Domain Name Service

Configuração e Administração

Rubens Queiroz de Almeida
(queiroz@unicamp.br)
11 de julho de 2000

1 Introdução

A grande popularização da Internet ocorrida nos últimos anos tem gerado uma grande demanda por profissionais capazes de manter a infraestrutura criada para acolher o número cada vez maior de usuários.

Além da própria Internet, temos também que considerar a crescente adoção do uso de intranets por parte das empresas. Como uma intranet utiliza basicamente as mesmas tecnologias utilizadas na Internet global, novamente nos vemos em busca de profissionais com conhecimentos dos protocolos TCP/IP e aplicações.

Infelizmente, exatamente devido à explosão da demanda por estes profissionais, temos situações em que uma quantidade enorme de redes TCP/IP é administrada por pessoas sem o conhecimento técnico adequado à esta tarefa.

Quais são os conhecimentos mínimos que um administrador de redes TCP/IP deve dominar para realizar suas tarefas competentemente? Primeiramente, é importante uma compreensão do funcionamento dos protocolos TCP/IP. Em segundo lugar, a configuração dos serviços correio eletrônico e a configuração dos serviços de resolução de nomes através do DNS (Domain Name Services).

O conhecimento dos protocolos TCP/IP é fundamental para resolver problemas que ocorrem no dia a dia da administração de uma rede de computadores. Quem tem um pouco de vivência com computadores sabe que problemas realmente acontecem. Saber configurar serviços de correio eletrônico é também fundamental. O correio eletrônico ainda é a aplicação mais popular da Internet. Qualquer administrador de redes que deixe seus usuários sem acesso ao correio eletrônico por mais do que algumas horas vai ouvir o que não quer. E finalmente temos o DNS, que será abordado neste livro. Quando o DNS não funciona corretamente é o caos completo. Qualquer programa que utilize o nome, ao invés do número IP, para es-

tabelecer conexão com outro computador na Internet ou em uma Intranet, não irá funcionar. Os exemplos são vários, mas podemos citar o próprio correio eletrônico, acesso à Web e a transferência de arquivos utilizando-se FTP (File Transfer Protocol).

Embora a administração competente de redes baseadas em protocolos TCP/IP requeira conhecimentos profundos sobre diversos tópicos, é perfeitamente possível, com alguns conhecimentos básicos sobre as áreas principais citadas (protocolos TCP/IP, correio eletrônico e DNS), manter a rede funcionando e os usuários satisfeitos.

1.1 Público Alvo

Este livro se destina a administradores de redes, grandes ou pequenas, conectadas à Internet. Provedores de acesso à Internet, universidades, empresas em geral, enfim, qualquer instituição que utilize a Internet global para qualquer fim.

O objetivo deste tutorial é justamente prover ao leitor todas as informações necessárias ao conhecimento básico do funcionamento do DNS para que possa configurar e identificar problemas no funcionamento do servidor de nomes. de nomes (nameserver).

De forma a manter a objetividade deste tutorial, pressupomos que o leitor possua conhecimentos básicos dos protocolos TCP/IP. Mesmo que o leitor não possua esta fundamentação teórica, é perfeitamente possível compreender a maior parte dos tópicos aqui abordados e realizar a configuração completa de um servidor DNS.

Assume-se que o leitor possua um bom conhecimento de computação em geral e dos protocolos TCP/IP em particular.

1.2 Organização

1.2.1 Escopo

Todo este tutorial foi desenvolvido em ambiente Unix (Debian Linux, versão 2.0.34). Independentemente deste fato, toda os conceitos expostos neste livro aplicam-se também a qualquer outro sistema operacional, que implemente os protocolos TCP/IP e que funcione como um servidor de nomes.

1.2.2 Convenções Adotadas

- Itálico
  Utilizado para representar nomes de arquivos, diretórios e nomes de computadores.
- Negrito
  Utilizado para representar comandos emitidos pelo usuário
- Courier
  Este fonte é utilizado para representar o resultado de comandos emitidos, ou o conteúdo de arquivos
- Courier Negrito
  Este fonte é utilizado para indicar comandos digitados pelo usuário
- %,#
  Para indicar as situações em que os comandos devem ser emitidos por usuários comuns ou pelo superusuário (root), utilizamos os caracteres a convenção utilizada em sistemas Unix para indicar (ou alertar) ao usuário que se a conta sob a qual os comandos estão sendo emitidos é ou não privilegiada.

