Disponibilização de Nomes e Diretórios Distribuídos através do DNS

José Helvécio Teixeira Jr [1] *José Antão Beltrão Moura* *Jácques Philippe Sauvé*

Departamento de Sistemas e Computação
Universidade Federal da Paraíba
Av. Aprígio Veloso, s/n - Bodocongó
CEP: 58.109-970 Campina Grande-Pb
E-mail: helvecio, antao, jacques@dsc.ufpb.br

Suzana Q. Ramos Teixeira

Departamento de Estatística e Computação
Universidade Estadual do Ceará
Av. Paranjana, 1700 CEP: 60740-000 Fortaleza-CE
E-mail: suzana@fortal.uece.br

Resumo

Apresentamos e discutimos neste artigo o esquema e o mecanismo que compõem o DNS, um exemplo pragmático de um serviço de nomes hierárquico e distribuído. Nosso interesse no tema é proveitoso porque, em primeiro lugar, ele tem sido usado para assinalar nomes de máquinas em toda a Internet, e em segundo lugar, porque fornece um ótimo exemplo de aplicação distribuída modelada através do paradigma cliente/servidor, empregando um conjunto de servidores distribuídos que cuidam do mapeamento de nomes simbólicos para endereços IP.

Abstract

In this paper, we present and discuss the schema and the mechanism that form the DNS, a pragmatic example of a hierarchical and distributed names system. Our interest in the topic is very valuable because, first it has been used for assigning names all over the Internet, and second, because it offers a fair example of a distributed application modelled with the client/server paradigm, employing a set of distributed servers to map between symbolic names and IP addresses.

1. Introdução

As novas tecnologias que compõem os ambientes da computação distribuída representam mudanças fabulosas para os ambientes de computação atuais [Tei95]. A disponibilidade de estações pessoais de trabalho, redes locais de computadores, e redes de longo alcance de alto desempenho fornecem as facilidades necessárias para permitir o compartilhamento de informações por todo o mundo. Com facilidades de comunicações avançadas, surgem inúmeras complicações para os usuários finais, para os administradores de sistemas, e para os projetistas de aplicações. Exige-se um ambiente de computação que explore este valioso potencial sem a necessidade de se conhecer em detalhes a complexidade da tecnologia que o suporta. É necessário, portanto, um meio que "esconda" as esquisitices da rede, e que ao mesmo tempo forneça facilidades de comunicações ótimas.

[1] O trabalho deste autor foi realizado através do Mestrado em Informática da UFPb.

O **Serviço de Nomes (ou de Diretórios**) ([Hal93b] [Com93a] [CTR92b]) é o componente da computação distribuída que permite que usuários e programas possam localizar e descrever pessoas, lugares, aplicações, e serviços em ambientes distribuídos. Ou seja, o Serviço de Nomes é o componente que permite que aplicações e serviços localizem e façam referência a objetos em um ambiente de computação distribuída. Os Nomes são uma parte crítica da computação distribuída, pois ajudam na administração dos recursos disponíveis na rede.

Para preservar a consistência de objetos em um ambiente de computação distribuída, cada modificação em um destes objetos deve ser propagada para todos os sítios na rede. Em uma pequena rede, esta é uma tarefa simples, porém, em ambientes com centenas ou milhares de nós dispersos, pequenas tarefas administrativas podem vir a tornar-se uma grande dor de cabeça. Sem ferramentas que automatizem estas tarefas, a probabilidade da ocorrência de erros aumenta com o tamanho da rede e com o número de locais onde as mudanças devem ser feitas.

2. A Necessidade de Nomes em Ambientes Distribuídos

As pessoas dependem de nomes para identificar e descrever objetos. Os nomes são compostos por uma série de informações que formam os seus atributos [Keh92]. Os nomes e as informações a respeito dos objetos que eles descrevem podem ser agrupados em listas e disponibilizados publicamente, de tal forma que se possa referenciar estas informações. Por exemplo, os nomes das pessoas e empresas em uma cidade são compilados em listas telefônicas (que são na realidade diretórios de telefones) que fornecem informações como números de telefones e endereços. Em nossa sociedade moderna, as listas telefônicas fornecem um meio fácil para localizar pessoas. Todos os anos são emitidas novas listas telefônicas, de tal forma que se uma pessoa mudar de endereço, esta informação será atualizada na nova lista.

De maneira similar, as redes de computadores precisam de nomes e diretórios para descrever e guardar as características dos diversos serviços e informações que elas fornecem. Em um ambiente de computação distribuída, qualquer objeto que possa ser referenciado individualmente pode receber um nome. Exemplos de tais objetos são os serviços da rede, caixas postais de correio-eletrônico, e computadores. Cada objeto possui um registro correspondente no serviço de nomes, chamado de ***entrada***, a qual contém as informações ou atributos que descrevem o objeto. As entradas são agrupadas em *listas de entradas*, chamadas de ***diretórios*** [Bar93b].

Os *atributos* podem ser qualquer tipo de informação que descreva um objeto, como por exemplo sua localização, sua cor, ou seu tamanho. O serviço de nomes permite que os diretórios sejam organizados em hierarquias, nas quais um diretório pode conter outros diretórios. Por exemplo, um diretório de telefones internacional contém os diretórios de telefones em cada país. Os diretórios de cada país contêm, por sua vez, as listas telefônicas estaduais, as quais contêm as listas telefônicas por cidades. Finalmente, as listas telefônicas de cidades contêm as entradas reais com informações de assinantes.

