada de esta red. Y la red le dijo: Muy bien; esta es tu dirección de IP y estos son otros parámetros de configuración que tienes que conocer. También esto incluye la dirección IP del servidor de nombres recursivos. De esta manera, quizás un dispositivo no sabe nada y la red le da toda la información que necesita. Esto no sucede con los servidores de nombres recursivos. No podemos activar un servidor de este tipo sin configuración. El servidor tiene que

---

saber cuáles son los servidores de nombre raíz y cuáles son las direcciones de IP. Entonces hay un archivo especial que necesitan todos los nombres de nombre recursivos que tienen los nombres y direcciones de IP de los servidores de nombre raíz.

La buena noticia es que ustedes instalan un servidor de nombres recursivo en un equipo Linux, alguien ya hizo esto de poner todo el software y los servidores con la información del archivo de raíz. Algunos resolutores recursivos también tienen las direcciones de IP y los nombres del servidor raíz, pero ese es el URL dónde pueden buscar el archivo. Este es el aspecto que tienen. Hay 13 servidores de nombre raíz. Son 13 servidores autoritativos para la zona raíz. A la izquierda tenemos la zona raíz y después 13 registros de NS, que son los nombres de los servidores de nombres, que se llaman A.ROOT-SERVERS.NET, B, etc. Tienen esos nombres. Después pueden ver por debajo las direcciones IPv4 de cada servidor y las direcciones IPv6. Entonces todos los servidores de nombre raíz tienen una dirección de IPv4 y una de IPv6. Los servidores de nombres recursivos para hacer una resolución necesitan esta información. Deben conocer los nombres y direcciones IP de los servidores de nombre raíz.

Entonces ahora hacemos un pequeño desvío y vamos a hablar de la zona raíz y de cómo llega la información a la zona raíz.

Recuerden. ¿Qué tenemos en la zona raíz? Tenemos información sobre las zonas de dominio de alto nivel. Tenemos registros NS para los TLD. Es complejo administrar la zona raíz. Aquí trabajan juntas dos organizaciones. La ICANN es el operador de las funciones de la IANA y la PTI maneja ese rol. Y VeriSign es la otra organización y son los mantenedores de la zona raíz. Esta forma de trabajar data de principios de los 90 cuando VeriSign era una empresa que tenía otro nombre, Network Solutions, que fue comprada por VeriSign en el año 2000. El operador de las funciones de la IANA era llevado a cabo por la Universidad de California del Sur, antes de que se creara la ICANN. Entonces esto es un acuerdo muy antiguo. Es un poco complejo esta forma de trabajar, pero así es como funciona la zona raíz. Entonces esas dos organizaciones, la ICANN y VeriSign, cooperan para poner los datos en la zona raíz, para crear el archivo de la zona raíz. Después necesitamos servidores autoritativos para la zona raíz. Hay 12 organizaciones que operan los servidores autoritativos. Es poco común. En la mayoría de las zonas hay una organización que opera todos los servidores autoritativos.

Tomemos como ejemplo. com. Yo trabajaba en VeriSign así que sé cómo funciona eso. VeriSign opera todos los servidores autoritativos de .com. Tenemos otra zona como por ejemplo... Bueno, muchas empresas hacen lo siguiente. Operan algunos

---

servidores autoritativos por si mismas o tercerizan esa actividad en un tercero o quizás en múltiples terceros para tener redundancia. Pero no hay 12 organizaciones. Eso es poco común. Aquí tenemos las letras de los 13 servidores de zona raíz. Se llaman A.ROOT-SERVERS.NET. En general lo llaman raíz A, como abreviatura. Y estas son las organizaciones que operan esos servidores. Es un grupo interesante de organizaciones que no tiene nada en común, solo que operan o servidores de zona raíz. Pueden ver empresas comerciales, instituciones educativas, organizaciones sin fines de lucro, ISP, ciertos departamentos del Gobierno de los Estados Unidos. Un poco de todo. Una vez más esto se retrotrae a hace muchos años. Estamos hablando de algo que pasó hace 20 años.