- opção
  Ao exibir comandos, os caracteres delimitados por [], indicam alternativas opcionais que podem ser fornecidas.

2 Introdução ao DNS

Você já parou para pensar como o seu computador é bem relacionado? No seu browser Web, basta digitar o nome de qualquer computador existente na Internet, que instantaneamente a conexão é efetuada. Seu computador conhece todos eles. Mas como isto é possível? O seu computador não passa de um velho 486 (ou pior), com 16MB de memória. E a Internet já possui milhões de computadores conectados. Como pode ele conhecer e conversar com todos eles?

O segredo está no DNS, ou Domain Name Service. Este é o protocolo que torna possível que qualquer computador encontre qualquer outro dentro da Internet em questão de segundos (ou muito menos do que isto). O seu computador pessoal faz uma pergunta a um outro computador que por sua vez se encarrega de encontrar a informação que você precisa, também fazendo perguntas a outros computadores.

Mas vamos voltar um pouco mais no tempo, aos primórdios da Internet. Naquele tempo, existiam poucos computadores na Internet (que nem se chamava Internet ainda). Além de existirem poucos computadores, as pessoas que cuidavam destes computadores também se conheciam. E existia uma lista, chamada *hosts.txt*, que continha os nomes de todos os computadores existentes. Esta lista na verdade não continha apenas nomes. Ela continha linhas relacionando nomes com números. Isto porque os computadores não se comunicam através dos nomes que possuem e sim por meio de números que lhes são atribuídos. Os chamados números IP (de Internet Protocol, você já deve ter ouvido falar de TCP/IP, não?). Mas é

muito difícil memorizar números, nós seres humanos nos lembramos muito mais facilmente de nomes. O que é mais fácil, lembrar-se que o servidor Web da Unicamp atende pelo nome de *www.unicamp.br* ou que o seu número IP é 143.106.80.11? Como os computadores só se conhecem pelo número, criou-se um mecanismo que permitiu a tradução do nome, usado pelos seres humanos que operam os computadores, para o número que os computadores usam em sua comunicação. E começamos com a lista.

A lista era mantida por uma entidade central, que cuidava da distribuição de números aos computadores que se ligavam à Internet. Sempre que um novo computador aparecia, a nova lista atualizada era distribuída a todos os administradores dos computadores ligados à Internet. Desta forma, cada computador conseguia se comunicar com todos os demais. Bastava olhar em sua lista. A Internet era então uma típica cidade do interior, todos se conheciam diretamente e os novatos eram apresentados a todos que já faziam parte da comunidade.

É claro que nem tudo dura para sempre. A Internet foi invadida por todos os tipos de pessoas e se tornou uma comunidade virtual, um espelho do mundo real. E o velho esquema de manter a listinha, contendo os nomes de todos os computadores passou a não ser mais viável. Afinal de contas, qual computador tem o poder de consultar uma "listinha" de alguns milhões de linhas sempre que precisasse enviar alguma coisa para outro computador na Internet? Certamente não o seu velho e ultrapassado 486. Precisou-se inventar uma outra maneira de fazer com que os computadores se encontrassem, mesmo sem possuir a tal listinha (que já nem era mais listinha).

Foi então que inventaram o DNS. Com o DNS, abolia-se a centralização da informação. Não existe mais um computador na Internet que conheça todos os demais. O que aconteceu foi que a autoridade sobre a informação foi diluída. Desta forma, não existe mais um dono da verdade, a informação está distribuída por milhares de computadores, que conhecem muito

bem apenas alguns computadores. Tomemos o exemplo da Unicamp. Nós temos aqui um computador que tem uma lista de todos os computadores conectados à Internet dentro da Universidade. Qualquer computador na Internet que queira achar algum computador dentro da Unicamp tem que perguntar a este computador. Desde que o computador procurado exista, ele fornece a informação solicitada. Mas e como chegar até a Unicamp? Novamente não é difícil descobrir. Os projetistas do DNS na verdade nada mais fizeram do que imitar a vida real. Imagine que você esteja em uma cidade grande e deseja chegar até um determinado bairro. Você pára alguém na rua e pergunta: "Como faço para chegar até o bairro X?". O seu interlocutor não sabe, mas te diz para ir até determinado lugar e perguntar novamente. E lá vai você perguntando, até que chega em alguém que sabe lhe dizer onde se encontra o local que está procurando.

A tradução de nomes em números na Internet funciona exatamente da mesma forma. Você configura o seu computador com o nome do servidor de nomes local. E é ele quem vai fazendo as perguntas para você, até obter uma resposta. A resposta pode ser o número IP do computador com o qual você quer se comunicar ou uma negativa, dizendo que o computador procurado não existe.