O serviço de nomes é de primordial importância para os ambientes da computação distribuída porque os objetos são definidos através de seus nomes. Aplicações e serviços ganham acesso a um objeto quando referenciam a entrada correspondente ao nome deste objeto no diretório e recuperam seus atributos. A separação de um objeto da sua localização e

de suas características de acesso leva-nos ao princípio da ***independência de localização*** ([Tan92] [TVR85] [LS90]). Este princípio permite que aplicações e serviços possam ter acesso a um determinado objeto mesmo se este objeto muda de lugar ou muda alguma outra de suas características.

2.1 Nomes e Endereços

Endereços são usados nos ambientes da computação distribuída para identificar processos envolvidos em uma sessão da rede. Um endereço consiste de duas partes: a primeira parte é usada pela rede (ou inter-rede) para rotear as mensagens para o sistema final assinalado; a segunda parte é usada pelo próprio sistema final para rotear as mensagens para o processo de aplicação destinatário [Com91a].

Façamos, por exemplo, uma analogia com os sistemas de telefonia convencionais. Imagine que uma chamada envolva um pabx de uma empresa. Neste caso, os números de ramais são relativamente pequenos, pois só precisam identificar o endereço do telefone dentro de um ambiente limitado (por exemplo, ramal 22). No caso de uma rede de computadores, esta situação é similar aos endereços associados aos nós de um segmento de rede local. Se considerarmos uma ligação para fora dos limites do pabx de nosso exemplo, certamente os números de telefones terão que ser maiores, de forma a permitir que as chamadas sejam identificadas e roteadas através das linhas nacionais e até mesmo internacionais. Lembre-se, porém, que um endereço é sempre único em toda a rede (local ou de longa distância).

Endereços em geral são compostos por muitos dígitos. Em **TCP/IP** (*Transmission Control Protocol/Internet Protocol*) ([Com91a] [Com93a] [Hal93b] [Mal92a]), a primeira parte do endereço corresponde a um valor inteiro de 32 bits, o qual em uma representação mais próxima dos usuários finais poderia representar até 12 dígitos decimais. Esta primeira parte é representada pelo ***endereço IP*** *(Internet Protocol)* na rede. A segunda parte do endereço é representada por um ***número de porta,*** e requer 3 dígitos adicionais (ou 8 bits). Em TCP/IP um computador na rede pode ter muitas conexões de transporte (**TCP** -*Transmission Control Protocol* - ou **UDP** - *User Datagram Protocol*) a qualquer momento. Cada uma destas conexões é diferenciada das demais por um número de porta, usada como uma espécie de número de caixa postal para os datagramas que chegam a um computador destinatário. Quando um programa de usuário abre uma conexão TCP ou UDP, ele é conectado a uma porta no computador local. Podem existir vários processos de aplicações utilizando conexões TCP ou UDP em uma única máquina, e os números de portas distinguem estes processos para os pacotes que chegam. A aplicação pode especificar uma porta, geralmente ao tentar alcançar algum serviço via um ***número de porta bem-conhecida***, ou pode permitir que o próprio sistema operacional escolha o número de porta a partir do próximo número de porta disponível. Quando um pacote é recebido e passado para o módulo TCP ou UDP, ele é direcionado para o processo de aplicação pertinente, com base na definição do número de porta no pacote. A seqüência***: (endereço-IP-de-origem, porta-de-origem, endereço-IP-de-destino, porta-de-destino)*** identifica de maneira única cada conexão entre sistemas na rede.

Devido à sua extensão, não é interessante para os usuários finais que sejam utilizados endereços (mesmo em formato decimal) para referenciar objetos na rede. Idealmente, os usuários (pessoas ou processos de aplicações) deveriam ser referenciados através de nomes

simbólicos, e não através de endereços numéricos. Assim, da mesma maneira que os usuários dos sistemas de telefonia utilizam as listas telefônicas (ou diretórios de telefones) para encontrar o número de uma pessoa conhecida, os serviços de nomes são usados para encontrar o endereço de um destinatário em particular (usuário final ou processo de aplicação). Ou seja, os serviços de tradução de nomes para endereços e vice-versa são primordiais para facilitar a vida dos usuários finais. Este ***serviço de resolução de endereços*** (*address resolution service*) [Com91a] é uma das funcionalidades fornecidas pelos serviços de nomes mais freqüentemente usadas.

Desta forma, a grande funcionalidade no uso de nomes fica por conta do mapeamento de endereços muito longos. Adicionalmente, o uso de nomes isola os usuários (ou processos de aplicações) de qualquer conhecimento aprofundado de detalhes sobre a configuração da rede, a qual, evidentemente, pode mudar. Por exemplo, um segmento de rede local pode ser adicionado ou removido de uma inter-rede, e este fato deve ser transparente para os usuários/processos de aplicações. Devemos considerar ainda as situações onde um usuário/processo de aplicação deve estar apto a poder migrar de uma localização na rede para outra, sem que outros usuários/processos de aplicação tenham que tomar conhecimento deste fato. Para permitir estas eventualidades, o serviço de nomes deve fornecer além do serviço de resolução de endereços, funcionalidades que permitam que o conteúdo do diretório (as entradas com a lista dos nomes simbólicos e endereços correspondentes) seja alterado e atualizado de maneira controlada. A esta ligação de um nome simbólico com o seu endereço na rede para um usuário/processo de aplicação chamamos de ***"binding***" ([Hal93b] [Hun92a]).

