SAN JUAN – Como funciona: Noções básicas do DNS
Segunda-feira, 12 de março de 2018 – 17h às 18h30 AST
ICANN61 | San Juan, Porto Rico

CATHY PETERSEN: Eu dou as boas-vindas a Como Funciona Aspectos Básicos do DNS. Senhor Larson, que é um presidente, a pessoa encarregada do CIO da ICANN. Como podem ver, há poucas pessoas presentes aqui fisicamente. Então se têm um problema, por favor, se aproximem.

MATT LARSON: Vamos começar com essa sessão. Boa tarde para todos. Como Funciona o DNS. Aspectos Básicos. Somos muito poucos aqui, então levante a mão, porque temos muito tempo para analisar o material.

Os endereços de IP são fáceis de entender para as máquinas, mas difíceis para as pessoas entenderem. Por isso vamos falar do DNS. Quando estamos com IPv4 pensamos que talvez os nomes podiam lembrar esses endereços, mas agora passamos ao IPv6. Esses possuem poucos caracteres, são endereços mais extensos, e é praticamente impossível lembrar de endereços IPv6. Mas as pessoas precisam usar nomes, os computadores a os routers também utilizam números e as pessoas nomes.

Quando começaram, os nomes eram simples, tinham uma etiquetagem única. Não tinham dados, não tinham nomes de domínio, porque ainda não tinham sido inventados. Então todos os nomes da internet tinham que entrar em um espaço de nomes de 24 caracteres. E se chamava nome do host. Host é uma palavra que quer dizer computadores, e mapeamento com endereços de IP para que os seres humanos pudessem usar esses nomes, e depois um computador transformasse em números. Isso é a resolução de nomes. A resolução de nomes antes de que existisse o DNS se usava um arquivo host. Isso se chama host.txt. Isso tem a mesma função, mas um formato um pouco diferente dos modelos atuais, como Linux e tal.

Esse é um arquivo de texto com nomes e endereços. Nome de host. Isso é o correspondente endereço de IP. E isso faz com que se mantenha de forma central o NIC. Centro de informação de Redes. Que tem o contrato com os Estados Unidos para administrar tarefas de administração de redes. Uma das tarefas era manter o arquivo host. Nesta época a internet era menor, nem se chama internet, era ARPA net. E era uma experimentação do departamento de defesa dos Estados Unidos. Naquela época existiam menos host do que hoje. Hoje há dezenas de milhares. Naquela época era razoável ter tudo centralizado. E a atualização se realizava através de correio eletrônico a um administrador de rede com computador, e aí se

mudava o nome, ou endereço de IP, e enviava um correio eletrônico para alguém que dizia “fiz isso, por favor, mude o nome ou o endereço de IP desse endereço e coloque nesse outro.” Então NIC mantinha a cópia mestre deste host.txt. Elobarava uma nova cópia uma vez por semana, e os administradores de rede, quando viam que seu arquivo estava desatualizado, descarregavam um novo arquivo host.txt e as coisas não mudavam muito. Então não tinha o que ser mudado muito, e tudo se fazia descarregando o FTP. Que é uma solução com poucos recursos tecnológicos, mas com alguns problemas, que poderiam ser previstos, e pensamos nessa situação. E alguns problemas eram a controvérsia pelos nomes. E nós temos 24 caracteres para definir o nome de um computador, e quanto mais dispositivos na rede, mais controvérsias existem quanto aos nomes. E mais difícil ainda é evitar duplicação.

E para pior as coisas, esse arquivo se mantinha de forma muito simples. NIC era editado através de um editor de texto. Não existia uma base de dados, apenas era um arquivo de texto. Então não existia um bom método para evitar duplicações, e as vezes apareciam. Essa era um problema.

Outro problema, óbvio, era a sincronização. Nunca todas as pessoas tinham a mesma versão do arquivo. Sempre alguém ficava por trás. Depois do tráfego ir era um problema. O arquivo começou a crescer tanto que exigia muita largura de banda para

baixar o arquivo, nesse momento, as conexões de 64 kilobytes eram as mais rápidas. Então me falaram, isso antes de eu ser adulto, mas naquela época levava mais tempo descarregar arquivo, do que leva atualizar. Ou seja, nunca podíamos obter a última versão, porque quando terminávamos de descarregar o arquivo anterior, já existia um arquivo novo que deveria ser descarregado.

Então claramente manter esses arquivos host a nível central não era escalável. Deveria se fazer alguma coisa. A princípio de 80 se falou como mudar esse host.txt. Primeiro para solucionar o problema de escalonamento, e em segundo lugar, está o problema do routing, pelo correio eletrônico. Quando pensamos em DNS lembramos o primeiro tema, que tem a ver com escalamento. Mas esquecemos, ou não sabemos, ou não levamos em conta os temas de roteamento de correios eletrônicos. E o resultado é o sistema do nome de domíno do DNS.

Esse é o resumo do DNS, se temos um slide resumido do DNS, é esse. DNS é uma base de dados distribuída. E nessa base de dados, os dados se mantém a nível local. Ou seja todo mundo tem a sua própria parte da base de dados, e mantém seus próprios dados. Mas esses dados estão disponíeis a nível global, porque a base de dados está distribuída em toda a internet. Então nós mantemos os dados a nível local, mas podemos ver

os dados de todos os demais. Os arquivos de DNS têm um sistema cliente resolutor, que estão do lado do cliente. Os resolutores enviam consultas. Os servidores de nomes são de servidores, e se queremos registrar algumas coisas aqui, se resolvem consultas. Se enviam consultas, e os outros as respondem. As atualizações de DNS utiliza a maioria cache para melhorar a performance. Isso significa que como estamos falando de uma base de dados distribuída em todo o mundo a velocidade da luz é uma definida.