Esta lista de operadores se ha mantenido estática, sin cambios durante 20 años. Y hay toda una serie de temas complejos que tienen que ver con esto que no les puedo explicar con detalle que, por falta de tiempo, pero tenemos 13 servidores raíz y 2 organizaciones. Son 12 porque VeriSign opera dos, el grupo A y el grupo J. Estos son los servidores de raíz y los operadores de esos servidores. Si quieren averiguar más información sobre esto pueden entrar al sitio web [ROOT-SERVERS.ORG](http://ROOT-SERVERS.ORG) y les va a decir dónde están los servidores raíz y les va a dar información adicional.

---

Después quiero mostrarles a nivel muy general cómo se hacen los cambios en la zona raíz. Toda la información en la zona raíz tiene que ver con los TLD. Entonces si un administrador TLD quiere hacer un cambio, cargar un servidor autoritativo para su TLD o sacar un servidor autoritativo o cambiar el nombre o dirección IP de un servidor, envía un cambio al operador de las funciones de la IANA, que es la PTI, que maneja la ICANN, y la PTI hace varias verificaciones y controles y actualiza la zona raíz y manda esta información al encargado de mantenimiento de la zona raíz, que es VeriSign. VeriSign hace otras verificaciones, actualizar su base de datos de la zona raíz, crea un archivo de la zona raíz y lo difunde. Después los 13 servidores raíz descargar ese archivo y lo exponen. Es una descripción muy muy general del proceso. Se lo estoy explicando para que vean como las organizaciones que operan para que esto funcione, pero el proceso es bastante más complejo de lo que vemos aquí. Es un pequeño desvío que tomamos hacia la zona raíz.

Hablemos ahora de cómo funciona la resolución. Supongamos que tenemos un teléfono, abajo a la izquierda, y alguien abrió el navegador en el teléfono y escribe `www.example.com`. Entonces el navegador web llama al resolutor stub, que es un código simple. Aquí vemos que es una llamada de una API, una interfaz de programación de aplicaciones. Es un programa pequeño que se llama resolutor stub. A veces es simplemente una función y

---

dice: Necesito la dirección de `www.ejemplo.com`. Entonces al resolutor stub es muy simple. Solo conoce la dirección de IP de varios servidores recursivos a los cuales se debe enviar la consulta. Entonces el resolutor stub transforma eso en una consulta y envía esa consulta al DNS, al servidor de nombre recursivo que en este caso es `4.2.2.2`. Es la dirección IP de un servidor de nombres recursivos. Este servidor es un servidor abierto al cual cualquier persona le puede mandar una consulta. Es una buena dirección de IP para usar como ejemplo.

Entonces el resolutor stub pide al resolutor recursivo que nos de la respuesta o la consulta que llegó. Para que el ejemplo sea más interesante, recuerden que los resolutores recursivos tienen una memoria caché, pero supongamos que este resolutor recursivo recién se activó, así que no tiene nada en la memoria caché. Solo conoce los nombres y direcciones IP de los servidores raíz. Entonces el resolutor recursivo toma uno de los servidores raíz, elige uno. Por ejemplo, elige este y le hace la misma pregunta que acabo de recibir del resolutor, que es cuál es la dirección IP de `www.ejemplo.com`. Ahora el servidor raíz no puede responder esa consulta directamente porque no conoce la dirección IP de `www.ejemplo.com`. No sabe nada sobre `ejemplo.com`, pero sí sabe algo sobre `.com` porque la zona raíz tiene la delegación de `.com`. Entonces el servidor raíz puede llevar un desvío a `.com`. Puede decirle: Esto son servidores de

---

nombres y direcciones de IP de .com. Entonces ahora el resolutor recursivo pone eso en la memoria caché para usar esa información en el futuro y después sigue la derivación, elige uno de los servidores .com y le manda la misma consulta. Entonces un servidor .com se llama C.gTLD-SERVERS.NET. Así es como VeriSign llama a los servidores .com. Usa la misma nomenclatura, la misma forma de nombres que utilizan los servidores raíz.