2.2 Como Funcionam os Serviços de Nomes

Do ponto de vista histórico, os usuários de um sistema de correio eletrónico eram endereçados simplesmente pelos seus nomes. Um usuário na rede poderia mandar uma mensagem para outro usuário por meio do comando: *% mail usuário <mensagem>.*

Vamos supor que *suzana* seja um usuário do sistema remoto UECE. Para que pudéssemos enviar uma mensagem para este usuário remoto, poderíamos adicionar o nome do sistema remoto (UECE) como um prefixo de localização na descrição do usuário, de tal forma que o comando passaria para: *% mail uece!suzana <mensagem>.*

Na realidade, este formato de comando é usado pelo ***UUCP*** *(Unix-to-Unix-CoPy*) [OT92], um utilitário bastante conhecido em ambientes Unix, usado para permitir que sistemas remotos troquem mensagens entre si. Este tipo de nomenclatura é adequado para pequenos segmentos de redes, porém, em redes maiores, é comum nos depararmos com comandos como: *% mail mec!cnpq!rnp!uece!dc!suzana <mensagem>*, que indica um caminho por onde a mensagem deve passar para ser despachada para o destinatário final Suzana, e que certamente representa uma forma muito inconveniente de se localizar recursos ou usuários na rede.

Para evitar estes tipos de complicações, foram desenvolvidos módulos de software para registrar os sistemas hospedeiros e os sítios em uma rede, através do conceito de ***nomes de domínios*** *(domain names*). Em sistemas que implementam os *nomes de domínios*, todos os sistemas hospedeiros interrelacionados são agrupados em ***domínios***. Cada domínio é dividido em sub-domínios, formando uma ***hierarquia de domínios***. Idealmente, um sistema de

roteamento de mensagens só precisa saber como alcançar o domínio mais alto na hierarquia, pois este domínio, chamado de ***domínio "raiz",*** irá indicar a localização do usuário de interesse. Ou seja, fica por conta do domínio mais alto na hierarquia descobrir como alcançar o sub-domínio desejado e, por sua vez, fica por conta deste sub-domínio alcançar cada um de seus próprios sub-domínios, e assim por diante.

Um endereço equivalente ao anterior (*mec!cnpq!rnp!uece!dc!suzana)*, utilizando o conceito de *nomes de domínios*, e não um caminho de roteamento explícito, poderia ser descrito simplesmente por: *suzana@dc.uece.br*. Este endereço especifica que Suzana pode ser encontrada na máquina *dc*, no domínio *uece.br* o qual está assinalado para uma universidade, a Universidade Estadual do Ceará.

3. O Domain Name System - DNS

O **DNS** (*Domain Name System*) ([Goe93a] [Goe93b] [Com91a] [Com93a] [Mal92a] [Mal92b] [Hal93b] [Ste91] [Ros91] [Dav89]) é um dos componentes básicos da ***Internet***. Podemos considerá-lo como um sistema de banco de dados distribuído a nível de mundo, que traduz nomes de estações em endereços, e vice-versa, fornecendo informações de roteamento e informações sobre domínios. O DNS é essencial para cada sítio que se conecta diretamente à Internet. Porém, mesmo para redes locais isoladas que utilizem protocolos da pilha TCP/IP, sempre faz sentido que se utilize o DNS.

O rápido crescimento da Internet levou à introdução de domínios e sub-domínios DNS. Os domínios compõem a estrutura do DNS e podem ser comparados a diretórios em um sistema de arquivos, com cada domínio aparecendo como sub-árvores de informações no espaço de nomes de domínios. Um nome de estação ("*host*") em um domínio deve ser único, porém, este mesmo nome pode existir em outros domínios. Os domínios são implementados como hierarquias, e podem ser vistos como uma árvore invertida começando no ***domínio raiz*** (cujo nome real é " ", ou seja, um *label* nulo).

O domínio raiz é atendido por um grupo de servidores de nomes chamados ***servidores raízes***. Os servidores raízes só possuem informações completas a respeito dos ***domínios de topo***, que são aqueles domínios que vêm imediatamente abaixo do domínio raiz na árvore de hierarquia de domínios. Estes domínios de topo por sua vez têm apontadores para os servidores em sub-domínios de níveis inferiores. Nenhum servidor, nem mesmo os servidores raízes, possuem informações completas sobre todos os domínios, mas os apontadores podem indicar-lhes que servidores fornecem a informação desejada. Assim, os servidores raízes podem até não saber a resposta para uma determinada consulta, mas certamente saberão para que outro servidor direcioná-la. A figura 1 [Goe93a] mostra uma parte da árvore de domínios da Internet.