Então quando fazemos uma busca da base de dados talvez devemos fazer um consulta cobrindo todo mundo. E isso leva tempo. Então é muito útil, de fato, é vital não só lembrar o último resultado da busca, mas todos os resultados intermediários dessa busca, e isso passa a memória cache, e acelera o processo para a vez seguinte. O DNS também utiliza replicamento e distribuição de cargas. O que significa isso?

Como já falei, cada um mantém a sua propriedade local dos dados, mas o replicamento não é de cada um. Há várias copias. Então há redundância. Ou seja, se nós tivéssemos uma cópia dos dados, ou se alguma coisa acontecesse com esses dados, poderia ser recuperado. Mas se temos múltiplas cópias, temos redundância. Isso também distribui a carga. Se muitas pessoas fazem buscas, e há muitas cópias, a carga de buscas se distribui entre todas as cópias.

Aqui temos os diferentes componentes, ou elementos, do DNS. Em um único slide. E vamos falar desses elementos no detalhe depois. Mas serve para que apareçam todos juntos, para sabermos para onde vamos.

Começamos na parte abaixo à esquerda. Aí aparece um dispositivo conectado à um telefone, mas todos os dispositivos conectados à internet devem transformar nomes em endereços. Todos os nomes, ou todos os dispositivos que utilizam DNS tem um cliente chamado resolutor stub. Um resolutor stub é uma ponte entre um aplicativo, neste caso um navegador. É o resto de uma ponte entre a aplicação e o navegador, e todo o resto do DNS.

Então resolutor stub toma o pedido de um aplicativo, por exemplo, de transformar o nome no endereço, depois transformar em uma consulta de DNS que envia para frente, e envia para o resolutor recursivo. Esse resolutor aceita o pedido e um aplicativo, transforma em uma consulta de DNS, e envia essa consulta e espera uma resposta.

O resolutor da outra parte tem uma função mais complexa. Sabe como entrar em contato com diferentes servidores, onde estão os dados do DNS. Os dados que se consultam, que se buscam. Um resolutor recursivo talvez tenha que entrar em contato com um resolutor para buscar resposta. Talvez entrando em contato

com o servidor. Eu não tenho resposta, mas eu posso talvez me referir a outro servidor, e esse segundo servidor diz que também tem resposta, mas eu posso derivar um terceiro que está mais próximo.

Então o resolutor recursivo inteligente vai navegando por todos esses servidores, até encontrar respostas. E se buscamos dentro do resolutor recursivo vemos que é um servidor de nomes, e o resolutor. O resolutor de nome responde consultas. Então do lado do resolutor enviam consultas, e o resolutor envia uma consulta aos servidores de nomes autoritativo. E também já vimos a parte do cache. Tudo que recebe o resolutor, todas as respostas que recebe o servidor autoritativo coloca no cache para poder utilizar essa informação depois para responder consultas. Esse é o sistema do DNS. Descrito a nível geral.

Vou identificar aos conceitos básicos estão os espaços de nomes. DNS é uma base de dados distribuída, e a estrutura dessa base de dados é o que chamamos o espaço de nomes. Talvez os senhores conheçam isso, quando falo de base de dados talvez pensem em uma base de dados da relação, e a estrutura dessa base de dados é que inclui múltiplas tabelas, e elas têm fileiras, e dentro de cada uma temos informações.

Assim estão estruturadas as bases de dados relacionados. Os DNSs são totalmente diferentes. É o que chamamos uma árvore

invertida, e aqui eu tenho um exemplo de uma parte muito pequena dos espaços de nomes de DNS. Em uma árvore invertida a raiz está acima. E os galhos para baixo. Esse seria um elemento informático, e essa base de informática é lógico que esteja assim. Raízes para cima, e galhos para baixo. E todos os nodos têm um nome, têm uma etiqueta.

E o nó da raiz que está na parte de cima não tem uma etiqueta. A etiqueta é uma etiqueta vazia, não tem nada. As vezes as árvores são apresentadas como um ponto entre aspas, para apenas demonstrar que não tem nada, que está vácuo.

Esse é o espaço dos nomes. E muitas vezes chamamos esses nodos, situações nodais, conforme sua posição na raiz. Esta acima né? E para a esquerda estão os nodais básicos, os de segundo nível. E vamos descendendo no espaço dos nomes.

E as vezes nós chamamos nomes de família, pais e filhos. Aqui a raiz seria o pai do .com, e .com seria o pai de exemplo, mas exemplo é por sua vez filho de .com. Falamos de relações pais e filhos. Cada uma dessas etiquetas têm uma série de caracteres que podem ser utilizados, chamamos LDH. Letras Dígitos e Hífen. É a única coisa que se pode utilizar no DNS. E a extensão máxima é de 63 caracteres. Então também os nomes das etiquetas não interessa se se escreve com caixa alta ou não. Podemos misturar e são equivalentes.

Todos esses nós têm um nome de domínio, e o objetivo de um nome de domínio é saber onde está o nó no espaço de nomes. E a definição nomes de domínio é muito clara. Começamos com um nó, e vamos avançando para a raiz, e vamos escrevendo os nomes das etiquetas, e não me recordo qual o ponto. Aqui vem um nó que está em cima. 3W. Depois vamos para o pai que é exemplo, escrevemos exemplo, colocamos com, e ponto e chegamos a raiz. Alguns nomes de domínio são nomes de domínio totalmente qualificados. O nome de domínio inequívoco diz exatamente onde está um nó no espaço dos nomes, e um nome de domínio inequívoco sempre termina com um ponto. E um ponto é o que separa o TLD, no caso .com, e a raiz. Chegamos a um domínio de alto nível. Colocamos um ponto, depois colocamos a etiqueta da raiz, que não existe na verdade. Então essa etiqueta termina em ponto. E isso é conhecido com o sistema Windows ou Linux, esse é um exemplo de uma estrutura de dados que também pode ser representada como árvore invertida.