Agora, quando é usado o DNS? Sempre que você usar um programa que usa o nome de um computador o DNS entra em ação. Você está mandando uma mensagem eletrônica para queiroz@unicamp.br. O DNS tem que descobrir para você qual o número IP do computador que recebe mensagens destinadas ao domínio unicamp.br. Você está acessando o servidor Web da Disney. Lá vai o DNS novamente buscar a tradução do nome *www.disney.com* para um número IP. Chat, FTP, e mil outras coisas que você se habituou a usar na Internet, todos fazem uso do DNS.

Por agora você já deve ter visto a importância do DNS no uso dos recursos da Internet. Por isto mesmo não se esqueça que a sua configuração correta é muito importante. A não ser que você saiba de cor o número IP de todos

os computadores da Internet, o que é pouco provável. Afinal de contas nem o seu computador consegue fazer isto e ele é muito melhor do que você para memorizar coisas.

Mas eu sou o responsável pelo DNS onde trabalho. Como faço para colocá-lo para funcionar? Bom, continue lendo...

3 Registro de Domínios

Todo computador ligado à Internet possui um nome e um sobrenome. O nome geralmente é escolhido pela pessoa que usa o computador. Em muitos locais escolhe-se um tema preferido e os computadores são batizados segundo este tema. Por exemplo, os fãs das aventuras do Asterix, podem resolver escolher os nomes dos personagens das histórias em quadrinhos para seus computadores. Temos o obelix, o próprio asterix, abracurcix, ideiafix, chatotorix e mais alguns.

Agora fica a questão do sobrenome. No nome *obelix.unicamp.br*, o nome do computador é *obelix* e o sobrenome é *unicamp.br*. O sobrenome já é um pouco mais complicado de definir e envolve o contato com algumas entidades que regulamentam e controlam a concessão de domínios. Em termos da Internet global, o sobrenome precisa ser registrado. Para que o registro seja concedido não pode haver outra empresa ou pessoa que o tenha registrado anteriormente. O sobrenome, no jargão da Internet, é o que chamamos de domínio. No Brasil, o órgão responsável pelo registro de domínios é a FAPESP (Fundação de Amparo a Pesquisa de São Paulo). Todo o processo, da consulta ao registro do domínio pode ser feito diretamente através do servidor Web da FAPESP, no endereço *http://www.registro.fapesp.br*.

O DNS possui estrutura semelhante a uma árvore de diretórios, tal como a que é encontrada em sistemas Unix ou MSDOS. O diretório de mais alto nível, ou diretório raiz, possui apontadores para os demais diretórios do

segundo nível, os diretórios do segundo nível possuem apontadores para os diretórios do terceiro nível e assim sucessivamente.

Mas para entender melhor tudo o que foi dito, vamos analisar o nome de um computador. Vejamos o nome *obelix.unicamp.br*. Dá para ver que o nome é composto de quatro componentes:

```
obelix + unicamp + br + "."
```

Isto mesmo, quatro componentes. Embora não pareça, o ".", que a maioria de nós nem se lembra de digitar quando escreve o nome de um computador (e muitos de nós nem mesmo sabemos que este ponto existe) representa o domínio de mais alto nível na hierarquia de nomes de computadores. No nome de computadores, a hierarquia (ou domínio) de mais alto nível fica à direita, ao passo que a mais baixa fica à esquerda.

Isto é fácil de se visualizar. O "." contém os servidores de nomes de mais alto nível, que possuem apontadores para os computadores de nível imediatamente inferior, os domínios geográficos, aos quais pertencem o Brasil (br), Japão (jp), Canadá (ca), Portugal (pt) e todos os demais países. Neste mesmo nível situam-se os domínios com (entidades comerciais), net (comunicações), , org (organizações), edu (educação), gov (governo) e mil (militares). Os domínios de segundo nível apontam para os domínios de terceiro nível. A Unicamp, por exemplo, está dentro do Brasil. Dentro do servidor de nomes do domínio br, mantidos pela FAPESP, existe uma referência ao servidor de nomes da Unicamp, que conhece todos os computadores do domínio *unicamp.br*.