Esos son los nombres de los servidores .com. Entonces el resolutor recursivo eligió uno de esos servidores .com, C.gTLD-SERVERS.NET, y le hace la misma pregunta que recibió del resolutor stub. Y le pregunta entonces: ¿Cuál es la dirección IP de www.ejemplo.com? El servidor .com no conoce la dirección IP de www.ejemplo.com tampoco, pero sí sabe cuál es el servidor de nombre autoritativo y le envía esa lista. Manda una lista hacia dónde puede ir el resolutor recursivo, que lo pone en su memoria caché y si llega esa derivación y manda la misma consulta por tercera vez a uno de los servidores autoritativos, que es NS1.ejemplo.com. Ahora este servidor sí sabe la dirección IP de www.ejemplo.com y le da respuesta al resolutor recursivo que la pone en su caché, la manda al resolutor stub, que se lo pasa a la aplicación, y ahora la aplicación conocer la dirección IP del sitio web y puede ponerse en contacto, bajar una página web, y así sigue la cosa. Ese es el proceso de resolución. Es la

---

versión más simple. Como la mayoría de las cosas que se hacen en DNS, puede ser mucho más complejo, pero esta es la versión más simple. Lo importante aquí es que empezamos de la raíz y vamos bajando, pero no siempre hay que empezar en la raíz porque usamos la memoria caché.

Ya mencioné todos los puntos dónde se hacía un caché. Pero ¿qué pasa si alguien viene e inmediatamente quiere ir a ftp.ejemplo.com? El resolutor stub prepara la consulta DNS y se la manda al resolutor recursivo, pero recuerda todo lo que tiene el resolutor recursivo en la memoria caché, conoce cuáles son los servidores de ejemplo.com, etc. El servidor tiene que la raíz de .com va directamente a ejemplo.com. El servidor que tiene esto recibe la respuesta y se la manda al resolutor stub, que se lo manda a la aplicación. Entonces poner cosas en la memoria caché acelera los procesos. Si no fuera por el uso del caché, todo sería mucho más lento en Internet.

La última diapositiva que tengo da una descripción general de lo que estuvimos hablando. Hasta ahora estuvimos hablando del DNS, pero si pensamos en nombres de dominio, nombres de la ICANN, pensamos en algo más que los servidores autoritativos. Pensamos en todo el panorama que incluye registros, registratarios. Y aquí quiero mostrarles dónde están los otros actores. En el mundo de los nombres tenemos registratarios que se comunican con los registradores a través de un sitio web en

---

general para comprar nombres de dominios y hacer cambios. Y el registrador se comunica con el registro. El registro tiene una base de datos donde están registrados los nombres de dominio y la información pertinente. El registro tiene información en la base de datos a través de un servidor de nombres autoritativo y eso es lo que utilizan los resolutores recursivos cuando hacen la consulta. Espero que esto les haya explicado un poco como encaja lo que estuvimos hablando en todo el cuadro general.

Este es el final de mis diapositivas y nos queda tiempo. Con mucho tiempo voy a responder las preguntas que puedan tener.

**CATHY PETERSEN:** Por favor antes de hacer una pregunta digan su nombre y a qué organización pertenecen. ¿Hay algo quizás del Adobe room?

**MATT LARSON:** Bueno, todos tienen hambre. Yo también tengo hambre.

**MALISA RICHARDS:** Yo tengo una pregunta. Soy becaria de Ghana. ¿Qué criterios utilizan al instalar un servidor de zona la raíz?

**MATT LARSON:** Depende del operador. Cada operador tiene su propia política. Yo no hablé sobre esto, pero todos los servidores raíz hoy por

---

hoy son any cast. Es decir que no hay un solo servidor para una dirección de IP de un servidor raíz. Múltiples servidores de hecho. Esa técnica any cast utiliza el servicio de enrutamiento de Internet que permite tener varios servidores con la misma IP respondiendo desde diferentes puntos de la red. Entonces todos los operadores utilizan técnicas any cast para tener múltiples instancias de su servidor raíz, y los diferentes operadores tienen diferentes políticas. Yo sé que, en el caso de la ICANN la raíz L, cualquier persona podría operar una instancia de raíz L porque es un solo servidor. Lo que pedimos es que se compre el servidor, lo instale en la red y brinde espacio, conectividad y potencia. Nosotros lo operamos desde la ICANN. Diferentes operadores de raíz tienen diferentes políticas.

MALISA RICHARDS:

Una pregunta más. Estoy mirando el sitio web que usted me recomendó y estoy viendo que en el Caribe en particular no hay muchos servidores raíz en comparación con América del Sur, América del Norte. ¿Podría explicarme quizás por qué es así esto?