O sistema os arquivos ou nodos representam arquivos ou diretórios, em lugar de nome de domínio temos o que nos diz o que está o arquivo ou diretório no sistema de arquivo igual que o sistema de nodos diz onde estão os arquivos de nomes.

E temos outro termo importante aqui, domínio, e nodo no espaço de nome e tudo que estiver por baixo. Por exemplo,

tenho aqui o domínio .com, então ponto com seria um nó ponto com e tudo quanto estiver por baixo disso. Em realidade de 131 milhões, ou seja, que há vários que não estão no slide, mas o domínio .com é muito grande, tudo quanto estiver por .com. Todos os que terminam em .com estão no espaço .com.

Agora vou comparar isso com o termo zona, e esse é um termo muito importante, porque é mencionado muitas vezes, e temos que entender o que é. Lembrem que chegamos aqui e temos o DNS, porque precisávamos de uma administração distribuída em manutenção central, cujo nome é host, e de endereços e não funcionava muito bem. Então tínhamos que procurar um sistema distribuído para que todos tivessem seu endereço, portanto o espaço de nomes está dividido para que esteja administração distribuída, e essas divisões administrativas são chamadas de zonas.

Então há diferentes zonas, e cada um pode jogar dentro da sua zona, se pode fazer todas as mudanças possíveis sem afetar ninguém, cada um maneja sua própria zona, e as zonas por delegação são divididas. Se delega desde o mais baixo até o mais baixo, então a zona delegação se chama pai, a zona criada é filho. Esse processo começa na raiz e avança para baixo. Um exemplo, temos o espaço de nomes de novo, e se olharem, e ouvirem assim, não temos suficiente informação para saber onde estão os limites das zonas. Então o espaço de nomes, se

simplesmente for olhado, não sabemos como estão divididos aos fins administrativos. Vou passar um exemplo. Imaginem que estão olhando desde um satélite, ou uma estação espacial. E vêm a América do Norte. Se olharem para a América do Norte desde cima, não sabem exatamente que em realidade tem ali 3 países. Estados Unidos, Canadá, e México. Desde o seu TLD não vão sabe-lo, embora estejam todo o dia vendo quais os limites administrativos. Porque não se vêm. Precisamos de dados adicionais.

Vemos os nomes, mas não sabemos onde estão os limites das zonas. Vou marcar alguns limites de zonas. Eu sei onde estão as delegações espaciais, então eu posso mostrar os limites das zonas. Acima de tudo, temos a zona raiz. E a zona raiz delega nas zonas que estão no primeiro nível. E dizemos que uma zona tem informação de delegação que se encaminha para a zona na qual delega esse processo de delegação contínua para baixo, ao longo do espaço de navegação. .com nesse caso delega a exemplo.com e bar.com e footbar.com. Então temos que pensar no tamanho dessas zonas, e vamos começar com a zona cinza.

A última vez que eu procurei havia 1.543 zonas de domínios de alto nível. A zona raiz é pequena. Só precisa de informação de delegação sobre 1.543 zonas que estão por baixo delas. Mas vamos passar para o .com. .com por outra parte tem o material que inclui a Verisign. O ultimo relatório diz que há 101 milhões

de nomes em .com. A zona .com é a maior que existe. Então deve ter informação que vai delegando 153 milhões de zonas de segundo nível que estão por baixo de .com. E a delegação pode continuar por baixo do segundo nível, mas pode haver uma delegação no segundo ou terceiro nível, avançando no espaço de nomes.

Eu lembro que falei, faz pouco tempo, que DNS utiliza a replicação para a redundância e melhorar o desempenho. Bom, essa replicação é do que estamos falando agora. Lembrem que os servidores de nomes respondem a consultas. Dizemos que um servidor de nomes é autorizado para uma zona se te conhecimento completo a respeito da zona, e que esse nome autoritativo pode responder consultas definitivamente se alguém as fizer. Pode dizer, me perguntam algo acerca da zona que está a informação, para dizer que você pediu algo, fez uma consulta sobre o nome que não existe. Então posso dizer que esse nome não existe nessa zona. As zonas deveriam ter múltiplos servidores autoritativos. E, como disse, fornece redundância, e distribui a carga. Então cada zona tem que ter pelo menos um servidor de nomes autoritativos, e em realidade vai ter mais de um. É uma melhor prática ter pelo menos 2 servidores autoritativos, porque com um só, se houver um problema, ninguém poderá fazer consultas na zona.

Se teremos múltiplos servidores autoritativos, é necessário manter a informação sincronizada. E a ideia é que a informação acerca de uma zona deveria ser a mesma em todos os servidores autoritativos.

Então, como fazê-lo? A boa ideia é que o protocolo tem uma forma incorporada. A sincronização de dados de zona nos servidores autoritativos, e eles estão incorporados a transferência de zona, que permite se mexer na zona. Então indicam o servidor autoritativo. E isso nos permite fazer modificações na zona, e depois temos outros servidores autoritativos chamados secundários, ou escravos. Eles se referem ao mesmo.

Esses servidores secundários carregam os dados da zona. O primeiro se contata com o primário, e fazem transferência de zona, e copia a zona do primário para o secundário. Então é importante apontar que todos os servidores autoritativos são iguais. Têm os mesmos dados. A única diferença é que entre primário e secundário, de onde obtém os dados.

O primário carrega os dados da zona até o seu disco, e o secundário carrega os dados da zona a partir do primário. Mas os dados do primário e secundário são os mesmos.

É claro que pode ser que por um momento não estejam sincronizados. Porque se fizer uma modificação no primário, há

uma demora, mas essa mudança se propaga ao secundário, se estão muito bem, estão incorporados ao protocolo de DNS. Você não tem que preocupar com manter os endereços de nomes sincronizados. Isso acontece. Nós queremos que vocês façam.

Vamos passar para o próximo nível. Falamos das zonas, vamos falar sobre o que acontece dentro da zona. Dentro da zona. Lembrem que todos os nodos no espaço de nomes, cada uma dessas caixas têm um nome de domínio. E a forma de vê-lo é a seguinte, o nome de domínio determinado pode ter diferentes classes de informação, de dados associados.