Agora vamos ver a situação em que alguém deseje encontrar o computador *acme.com.br*. Ele primeiro faz a pergunta ao seu servidor de nomes local. Este servidor de nomes não conhece o computador *acme.com.br*. O que faz então? Pergunta a outro, neste caso, aos servidores do domínio ".". Estes servidores de nomes também não conhecem o computador *acme.com.br*, mas analisando o nome descobrem que este computador está

dentro do Brasil (br) e instruem o servidor de nomes local a perguntar aos servidores do domínio br. E lá vai o nosso servidor de nomes perguntar aos servidores do domínio *br* onde está *acme.com.br*. Eles (servidores do domínio br) também não sabem, mas sabem que tal computador, se existir, está dentro da empresa chamada Acme. E novamente instruem o servidor de nomes local a perguntar aos servidores de nomes da Acme. Desta vez a resposta é definitiva. O computador *acme.com.br* existe e o número IP correspondente é 200.200.20.1. Acabou a busca. Não foi tão difícil assim. Para localizar o computador *acme.com.br*, dentre os milhões existentes na Internet, foram necessárias apenas quatro perguntas. Antes que vocês saiam procurando o computador *acme.com.br*, ele não existe. Utilizei este nome aqui apenas para ilustrar o conceito.

4 Configuração

4.1 Clientes

Configurar o serviço DNS em um ambiente envolve dois passos principais: a configuração das máquinas clientes e a configuração das máquinas servidoras de nomes, ou *nameservers*.

Os clientes DNS fazem uso de uma biblioteca chamada *resolver*. Esta biblioteca é invocada pelos aplicativos do sistema sempre que se fizer necessária a tradução de um nome em um número IP.

Em sistemas Unix, é preciso criar um arquivo chamado */etc/resolv.conf*. Dentro deste arquivo especifica-se o nome do servidor de nomes, o nome do domínio e opcionalmente, uma lista de argumentos a serem utilizados como sufixo de nomes que não sejam totalmente qualificados.

Vejamos agora um arquivo */etc/resolv.conf* típico:

```
nameserver 200.200.20.1            [1]
nameserver 200.200.30.15           [2]
nameserver 148.100.1.20            [3]
search com.br acme.com.br com      [4]
```

No arquivo acima, a numeração das linhas foi incluída apenas para fins de ilustração e não fazem parte da configuração original.

As linhas de 1 a 3 indicam os servidores de nomes que este cliente pode consultar. As consultas são sempre dirigidas preferencialmente ao primeiro nome da lista, o servidor de nomes cujo número IP é 200.200.20.1. Caso este servidor de nomes não esteja operacional por alguma razão, o cliente consulta então o próximo nome na lista, e assim por diante. Podem ser especificados até 3 servidores de nomes. Servidores de nomes adicionais serão ignorados. Este processo é demorado, visto que para passar de um servidor de nomes para outro, o sistema espera por um determinado tempo. Ao se chegar ao último servidor de nomes, retorna-se então ao primeiro deles e este processo se repete até que o número máximo de tentativas seja feito. Neste momento então o sistema retorna a informação de que não existem servidores de nomes disponíveis.

A diretiva *search* indica os sufixos que devem ser afixados a nomes que não sejam completamente qualificados, ou completos. Um nome totalmente qualificado, ou no jargão da Internet, FQDN (*Fully Qualified Domain Name*), é qualquer nome que não seja terminado em "." . O nome *obelix.unicamp.br.* é totalmente qualificado, visto terminar em "." . Já o nome *obelix* não o é. Na Internet não existe um nome que possua apenas um elemento (apenas *obelix*, por exemplo). Nestes casos, para conveniência dos usuários, o sistema automaticamente afixa os sufixos constantes da diretiva *search* ao nome fornecido. No nosso caso a pesquisa enviada aos servidores de nomes pesquisará os nomes *obelix.com.br*, *obelix.acme.com.br* e *obelix.com*, exatamente neste ordem. Caso existam computadores chamados *obelix.com.br* e *obelix.acme.com.br*, a resposta retornada pelo ser-

vidor de nomes será o número IP do primeiro nome que for traduzido, *obelix.com.br*.