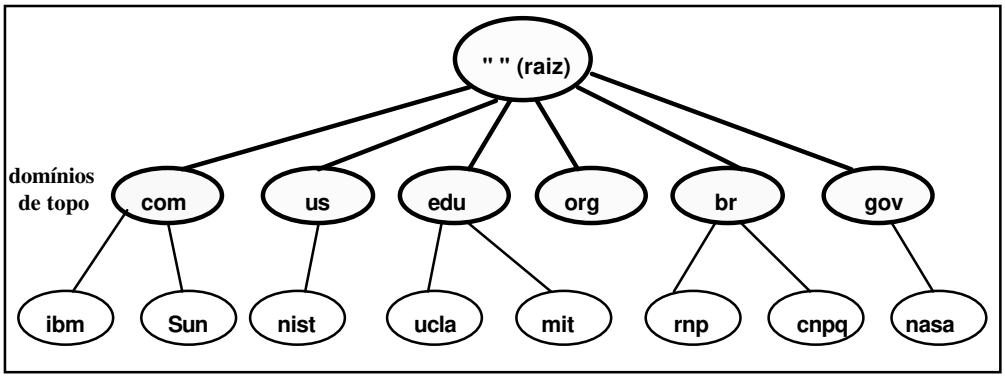

Figura 1. Uma parte da árvore de domínios da Internet

Na Internet, foram inicialmente criados seis ***domínios topo*** de hierarquia: ***.com*** para sítios comerciais; ***.edu*** para instituições educacionais; ***.gov*** para instituições governamentais americanas; ***.mil*** para agências militares; ***.net*** para centros de suporte Internet; e ***.org***, para a administração de redes de outras organizações. Domínios topo são aqueles domínios que vêm imediatamente após o domínio "raiz". Atualmente, a maioria dos sub-domínios faz parte ou do domínio topo ***.com*** ou ***.edu***, sendo que foram criados outros domínios topo, como os domínios nacionais ***.us*** para Estados Unidos, ***.es*** para Espanha, ***.br*** para Brasil, ou ***.ca*** para Canadá (são nomes padronizados pela norma ISO 3166 [FA94]), conforme ilustrados na tabela 1.

Nome dos Domínios	*Significado*
COM	Organizações Comerciais
EDU	Instituições Educacionais
GOV	Instituições Governamentais
MIL	Grupos Militares
NET	(Internet) Centros de Suporte da Rede
ORG	Outras Organizações
INT	Organizações Nacionais (internas)
USA	
BR	
|	
|	Códigos de países para atribuições geográficas
|	

Tabela 1. Nomes de domínios na Internet

Cada um destes domínios, por sua vez, utiliza uma hierarquia de nomes apropriada. Por exemplo, no domínio ***.edu,*** o próximo nível na hierarquia são os nomes das diferentes instituições educacionais que o compõem, enquanto no domínio ***.com*** estão as organizações comerciais. Assim, cada um destes domínios topo são subdivididos em sub-domínios, que por sua vez podem chegar até 127 outros níveis de sub-domínios. Parte da hierarquia de domínios e as convenções de nomes na Internet estão ilustrados na figura 2. Na prática, o endereçamento de domínios da Internet é o mais largamente usado em todo o mundo.

3.1 Como Funciona o Endereçamento de Domínios na Internet

No esquema de nomes de domínios do DNS, cada nó na Internet recebe um endereço numérico único: o endereço IP. Por exemplo, *estação-B.dsc.ufpb.br* corresponde ao endereço numérico ***150.165.1.1*** (nota: usamos neste artigo valores fictícios para os endereços IP dos

exemplos). Este endereço numérico pode ser usado tanto em segmentos de redes locais que utilizam os protocolos da pilha TCP/IP quanto na própria Internet. Endereços numéricos na Internet, assim como seus nomes de domínios, são assinalados pela organização *Network Information Center* (**NIC**) ([Goe93a] [Goe93b] [Qua90]), a qual é administrada pela empresa Government Systems, dos Estados Unidos.

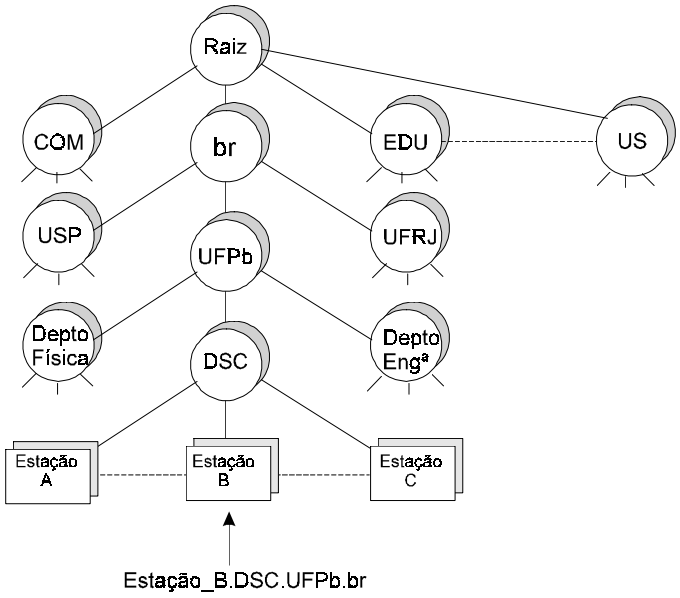

Figura 2. Hierarquia de nomes para domínios Internet

Se um determinado usuário Internet desejar enviar uma mensagem para *estação-B.dsc.ufpb.br*, ele deverá primeiramente requisitar a tradução do endereço simbólico (estação-B.dsc.ufpb.br) para o endereço IP correspondente (150.165.1.1). Esta tradução é executada por um programa chamado **named** ("*name daemon*") [FA94]. O programa *named,* no sítio de origem, requisita de ***servidores raízes*** o nome do servidor que mantém o domínio *ufpb.br*. Digamos que a resposta seja ***servidorX***. A seguir, o sítio de origem requisita a este servidor ***servidorX*** a tradução do nome *estação-B.dsc.ufpb.br* e recebe como resposta o endereço IP correspondente, o qual será usado para estabelecer a conexão.