A classe mais comum é o endereço IP, se pode vincular um endereço IP a um nome de domínio. Essa classe de informação que acompanha o nome de domínio é chamada de relatório de recursos. Registro de recursos, perdão. Há diferentes tipos de registros de recursos que armazenam diferentes classes de dados. O registro de recursos mais comum são os que registram endereços IP. IPv4, IPv6.

Mas esse registro de recursos também armazenam outro tipo de dados. Então uma zona consta simplesmente de uma série de registro de recursos, e todos os registros de recursos de uma zona se colocam em um arquivo que é chamado arquivo de zona.

E há arquivo de zona para cada zona, e nunca se mistura um registro de múltiplas zonas em um arquivo. Então, uma zona não é mais do que um conjunto de todos os registros de recursos.

Vou mostrar como são esses registros de recurso. De fato, há uma forma padrão de escreve-los com texto. O registro de recurso tem 5 campos. Não é necessário ver todos. A mensagem importante aqui é que o registro de recurso pode ser diferente, e transferem, transportam dados diferentes. Esses são os tipos de registro de recurso mais comuns.

Temos um tipo de registro de recurso para IPv4, e para IPv6. Chamado registro de endereço. Depois temos um tipo de registro que armazena endereços IPv6, e esse é chamado de quadruple A, ou 4 As. E depois temos outros tipos mencionados aqui.

Essa lista é a que inclui os tipos de registro de recurso mais comuns, mas há muitas outras classes de registro de recursos. Da outra vez que eu vi, há muitos desses registros. O registro da IANA, que é denominado, que vocês podem ver na tela, se vocês entrarem nesse website vão se encontrar com isso.

Então com base no tamanho do campo do registro de recurso podemos chegar até 65.000. Agora se pensarmos em alguma coisa nova, podemos entrar no IFT, convencer as pessoas de que

a nossa ideia tem que se transformar em um novo tipo de DNS, e, nesse caso, vai se criar, vai se dar uma classe no registro. E vamos poder colocar coisas novas no DNS.

As pessoas fazem isso sempre. Não com tanta frequência, mas as pessoas vão criando coisas novas, que podem, se colocam no DNS e se armazenam novas classes de dados.

Mas de longe, a classe de dados mais comuns de DNS são os endereços. Os mais comuns de DNS são para mapear nomes de domínio com endereços. E aqui temos esses 2 tipos de registro que mostrei antes.

Aqui temos a representação, o texto do que é um registro de endereço. E o registro quadruple A. Temos exemplo.com, depois o tipo, que no primeiro caso é A pro endereço, e depois temos o endereço propriamente dito. Esse é um exemplo de um registro de recurso que vai estar no arquivo zona para o caso de exemplo.com. É aí que vemos o exemplo.com, e de IP. E abaixo temos um registro quadruple exemplo.com quadruple A. E o endereço 6. Porque essa é a maneira mais comum de mapear os nomes com endereços. Há outro tipo de registros.

A maioria dessas classes são utilizadas por gente que consome informação de DNS, que tenta se conectar com um DNS, mas as vezes algumas classes que são utilizadas pelo DNS propriamente dito, e um exemplo disso, é o registro DNS e

registro SOA. E essas são coisas que não são importantes para ninguém, mais do que para o DNS. E o interessante é que essas classes iguais à outras classes.

Eu gosto de ver isso, usar analogia de um depósito. Alugamos um depósito, porque queremos armazenar um monte de coisas ali. Não trazemos um caminhão e começamos a jogar as coisas, precisamos de prateleiras, e se tivermos temos que tirar as caixas do caminhão, e colocar as caixas nas prateleiras. Então se o depósito não tem prateleiras não serve de nada.

O DNS é similar. Podemos pensar no registro DNS SOA, como se fossem prateleiras, mas ninguém fora do DNS se importa sobre isso. Os que estão fora disso se importam com as outras classes de registro.

Vamos falar sobre esse registro DNS. É assim como dizemos quais são os servidores de nome autoritativo para uma zona. Esse exemplo mostra os registros DNS. Esses registros DNS dizem o seguinte, dizem que exemplo.com, essa zona tem 2 servidores de nome autoritativos. Um é NS1 e o outro é NS2. A esquerda está o nome da zona, e a direita o do servidor de nomes.

Os registros NSs são um pouco complexos, porque aparecem em 2 lugares. Aparecem na zona pai e na zona filho. Nesta caixa tenho toda a lista de registros DNS para .com. Há 13 servidores

de nomes autoritativos para essa zona, e são chamados, como podem ver, A.gTLD-SERVERS.NET. Essa lista de registros aparece em 2 lugares.

Vamos agora para cá. Aparece na zona raiz, neste caso a zona raiz é a zona pai, e esses registros NS na zona pai são os que fazem a delegação. Isso que se diz ao resto de DNS que por baixo da zona raiz temos a zona .com.

E a lista de registro do DNS volta a aparecer na zona, que neste caso, é chamada de .com. O registro DNS .com aparece na raiz, no pai, depois em .com. E quando falamos novamente para procura-lo, veja quão importante que esses registros NSs estejam na zona pai.

E vou dizer agora que a forma em que funciona a resolução em DNS é começando na raiz, e seguindo esses pontos de delegação procurando registros DNSs, e se estamos procurando alguma coisa que está por baixo do .com, começamos na raiz, na zona raiz tivemos a delegação .com, e vamos ao servidor de nomes .com e procuramos a delegação que está por baixo, e assim por diante.

Depois voltaremos a esse tema. Os registros DNS incluem apenas nomes, então se procuramos o exemplo de antes, vemos que exemplo.com, dos servidores de nomes NS1.exemplo.com. Mas isso não é suficiente, porque precisamos dos endereços IP.