MATT LARSON:

No sé qué ocurre con los otros a operadores. Podría ser que los operadores de esas redes no se hayan acercado a los operadores de raíz como para que instalen servidores raíz. En el

---

caso de ICANN, la barra es muy baja. Si uno está dispuesto a comprar un servidor, nosotros lo podemos poner en la red de ustedes. Muchas gracias a todos. Adelante. ¿Hay otra pregunta?

**NORMAN WARPUT:** Soy becario de la ICANN. Gracias por la presentación. Tengo una pregunta acerca del proceso de cambios en la zona raíz. ¿Se utilizan también para solicitar dominios superiores, dominios de alto nivel?

**MATT LARSON:** No entendí bien su pregunta.

**NORMAN WARPUT:** ¿El proceso para cambiar el archivo de zona raíz se aplica también a los nombres de dominio de alto nivel con código de país?

**MATT LARSON:** Sí, desde la perspectiva de DNS, un TLD es un TLD. Nosotros tenemos categorías de TLD, como ccTLD, gTLD. Y dentro de gTLD, tenemos patrocinados o no patrocinados. Pero tenemos diferentes categorías desde el punto de vista del DNS. De todas formas, un TLD es un TLD, independientemente de que sea ccTLD, gTLD, o lo que fuera. ¿Alguna otra pregunta?

---

ANDREW FRASER: Quisiera saber dónde se describe cómo se apunta a un servidor recursivo. ¿Es a través del ISP? Porque hay diferentes alternativas para los servidores recursivos.

MATT LARSON: En última instancia está configurado en cada dispositivo. Cuando el dispositivo se conecta con la red, la red tiene que decirnos. Todas las redes nos van a decir, además de la dirección IP, cuál es la dirección IP del servidor recursivo. Cualquiera que operar una red entonces va a tener que tener un servidor recursivo.

ANDREW FRASER: Entonces mi teléfono va a estar preconfigurado por mi proveedor.

MATT LARSON: No preconfigurado, sino que cuando se conecte a la red el proveedor va a decir esta es su dirección IP. Esta es otra configuración incluido el servidor recursivo que ellos operen. Sin embargo, si usted quisiera, podría ir por encima de esto y elegir otro. Podría descartar esto y elegir otro. Elegir un proveedor que sea diferente. Los resolutores abiertos tienen diferentes

---

nombres y son empresas que se dedican a operar servidores recursivos que cualquiera puede utilizar.

La primera empresa que hizo esto, que yo sepa, fue Open DNS. Recuerdo que cuando lo hicieron mi reacción y la de mucha gente fue: ¿Para qué quiere hacer eso? Los servidores recursivos son un servicio central de la red. ¿Para qué tomar algo tan importante de lo que yo dependo para una empresa? ¿Por qué voy a depender un servidor porque está fuera de mi red? Y la propuesta de valor de Open DNS era: Nosotros podemos brindar servicios adicionales, como filtrado de contenidos sobre la base de nombres. Usted puede decirnos: Yo no quiero resolver nombres que correspondan con apostar o con contenido para adultos, etc. Y también pueden hacer cosas como: Nosotros no vamos a resolver los nombres de dominio que sabemos que son sitios que tienen malware o algo que es malo desde el punto de vista de la seguridad.

Entonces yo recuerdo haber pensado que lo que sugería Open DNS era ridículo, pero después Open DNS fue comprado por Cisco por muchos millones de dólares. Así que el que ríe último, ríe mejor. Tenemos Google Public DNS y Google dijo que quería tener esto porque quería tener un servidor recursivo muy confiable para hacer validación de DNSSEC. Se trata de agregar autenticación criptográfica al DNS. Entonces Google cree mucho en DNSSEC y dice: Este es el servicio que ofrecemos, que es

---

gratuito, y de esta forma ustedes van a tener una protección adicional de DNSSEC y hay otros servicios públicos de DNS. VeriSign también tiene uno. Quad 9 es otro. Si uno quiere puede cambiar la configuración del dispositivo y utilizar esos. Muchas gracias.

¿Alguna otra pregunta? Muchas gracias a todos entonces.

CATHY PETERSEN:

Gracias a todos. Gracias, Matt. Gracias a los que hacen la transcripción, a los intérpretes y al equipo técnico. Excelente trabajo. Gracias nuevamente.

**[FIN DE LA TRANSCRIPCIÓN]**