Na prática, tanto nomes de estações quanto endereços numéricos podem ser usados em comandos de utilitários da pilha TCP/IP. Por exemplo, um usuário que deseja emular uma estação no endereço IP 150.165.1.1 pode digitar o comando: *% telnet 150.165.1.1*, ou então usar o nome simbólico da estação associado a este endereço no comando equivalente: *% telnet estação-B.dsc.ufpb.br.*

Não interessa para o serviço de nomes se o endereço foi fornecido através de um nome simbólico ou através de um endereço IP. A conexão da rede será sempre estabelecida levando-se em consideração o endereço IP. O serviço de nomes converte o nome da estação para um endereço antes que a conexão da rede seja estabelecida. Os administradores de redes são os responsáveis pela criação de nomes simbólicos e endereços locais, e são também responsáveis pelo seu armazenamento nos bancos de dados correspondentes.

Em uma rede tão abrangente quanto a Internet, onde existem poucas pessoas para administrarem tantos detalhes que envolvem toda a rede (de alguns milhões de usuários), faz sentido que se utilize o enfoque hierárquico para nomes de domínios. Cada organização ligada

à Internet é responsável pela manutenção de seus próprios servidores de nomes e de seus próprios sub-domínios, através de seu administrador de rede local. Quando surge a necessidade de se usar endereços remotos, os servidores raízes passam adiante os requisitos para os domínios corretos.

A tradução de nomes para endereços não é uma simples tarefa local [Dav89]. Por exemplo, o comando *% telnet estação-B.dsc.ufpb.br* deve funcionar corretamente para qualquer estação que estiver conectada à rede. Se *estação-B.dsc.ufpb.br* estiver conectada à Internet, todos os nós no mundo estarão habilitados a traduzir o nome *estação-B.dsc.ufpb.br* para o endereço IP apropriado. Portanto, é necessário que exista uma facilidade que possa propagar as informações sobre nomes simbólicos para todos os sítios que compõem um sub-domínio em particular. Esta é uma das funções mais importantes do **DNS,** que representa um sistema completo de nomes para a pilha de protocolos TCP/IP.

3.2 Características do DNS

O DNS é um sistema perfeitamente escalável. Ele não se baseia em uma única tabela; é na realidade um sistema de banco de dados distribuído que não se degrada à medida que as bases de dados crescem. O DNS garante que a propagação das novas informações a respeito de novos sítios seja efetuada para o restante da rede à proporção que estas forem sendo requisitadas.

Além de propagar as informações automaticamente, estas informações são propagadas somente para quem estiver interessado nelas. Vejamos como funciona o DNS: se um servidor DNS recebe um pedido sobre informações a respeito de uma estação, este servidor passa adiante o pedido para um ***servidor autorizado*** ([Com91a] [Goe93a]). Um servidor autorizado é qualquer servidor que seja responsável pela manutenção de informações precisas e atuais sobre o domínio que estiver sendo pesquisado. Quando um servidor autorizado emite uma resposta, o servidor local salva (em *cache*) esta resposta para uso futuro. Na próxima vez que o servidor local receber um pedido por esta informação, ele mesmo fornecerá a resposta.

A habilidade de controlar as informações a respeito de estações a partir de uma fonte autorizada, e também a possibilidade de propagar informações precisas de maneira automática tornam o DNS um serviço de alto nível, mesmo para pequenas redes que não estejam conectadas à Internet.

3.3 A Criação de Domínios e Sub-domínios DNS

A autoridade para alocar domínios é de responsabilidade do **NIC** (*Network Information Center*) ([Goe93a] [Goe93b] [Qua90]). Para conseguir um domínio autorizado, uma organização deve candidatar-se ao NIC para que ele possa designar um novo domínio abaixo de um dos domínios de topo. Depois de conseguir a autorização, a organização pode então criar domínios adicionais, chamados de sub-domínios, abaixo daquele novo domínio recém fornecido pelo NIC.

Como exemplo, vamos supor que a empresa Softech, em Campina Grande, na Paraíba, tenha se candidatado para conseguir um domínio junto ao NIC. A Softech é um instituição comercial com fins lucrativos e, portanto, encaixa-se claramente no domínio de topo ***.com***. A

empresa encaminhou os documentos necessários para o NIC na tentativa de conseguir um domínio chamado *softech* abaixo do domínio de topo *.com*. Os documentos enviados contêm os nomes simbólicos de pelo menos duas estações (seus endereços serão fornecidos pelo próprio NIC) escolhidas pela empresa para fornecer serviços de nomes no novo domínio. Quando o NIC aprova o pedido deste novo domínio, ele adiciona registros apontadores ao domínio de topo *.com,* os quais irão apontar para os novos servidores de nomes no domínio recém autorizado. A partir deste momento, quando consultas forem recebidas pelos servidores raízes para o domínio *softech.com*, elas serão direcionadas para estes novos servidores de nomes.

A aprovação do novo domínio por parte do NIC garante completa autoridade à Softech sobre os novos domínios que forem criados abaixo de *softech.com*. Qualquer domínio registrado tem autoridade para dividir-se em sub-domínios. No caso da Softech, podemos criar domínios separados para o departamento de planejamento (*pnjto.softech.com*), e para o departamento de engenharia (*enga.softech.com*). Isto pode ser feito sem se consultar o NIC. A decisão de se adicionar sub-domínios fica completamente a cargo do administrador do domínio local.