Então a informação de delegação também deve incluir registros de endereços em um dos casos. E isso chamamos registro de aderência.

É o registro de direção para o registro do nome. Há outro registro que está em todas as zonas, que se chama registro SOA. Eu não vou falar muito sobre esse registro, apenas quero mencionar que existe, de que há um registro SOA por zona. Esta na parte superior no que chamamos o ápice da zona.

A maior parte dos valores estão relacionadas com essa transferência de que falei antes. Dizem os servidores autoritativos como sincronizar, com que frequência sincronizar a zona.

Temos agora para o segundo objetivo. O segundo problema que deve resolver o DNS. O roteamento do correio. O problema que deveria resolver o DNS era como entregar na base de um endereço de email? No passado, antes de que existisse o DNS, tínhamos endereço de email com o usuário @ e o nome do host. Era um desses nomes que tinham 24 caracteres. Esse era um endereço de email. E antes de que existisse DNS, isso significava que o meu email iria ao nome do host que estava a direita do endereço de email, não tinham forma de dizer, o que endereço de email é uma, mas a mensagem deve ser enviada à outro aparelho.

Se o meu endereço era MATT@FOO tinha que chegar à esse servidor. Agora um dos objetivos do DNS era separar isso, e poder dizer pronto, esse é meu endereço de email, e este é o endereço que deve ir o email. Mas deve ir à outro lugar. Então o DNS oferece a flexibilidade necessária, temos um registro que se chama o registro de intercambio de emails, que diz para onde deve ir o email. Esse é um exemplo.

Então esses registros MXs, assim se chamam, para o caso de exemplo.com. Dizem para onde deve ir o email. Então no caso de qualquer nome de usuário@exemplo.com, o registro MX diz que deve ir a um aparelho chamado mail.exemplo.com. Há um valor de preferencia, que é o nome. E isso vai um pouco contra a intuição, porque há um que é mais desejável do que o outro. Então esses 2 registros é o que dizem.

Qualquer endereço de email @exemplo.com deve ser enviado a mail.exemplo.com, mas se não pode ser feito por algum motivo, então devem tentar de enviar a mail-backup.exemplo.com.

Então o servidor de email que vai enviar email devem poder enviar emails para os nomes de domínio, há uma união muito estreita entre o DNS, e o mail com base no SMTP. Então o servidor de email contem uma mensagem para entregar, busca o MX para o endereço de email, e aí sabe onde enviar a mensagem.

Até agora estivemos falando em como mapear os nomes com os endereços de IP. O que é um trabalho importante, mas as vezes queremos fazer o contrário. O mapeamento de nomes de IP se chama mapeamento para frente. Agora o que acontece se queremos fazer um mapeamento de IP para o nome. Em alguns casos isso é necessário.

Imagine, por exemplo, que existe uma rede que quer detectar um problema, e que ver quais são todos os roteamentos. Quando procuramos o IP através de todos os routers, talvez nos interessa algum endereço IP, e também saber onde está, quem opera, como se chama. Esse é um exemplo de mapeamento inverso. Se trata de tomar um endereço IP e encontrar um correspondente.

Se temos apenas um endereço IP, e queremos fazer um mapeamento para frente é relativamente fácil. Temos um nome, quero mapear um nome IP, procuramos até encontrar. E temos o endereço IP.

Com que seria fazer um aperto inverso. Então isso funciona muito bem através do host. O que fazemos com o DNS? Vou voltar para ver o exemplo do espaço de nomes. Como está estruturado esse espaço de nomes? É uma forma que seja muito fácil encontrar os nomes e domínio, mas no caso é impossível se queremos encontrar um nome de domínio, por exemplo,

exemplo.com. Pode ver que começo na raiz, e vou descendo até chegar a www. E se começo com o endereço IP pronto, a resposta é que não podemos fazer isso no DNS. Porque não podemos buscar um endereço IP.

Então o que tinha que passar era que devemos encontrar uma forma de converter os endereços de IP para depois encontrar como nome de domínio, e isso é exatamente o que teremos. Então outro tipo de registro que se chama PTR por pointer, ou ponteiro. E os endereços IPs vão a nomes de domínios, sendo que temos registro PTR, como nome de domínio, e encontrar-se. Então se tem o endereço de iPv4, tem o nome de domínio que se chama INADDER.ARPA, e o IPv6 vai com IPv6.ARPA.

Eu vou mostrar um exemplo, aqui está a árvore de nomes. Aqui está uma parte nova, que talvez os senhores não saibam que existe. Temos .com, que vocês já conhecem. Mas aqui tem o domínio ARPA, e o domínio IN-ADDER. E aqui deveríamos ter o registro. Então a direita temos o registro .com, é 192.0.2.7. Então procuramos se sabemos qual é o endereço, agora o que acontece se eu gostaria de saber qual o nome de domínio que corresponde ao 192.0.2.7.

No caso eu tenho que converter esse endereço de IP em um nome de domínio. O que eu faço? Pego esse endereço IP, inverto, coloco IN-ADDER.ARPA. E assim procuro o registro PTR.

É assim e todos conhecem e sabem quais são as regras. Todos sabem tudo que estou explicando. Mas se eu tenho um espaço de nomear sem nada uma pessoa, os registros regionais cooperam para administrar este domínio. E assim podemos ter uma zona que corresponde ao endereço de IP delegada.

Digamos, por exemplo, que temos 192.0.2 e tudo começa assim. Nesse caso eu vou ter esse domínio de 192.INADDER.ARPA assinado para mim mesmo. E querem fazer o inverso, vai acontecer a mesma coisa. Então é um pouco estranho, mas funciona.

Talvez a melhor das funções não é importante como mapeamento para frente. Esse mapeamento é o que permite marcar nomes de domínio em um website e chegar no navegador e chegar ao website. O mapeamento inverso se faz com fins para detectar os problemas. Em termos gerais, uma pessoa normal não faz isso. Apenas os engenheiros de sistema, e os especialistas se preocupam do mapeamento inverso.

Como falei antes, há outras classes de registro de investigação. Esse é um exemplo de alguns casos a mais para que saibam quais são os outros tipos de dados que as pessoas colocam no DNS.

Aqui temos um exemplo do que é um arquivo de zona para uma zona muito pequena. Este é um arquivo de zona para uma zona

exemplo.com. E não falei dos outros registros no detalhe, mas esta é uma zona pequena, parecida. Ou que tem todas as zonas de internet.

Porque se pensamos, a maioria dos nomes de domínio na internet provavelmente vão ter duas coisas. Temos o servidor web, e a ideia é receber domínios que fazem muitas outras coisas. Têm muitos nomes, mas são a maioria dos nomes, e muitos nomes de domínio.

Por exemplo, nome pessoal que tem para correio eletrônico. É a única coisa que precisamos no meu caso. Temos um endereço de IP exemplo.com, que é onde está o servidor web. Aí podem ver o arquivo de zona, e podem supor que 192.0.2.7 é o endereço do website, e depois vem alguns registros adicionais que dizem onde devemos enviar o correio eletrônico, que chegue à exemplo.com.

Agora eu quero falar do processo de resolução, e é dessa forma como se buscam coisas no DNS. Os componentes que eu marquei no princípio da sessão, os resolutores recursivos. E os servidores de nomes autoritativos, todos cooperam para buscar dados no espaço dos nomes. O importante é que uma consulta do DNS, sempre tem três parâmetros. O nome de domínio, como www.exemplo.com. O exemplo que procuramos, no caso a classe, da qual não falei.

Classe é uma forma que poderia utilizar para levar o DNS para outro tipos de rede, mas não se utilizou na verdade. Mas está no DNS. Então ali ficou, nesse caso, classe. Sempre será classe de internet, e não devemos nos preocupar por outra coisa. Apenas é um nome de domínio e o tipo o que é importante. O tipo de dados.

Então se fazemos uma consulta ao DNS, se fazemos uma pergunta, sempre devemos especificar o nome de domínio, e o tipo. E há 2 tipos de consultas. O resultor stub, e é o nosso telefone, a torradeira, a geladeira. Tudo que se conecta à internet. E qualquer outro tipo de informação. E tudo isso tem um resultor stub, manda o que chamamos consultas recursivas.

Essas consultas são sinais para que o resolutor recursivo escute. Bom, eu sou recursor stub, e me deu a resposta uma mensagem de erro. Não me interessa uma resposta parcial, eu quero uma resposta completa. Os servidores recursivos, por outra parte, são mais inteligentes. E podem aceitar respostas parciais que se chamam derivações, então enviam uma consulta que indica que estão disposto a aceitar respostas parciais, derivações.

Então como já falei antes, buscamos o DNS como na zona raiz, e continuamos avançando. Seguimos a delegação através dos ponteiros de delegação. Então há servidores autoritativos para a

zona raiz, que tem toda a informação da zona raiz, isso significa autoritativo, e isso chama servidores da zona raiz.

Então, quando começamos a ver a solução na zona raiz, temos que conectar a um servidor de nomes raiz, e como encontramos quais são os servidores raiz, a resposta é que é necessário configurar. Não há forma de descobrir. Devemos configurar em todos os servidores de nomes recursivos, e isso é diferente de outros parâmetros de rede. Por exemplo, quando o meu telefone se conectou com a rede Wi-Fi da ICANN aqui no centro de conferência, se utilizou um processo, ou um protocolo especial de configuração dinâmica. E o meu telefone tem sua rede. Eu não conheci então a rede diz, esse é o servidor IP, e esses são outros parâmetros de configuração que tem que conhecer também. Isso inclui o endereço IP de nomes recursivos. Dessa forma talvez se deposite toda a informação que precisa.

Isso não acontece com os servidores recursivos. Não podemos ativar um servidor desse tipo sem configuração. Ou seja, o configurador deve saber quais são os nomes raiz, e qual o endereço de IP. Então o arquivo especial que precisa de todos os servidores de nomes recursivos que tem nomes, e endereço de IP de todos os servidores raiz. A boa notícia é que se instala um servidor recursivo no Linux, alguém já fez isso de colocar todo o software dos servidores com informação do arquivo de raiz.

Alguns resolutores recursivos também têm os endereços de IP, e os nomes dos servidor raiz.

Mas esse é o LR, onde podem procurar o arquivo. E esse é o aspecto que tem. Há 13 servidores de nomes raiz. São 13 servidores autoritativos para a zona raiz. A esquerda temos a zona raiz, depois há 13 registros DNS que são os nomes dos servidores de nomes, que são chamamos A.ROOT-SERVERS.NET. Depois colocam por baixo os endereços IPv4 e depois os IPv6. Então todos os servidores têm um endereço IPv4, e IPv6.

Eles precisam dessa informação, devem conhecer os nomes, os servidores do nome raiz. Então agora fazemos um desvio, e vamos falar da zona raiz. E de como chega a informação da zona raiz. O que temos na zona raiz? Temos informação sobre as zonas de domínio de alto nível. Temos registros para o DNS. É difícil, complexo.

Aqui há duas organizações, a ICANN, que é o operador da zona da IANA, e as PTI manejam isso. E Verisign é a outra organização, e eles são os que mantém a zona raiz.

Essa forma de trabalhar data de princípios dos 90, quando Verisign era uma empresa que tinha outro nome, Network Solutions, que foi comprada por Verisign em 2000, e é levado a cabo pela Universidade da Califórnia do Sul, antes de ser criada a ICANN. Então esse é um acordo muito antigo, um pouco

complexa a form ade trabalho, mas é assim que funciona a zona raiz.