O assinalamento de nomes é similar ao assinalamento de endereços. O NIC assinala um endereço IP para uma organização, e a organização cuida de assinalar endereços de sub-redes e de estações dentro da abrangência do endereço original fornecido pelo NIC. Similarmente, o NIC assinala um domínio para uma organização, e a própria organização cuida de assinalar sub-domínios e nomes de estações naquele domínio. O NIC é a autoridade central, e é quem delega autoridade e distribui o controle sobre nomes e endereços para as organizações individualmente. Desde que esta autorização tenha sido delegada, as organizações por si são responsáveis pelo gerenciamento de nomes e endereços que lhes sejam assinalados.

Um novo sub-domínio torna-se acessível quando apontadores para os servidores no novo domínio são criados nos domínios acima dele, conforme mostrado na figura 3.

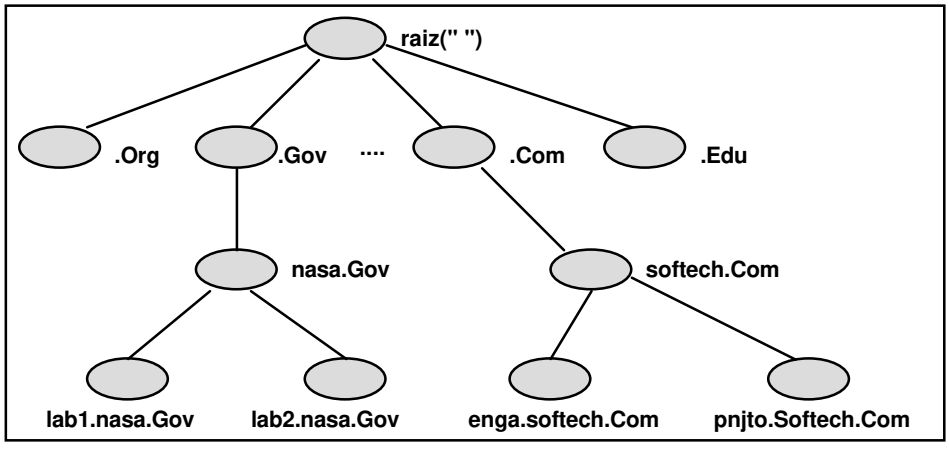

Figura 3. A hierarquia de domínios

Servidores remotos não estarão habilitados para acessar o domínio *softech.com* de nosso exemplo enquanto um apontador para seus servidores de nomes não for criado no domínio *.com*. Da mesma forma, os sub-domínios *.enga* e *.pnjto* não poderão ser acessados enquanto apontadores para eles não forem criados em *softech.com*. Os registros de banco de dados do DNS que apontam para os servidores de nomes de um domínio são chamados de

registros NS (*Name Server*) [Hun92a]. Estes registros contêm o nome dos domínios e o nome da estação correspondente que é servidora de nomes para cada domínio. O exemplo mostrado na figura 4 ilustra como os registros NS são usados como apontadores.

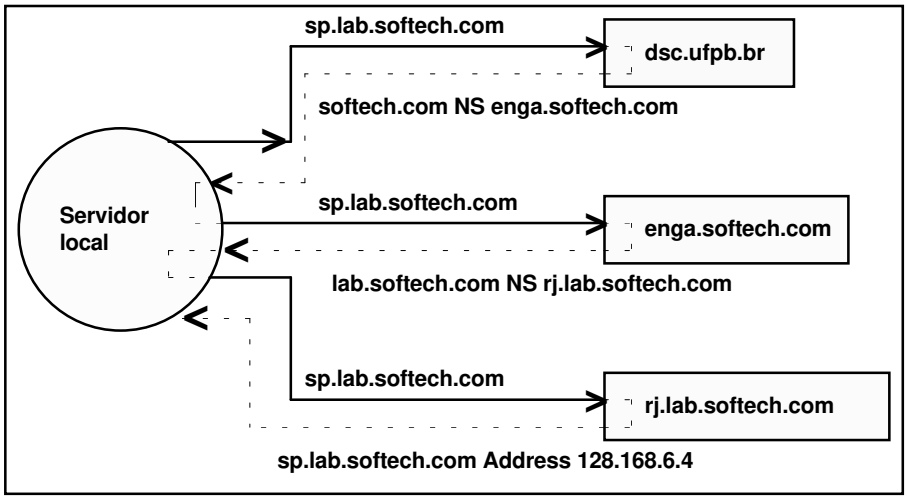

Figura 4. Um busca não-recursiva

Um servidor local faz um pedido para traduzir (resolver) o nome simbólico *sp.lab.softech.com* para um endereço IP. O servidor local (de um domínio, digamos, em uma outra empresa de Campina Grande) não possui informações sobre *softech.com* em sua *cache*, e portanto, ele consulta um servidor raiz *dsc.ufpb.br* (o servidor de nomes na Universidade Federal da Paraíba) para descobrir o endereço procurado. Este servidor raiz responde com um registro NS (um apontador) que aponta para *enga.softech.com* como a fonte de informações sobre *softech.com*. O servidor local então consulta o servidor *enga.softech.com*, que por sua vez aponta para *rj.lab.softech.com* indicando o servidor para *sp.lab.softech.com*. O servidor local consulta agora *rj.lab.softech.com* e finalmente recebe o endereço IP desejado.

O servidor local guarda em *cache* o registro de endereço e cada um dos registros NS. A próxima vez que ele for consultado a respeito de *sp.lab.softech.com*, ele mesmo cuidará de responder à consulta. E todas as vezes que o servidor venha a receber consultas sobre outras informações no domínio *softech.com*, ele irá diretamente para *enga.softech.com*, sem ter que passar por outros servidores raízes.