Então essas 2 organizacoes, ICANN e Verisign, cooperam para colocar os dados da zona raiz, para criar o arquivo da zona raiz. Depois precisamos de servidores autoritativos para a zona raiz, e há 12 organizações que operam os servidores autoritativos. Isto é pouco. Na maioria das zonas há uma organização que opera todos os servidores autoritativos. Vamos trabalhar com .com, eu trabalhei na Verisign, então sei como funciona isso.

Verisign opera todos os servidores autoritativos de .com. Temos outra zona, como por exemplo, muitas empresas fazem o seguinte, operam alguns servidores autoritativos por si próprios, ou terceirizam essa atividade em um terceiro, ou haja múltiplos terceiros para que haja redundância, mas não há só 2 organizações, isso é pouco comum.

Aqui temos as letras dos 13 servidores da zona raiz, se chamam raiz A, como abreviatura, e essas são as organizações que operam esses servidores. É um grupo interessante, são organizações que não têm nada em comum, mas operam o servidor da zona raiz. Depois há instituições organizações, organizações sem fins lucrativos, ISPs. Certos departamentos do governo dos Estados Unidos. Um pouco de tudo Mais uma vez isso se volta ,faz muitos anos, é algo que aconteceu faz 20 anos.

Essa lista de operadores, mais ou menos, tem uns 20 anos e há uma série de temas complexos que têm a ver com isso que não posso explicar em detalhe por falta de tempo.

Temos 3 servidores raiz. O grupo A e o grupo B são operados, e o grupo J são operados por Verisign. Esses são os servidores raiz, se querem ter mais informação podem entrar ao website ROOT-SERVERS.ORG, e ele vai dar informação adicional. Vão encontrar ali. Depois quero mostrar em nível bem geral, como se fazem as modificações na zona raiz. Toda a informação na zona raiz tem aver com os TLDs. Então se queremos fazer uma modificação, adicionar um servidor autoritativo para ser TLD, ou tirar um servidor autoritativo, modificar o endereço IP envia uma modificação as funções da IANA, que é manejada pela ICANN, e a PTI faz modificações, e atualiza a zona raiz, e manda essa informação ao encarregado de informação. Verisign faz outras verificações, atualiza sua base de dados, cria um arquivo de zona raiz, e o difundo depois dos 13 servidores raiz. Expõe esse arquivo. Essa é uma descrição muito geral do processo. Estou explicando para que vejam como as organizações cooperam para que isso funcione.

Mas o processo é bem mais complexo do que parece. Esse é um pequeno desvio que tomamos para a zona raiz. Vamos falar agora de como funciona a resolução. Vamos supor que temos um telefone abaixo a esquerda, e alguém abriu o navegador do

telefone, e escreveu www.exemplo.com. Então o navegador stub é um código simples aqui, vemos que há uma chamada de uma API. Uma interface de programação de aplicativos. É um resolutor stub. É simplesmente uma função e diz “preciso do endereço de www.exemplo.com.”

Então essa stub é muito simples. Só conhece o endereço IP de vários servidores recursivos aos quais deve enviar a consulta. Então o resolutor stub transforma isso em uma consulta, e envia a consulta ao DNS ao servidor de nomes recursivos, que é 4.2.2.2, esse é o endereço IP de um servidor de nomes recursivos. E esse servidor é um servidor aberto, ao qual qualquer pessoa pode fazer consulta.

Um bom endereço de IP, e o stub pede ao resolutor discursivo que dê a resposta, ou consulta que chegou, para que o exemplo seja mais interessante.

Lembrem que os resolutores discursivos têm memória cache, mas lembremos que só agora que se ativou. Só sabe ou conhece os nomes de endereço IP dos servidores raiz. Então o resultado recursivo toma um dos servidores raiz, escolhe um. Por exemplo, escolhe esse e faz a mesma pergunta que acaba de receber do resolutor, que é “qual é o endereço IP de www.exemplo.com.” Agora o servidor raiz não pode responder essa consulta, porque não conhece o endereço IP de

www.exemplo.com. Não sabe nada sobre .com, mas sabe sobre .com porque a zona raiz tem a delegação de .com.

Então o servidor raiz pode ser levado, e o resolutor recursivo coloca ali no endereço IP, e depois faz o que chamamos de uma derivação. Escolhe um dos servidores .com e manda a mesma consulta. Então o servidor .com é chamado de C.gTLD-SERVERS.NET, e está ali na tela. Podem ver aí o endereço.

Esse é nome dos servidores .com. Então escolheu um dos servidores .com C.gTLD e faz a mesma pergunta, que recebeu do resolutor stub. E depois com o endereço IP de www.exemplo.com. O servidor também não conhece o endereço, mas sabe qual é o servidor de nomes autoritativos, e envia essa lista. Manda uma lista para onde pode ir o resolutor recursivo, que agora coloca isso na segurança cache. Segue a derivação, e manda a mesma consulta por terceira vez à um dos servidores autoritativos, que é NS1.exemplo.com.

Esse servidor sim dá resposta ao resolutor recursivo que coloca seu cache, manda ao resolutor stub, que passa a aplicação. E agora, a aplicação com o endereço IP, pode entrar em contato, fazer download da página do website e pronto. Deve ser mais simples. Como a maior parte das coisas que se faz em DNS, e o importante é que ver que começamos pela raiz, e vamos

descendo. Nem sempre começamos pela raiz, porque usamos a memória cache.

Mas o que acontece se alguém vem imediatamente, que é FTP.exemplo.com?

O resolutor stub prepara a consulta do DNS, manda ao resolutor discursivo, mas lembra tudo quanto tem resolutor discursivo com a memória cache.

Por exemplo, os resolutores .exemplo.com, então o resolutor vai direto para exemplo.com ao servidor que tem isso, recebe a resposta, e a manda ao resolutor stub, que manda à aplicação. Então colocar coisas na memória cache acelera os processos, se não fosse pelo uso do cache, tudo seria muito mais lento.