A figura 4 ilustra um exemplo para uma ***busca não-recursiva*** [Hun92a]. Em uma busca não-recursiva, o servidor remoto indica ao servidor local quem deve ser o próximo servidor a ser consultado. O servidor local deve então seguir os apontadores por sua própria conta. Em uma busca recursiva, o servidor remoto segue ele mesmo os ponteiros, e retorna o resultado final para o servidor local. Os servidores raízes executam somente buscas não-recursivas.

3.4 Nomes de Domínios no DNS

Os nomes de domínios devem refletir a hierarquia de domínios. Os nomes de domínios são escritos a partir do nome mais específico (o nome de uma estação) para o menos específico (um domínio de topo), com cada parte do domínio separada por pontos. Um *nome de domínio totalmente qualificado* (**FQDN** - *Fully Qualified Domain Name*) ([Hal93b] [Ros91]) começa com o nome de uma estação específica e termina com o nome de um domínio de topo. Por

exemplo, *estação22.pnjto.softech.com* é o FQDN para a estação *estação22* no sub-domínio *pnjto*, no domínio *softech*, do domínio de topo *com*.

Na prática, nomes de domínios são raramente escritos na forma totalmente qualificada. Eles são geralmente escritos com relação ao *domínio default* [FA94] estabelecido para uma estação. O próprio DNS cuida de adicionar o domínio default ao nome fornecido pelo usuário, quando são construídos os requisitos para o servidor de nomes. Por exemplo, se definirmos como domínio default *softech.com* para uma determinada estação, o usuário poderá omitir esta extensão para quaisquer referências a nomes neste domínio. Assim, uma referência a *pnjto.softech.com* poderia ser feita simplesmente através de *pnjto*, e o DNS cuidaria de adicionar o domínio *softech.com* como sufixo.

3.5 BIND, Resolver e Named

A implementação do software de serviços de nomes do DNS na maioria dos sistemas Unix é baseada no software **BIND** (*Berkeley Internet Name Domain*) ([Hal93b] [Hun92a]). O BIND é dividido conceitualmente em dois componentes:

- *resolver*: responsável pela montagem de consultas; e
- *servidor de nomes*: processo que atende as consultas, fornecendo as respostas apropriadas.

De fato, o resolver não existe como um processo distinto rodando no computador. Ele é na realidade uma biblioteca de rotinas de software (chamada de "***código do resolver***") que é ligada (“*linkeditada”*) a qualquer programa que deseje procurar endereços. Toda estação disponibiliza o código do resolver, porém nem todas rodam o processo servidor de nomes. Uma estação que não roda um processo servidor de nomes local e depende de outras estações para atender aos seus pedidos ao serviço de nomes é chamada de ***sistema somente-resolvedor*** ("*resolver-only system*"). As configurações somente-resolvedoras são mais comuns em estações isoladas da rede. No entanto, a grande maioria das estações na rede rodam um processo servidor de nomes localmente.

Por outro lado, o servidor de nomes roda como um processo distinto chamado **named** *("name daemon*"), a respeito do qual já falamos rapidamente na seção 3.1. Dependendo de como estejam configurados, os servidores de nomes podem ser classificados de diferentes maneiras. As três principais categorias de servidores de nomes são ([Com91a] [Com93a]):

- *servidor primário*: é o servidor a partir do qual todos os dados a respeito de um domínio são originados. A carga inicial de suas informações a respeito de domínios é feita a partir de um arquivo texto criado pelo administrador de sistemas, obedecendo a uma sintaxe em particular. Os servidores primários são *autorizados*, isto é, possuem informações completas e atualizadas a respeito de seus domínios. Só pode existir um único servidor primário responsável por um determinado domínio;
- *servidor secundário*: os servidores primários transferem a base de dados completa a respeito de um domínio para servidores secundários. Uma base de dados particular para um determinado domínio é chamada de ***arquivo de zona***. A cópia do arquivo de zona do servidor primário para servidores secundários chama-se de *transferência de arquivo de zona*. A garantia de que os servidores secundários estejam sempre atualizados com relação

a um domínio em particular dá-se por meio da transferência de arquivos de zona periódica deste domínio. Os servidores secundários são autorizados somente para seus domínios; e

- *servidor de caching*: recebe as respostas para todas as consultas que venham de outros servidores de nomes. Quando um servidor de *caching* recebe uma resposta para uma consulta, ele guarda esta informação e a utiliza no futuro para responder ele mesmo a outras consultas que venham a ocorrer. A maioria dos servidores de nomes guardam respostas e as utilizam desta maneira. Servidores de caching são *não autorizados*, isto é, suas informações podem ser incompletas ou desatualizadas, apesar de geralmente poderem estar coerentes.

No DNS, existe somente um servidor primário para cada domínio. Os dados do DNS são fornecidos à base de dados do servidor primário pelo administrador do domínio. Portanto, os administradores têm um controle centralizado sobre as informações de nomes de estações. Este banco de dados controlado de maneira centralizada e distribuído de maneira automática é uma vantagem para redes de qualquer tamanho. Quando um novo segmento de rede é adicionado à inter-rede, só é necessário que se modifique a base de dados do DNS no servidor primário. As informações são a seguir propagadas automaticamente para os outros servidores, seja por meio de transferências de zona totais, seja pelo *caching* de respostas individualmente.

4. Conclusão

Um nome é um identificador que consiste de uma seqüência de caracteres escolhidos de um alfabeto finito, e sua funcionalidade reside na possibilidade de um sistema poder mapear efetivamente este nome ao objeto que ele referencia.

Consideramos um endereço numérico, como por exemplo um endereço IP, como sendo um *nome de baixo nível*. Usuários finais, consideram os nomes de baixo nível pouco significativos e, portanto, preferem referenciar objetos através de nomes de mais alto nível, ou nomes simbólicos. Apesar de nomes de baixo nível serem convenientes por fornecerem representações compactas para especificar a fonte e o destinatário de pacotes que trafegam em uma inter-rede, é muito mais natural para os usuários finais que sejam assinalados nomes pronunciáveis e fáceis de serem lembrados para os objetos.

A forma dos nomes simbólicos é importante porque determina como eles serão traduzidos para nomes de baixo nível e também porque determina como deve ser feito o assinalamento destes nomes simbólicos. Com uma inter-rede pequena, composta por poucas máquinas interligadas, o assinalamento de nomes é tarefa trivial. Em grandes redes corporativas (ou na Internet), onde encontramos centenas ou mesmo milhares de máquinas interconectadas, a escolha de nomes simbólicos pode tornar-se extremamente complicada.

Analisamos neste artigo um esquema para facilitar o assinalamento de nomes simbólicos para um grande conjunto de nós em uma rede, e discutimos um mecanismo para mapear nomes simbólicos em endereços IP e vice-versa. Este esquema de nomes é interessante porque, em primeiro lugar, tem sido usado para assinalar nomes de máquinas em toda a Internet, e em segundo lugar, porque fornece um ótimo exemplo de aplicação distribuída modelada através do paradigma cliente/servidor, empregando um conjunto de servidores distribuídos que cuidam do mapeamento de nomes simbólicos para endereços IP.

O esquema e o mecanismo apresentado e discutido compõem o DNS, um exemplo pragmático para um esquema de nomes hierárquico e distribuído. No DNS, servidores de nomes mapeiam nomes simbólicos para endereços IP. Os clientes começam seus serviços através da tentativa de resolver nomes localmente. Quando um servidor local não consegue resolver um nome, o cliente escolhe iterativamente, através de uma busca não recursiva em uma árvore de domínios, o próximo servidor que consiga resolver o nome em questão.

Referências

[Bar93b] Andrew Bart. Locating and Retrieving Internet Files. UnixWorld,10(6):75-82, July 1993.

[Com91a] Douglas E.Comer. Internetworking with TCP/IP Volume I: Principles, Protocols and Architecture. Prentice-Hall, Englewood-Cliffs, 2nd. edition, NJ, 1991.

[Com93a] Douglas E.Comer. *Internetworking with TCP/IP Volume III: Client/Server Programming and Applications.* Prentice-Hall, Englewood-Cliffs, NJ, 1993.

[CTR92b] CTR. Unix Standards.Tech Report, Computer Technologies Research, Charleston, SC, 1992.

[Dav89] John Davidson. *An Introduction to TCP/IP.* Springer-Verlag, New York, NY, 3rd edition, 1989.

[FA94] D.Frey and R.Adams. !@%#:A Directory of Electronic Mail Addressing and Networks. O' Reilly and Associates, Sebastopol, CA, 4th edition, 1994.

[Goe93a] Felix Goehtgens. Configuring DNS. UnixWorld, 10(7):72-77, August 1993.

[Goe93b] Felix Goehtgens. Configuring DNS part II. UnixWorld,10(9):70-72, September 1993.

[Hal93b] Fred Halsall. Data Communications, Computer Networks and Open Systems. Addison-Wesley, Workingham, England, 3rd edition, 1993.

[Hun92a] Craig Hunt. *TCP/IP Network Administration.* O'Reilly & Associates, Sebastopol, CA, 1992.

[Keh92] Brendan P. Kehoe. *Zen and The Art of the Internet: A Beginner's Guide to the Internet*. Prentice-Hall, Widener University, Chester, PA, January 1992.

[LS90] Eliezer Levy and Abraham Silberschatz. Distributed File Systems: Concepts and Examples. *ACM Computing Surveys,* 22(4):321-374, December 1990.

[Mal92a] Carl Malamud. Analyzing SUN Networks. Van Nostrand Heinhold, New York, NY, 1992.

[Mal92b] Carl Malamud. STACKS: Interoperability in Today's Computer Networks. Prentice-Ha11, Englewood-Cliffs, NJ, 1992.

[OT92] Tim O'Reilly and Grace Todino. Managing UUCP and Usenet. O'Reilly & Associates, Sebastopol, CA, 10th. edition, January 1992.

[Qua90] John S. Quaterman. THE MATRIX: Computer Networks and Conferencing Systems Worldwide. Digital Press, 1990.

[Ros91] Marshall T.Rose.*An Introduction to Management of TCP/IP Based Internets.* Prentice-Hall, Englewood-Cliffs, NJ, 1991.

[Ste91] Hal Stern. Managing NFS and NIS. O'Reilly & Associates, Sebastopol, CA, 1991.

[Tan92] Andrew S. Tannenbaum. Modern Operating Systems. Prentice-Hall, Englewood-Cliffs, NJ,1992.

[Tei95] J Helvécio Teixeira Jr. *Compêndio de Computação Distribuída*. Dissertação de Mestrado. Universidade Federal da Paraíba, junho de 1995.

[TVR85] Andrew S. Tannembaum and Robert Van Renesse. Dístributed Operating Systems. *ACM Computing Survey*, 17(4):419-470, December 1985.