Último slide que eu tenho, dá uma descrição geral de tudo isso. Nomes de domínio, pensamos em mais alguma coisa do que os servidores autoritativos. Pensamos em todo o panorama que inclui registro, ou registratários. E aqui quero mostrar onde estão os outros atores do mundo dos nomes. Temos registratário que se comunica com registradores, website, para o registrador se comunicar com o registro. E o registro tem uma base de dados onde estão registrados os nomes de domínio, a informação pertinente, o registro. Tem informação na base de dados através de um servidor de nome autoritativo, e é isso que é utilizado pelos resolutores recursivos.

Espero que isso tenha explicado um pouco como encaixa o que estivemos falando em todo o quadro geral. Este é o final dos meus slides, e com muito prazer vou responder as perguntas que vocês possam ter.

CATHY PETERSEN: Por favor, antes de fazer uma pergunta digam seu nome, e a que organização pertencem.

MATT LARSON: Alguma coisa no Adobe room? Todos estão com fome né? Eu também estou com fome. Aqui tem uma pessoa do público.

MALISA RICHARDS: Eu tenho uma pergunta, sou Malisa Richards. Bolsista da Gana. Que critérios utilizam ao instalar um servidor de zona raiz?

MATT LARSON: Depende do operador. Cada operador tem sua própria política. Eu não falei sobre isso, mas todos os servidores raiz são any cast. Ou seja que não há um único servidor de um servidor de IP para um servidor raiz. Há muitos servidores de fato. Essa técnica any cast usa sistema de roteamento de internet, que permite ter vários servidores com o mesmo IP respondendo desses diferentes pontos da rede. Então todos os operadores utilizam

técnica any cast para ter múltiplas instâncias de seu servidor raiz.

E os diferentes operadores têm diferentes políticas. Eu sei, que no caso da ICANN, a raiz L, qualquer pessoa poderia operar isso, porque é um só servidor. Mas o que pedimos é que se compre o servidor, se instale na rede, e ofereçam espaço, conectividade e potencia. Nós não operamos desde a ICANN. Diferentes servidores de raiz têm diferentes políticas.

MALISA RICHARDS: Mais uma pergunta. Estou vendo o website recomendado, e vejo que no Caribe, em particular, não há muitos servidores raiz em comparação com a América do Sul, América do Norte. Pode explicar porque é que isso acontece?

MATT LARSON: Não sei o que acontece com os outros operadores. Pode ser que os operadores dessas redes não se tenham aproximado dos operadores de raiz como para instalar um servidor raiz. No caso da ICANN, a barra é muito baixa. Se está disposto a comprar um servidor, nós podemos coloca-lo na rede de vocês. Muito obrigado a todos. Outra pergunta?

NORMAN WARPUT: Sou Norman, sou bolsista da ICANN. Obrigado pela apresentação. Tenho uma pergunta sobre o processo de modificações na zona raiz. São utilizados também para solicitar domínios superiores de alto nível?

MATT LARSON: Não entendi bem a pergunta.

NORMAN WARPUT: O processo para modificar o arquivo de zona raiz se aplica também aos nomes de domínio de alto nível com código de país?

MATT LARSON: Sim. Da perspectiva do DNS é um TLD. Nós temos categoria de ccTLDs, TLDs, gTLDs. E dentro dos gTLDs temos os que têm patrocínio e não. Temos diferentes categorias do ponto de vista do DNS, de qualquer maneira um TLD é um TLD independentemente se for ccTLD, ou gTLD. O que for.

ANDREW FRASER: Eu sou Andrew, e gostaria de saber onde está descrito como se aponta um servidor recursivo. É através do ISP? Porque há diferentes alternativas para os servidores recursivos.

MATT LARSON: Em última instância está configurado em cada dispositivo. Quando um dispositivo se conecta com a rede, a rede tem que dizer, todas as redes dizem, além do endereço, qual é o endereço IP recursivo. Qualquer um que opere uma rede vai ter que ter um servidor recursivo.

ANDREW FRASER: Então meu telefone está pré configurado por meu servidor.

MATT LARSON: Não pré conectado, mas quando conecta o endereço diz “esse é o endereço IP” “essa é a configuração”. Inclusive o recursivo que operam. Então se o senhor quisesse poderia descartar isso e escolher outro servidor. Ou um fornecedor que seja diferentes. Os abertos. Há diferentes nomes, e são empresas que se dedicam a operar servidores recursivos que qualquer um pode usar.

A primeira empresa que fez isso que eu conheça é Open DNS. Lembro que quando fizeram isso, a minha reação, e de muitos outros foi “para que fazer isso” servidores recursivos são o central da rede, para que tomar algo tão importante, do que eu dependo para uma empresa, porque vou depender de um servidor que está fora da minha rede. E a proposta da Open DNS era nós podemos dar serviços adicionadas como filtragem de

conteúdos com base em nomes. O senhor pode falar “não quero nomes que têm a ver com apostar, que tem a ver com conteúdo adulto, etc” e também podem fazer coisas como nós não vamos resolver nomes de domínio que sabemos que são sites com malware. Então eu lembro que pensava naquela época que o Open DNS era ridículo, mas depois o Open DNS foi comprado por Cisco, por muitos milhões de dólares.

Então o último que ri, ri melhor. Então o Google Public DNS e o Google disse que queria ter isso porque queria ter um servidor recursivo muito confiável, para ser a validação de DNSSEC. Se trata de adicionar a autenticação criptográfica ao DNS, então o Google acredita muito no DNSSEC, e diz, esse é o serviço qeu oferecemos, que é gratuito. E dessa forma vou acrescentar a proteção. Os de Verisign também tem um. Se a pessoa quiser pode mudar a configuração do dispositivo e usar esses.

Alguma outra pergunta? Muito bem, muito obrigado a todos então.

CATHY PETERSEN: Obrigado a todos, obrigado Matt. Obrigado transcrição, aos intérpretes, a equipe técnica de trabalho. Obrigado de novo.