É importante notar que esta facilidade não será utilizada apenas quando o nome fornecido terminar em ".". Caso o nome especificado seja *obelix.com.br*, sem o ponto final, serão pesquisados os seguintes nomes:

```
obelix.com.br.com.br
obelix.com.br.acme.com.br
obelix.com.br.com
obelix.com.br
```

O arquivo */etc/resolv.conf* abaixo

```
domain acme.com.br
nameserver 200.200.20.1
```

não faz uso da diretiva *search*. Neste caso a pesquisa de nomes não qualificados, novamente tomando como exemplo o nome *obelix*, é feita da seguinte forma:

```
obelix.acme.com.br
obelix.com.br
obelix.br
obelix
```

Quando se especificam as diretivas *search* e *domain* simultaneamente, a diretiva *search* tem precedência sobre os valores especificados na diretiva *domain.*

Na inexistência do arquivo */etc/resolv.conf,* o comportamento normal é assumir que o servidor de nomes é o computador local e o nome do domínio

é obtido a partir do nome da máquina. Por exemplo, se o computador foi configurado com o nome *obelix.unicamp.br,* o nome de domínio é obtido a partir do resultado do comando *hostname:*

```
% hostname
obelix.com.br
```

O nome do domínio será com.br, obtido pela remoção do nome *obelix.* O que sobrar é utilizado como o nome de domínio.

4.2 Servidores

Os servidores DNS podem ser divididos em três tipos principais: servidores que apenas armazenam as informações recebidas de outros servidores na Internet, também conhecidos como *caching only*, servidores mestres primários e servidores mestres secundários.

Todo servidor de nomes interage com outros servidores de nomes na Internet para obter as informações solicitadas por seus clientes. Esta informação, uma vez obtida, passa a residir no cache do servidor de nomes. Desta forma, da próxima vez que a mesma informação for solicitada, não mais haverá a necessidade de se consultar outros servidores de nomes. A informação será fornecida a quem a solicitar diretamente a partir do cache local.

Servidores mestres primários possuem autoridade sobre um ou mais domínios. Além das informações mantidas em seu cache, obtidas de outros servidores de nomes, o servidor primário é a fonte de informação oficial a respeito de um domínio. A informação que os servidores mestres primários disponibilizam é lida a partir de arquivos locais, configurados pelo administrador do domínio.

Tomemos por exemplo o servidor de nomes da Unicamp, *ns.unicamp.br.* Este computador fornece informações oficiais a respeito de todos os computadores existentes dentro da Unicamp. Servidores de nomes de todo o mundo, ao desejarem informações sobre algum computador dentro da Unicamp, precisam enviar uma solicitação ao servidor de nomes ns.unicamp.br.

Além destas informações oficiais, o servidor de nomes da Unicamp, possui também informações não oficiais, armazenadas em seu cache. Qualquer servidor primário, além de fornecer as informações oficiais a respeito do domínio pelo qual são responsáveis, possui em seu cache informações não oficiais, obtidas de outros servidores de nomes. Estas informações são consideradas não oficiais visto terem sido obtidas de outros servidores de nomes e, como tal, estão sujeitas a mudanças.

As informações mantidas no cache possuem um prazo de validade. Todo servidor de nomes oficial de um domínio, ao disponibilizar esta informação para outros computadores na Internet, a fornece juntamente com o prazo de validade ou TTL (*Time to Live*). O TTL indica por quanto tempo a informação é válida. Após este tempo a informação deve ser descartada e novamente solicitada junto ao servidor de nomes oficial do domínio. A definição do valor que o TTL deve assumir é decisão do administrador do domínio. O administrador deve considerar o nível de volatilidade das informações sobre seus computadores e especificar um valor compatível para o campo TTL.

Os servidores mestres secundários, como o nome diz, fornecem informações oficiais a respeito de um ou mais domínios. Esta informação, todavia, não é lida a partir de arquivos locais mas transferida via rede do servidor mestre primário. O processo *named*, ao entrar no ar, determina quais os domínios para os quais deve atuar como servidor mestre secundário, e então entra em contato com os servidores mestres primários e solicita a transferência das zonas correspondentes.

Não existe uma necessidade de que um servidor de nomes seja estritamente

primário ou secundário. É perfeitamente possível e bastante comum que um servidor de nomes seja primário para alguns domínios e secundário para outros.

4.3 Arquivos de Configuração

Iremos proceder agora à construção dos arquivos de configuração de um servidor de nomes de um provedor de acesso fictício. A empresa chama-se *NetRoad* e seu domínio na Internet denomina-se *netroad.com.br.*

Este provedor de acesso, em começo de atividades, possui apenas um cliente, uma empresa chamada *NetMasters.* Adicionalmente, estão prestando serviço de servidor mestre secundário para a empresa *NetWizards,* detentora do domínio *netwizards.com.br.*

Este provedor possui os seguintes equipamentos:

- roteador com 16 portas assíncronas
- servidor de nomes
- servidor Web
- servidor FTP
- servidor de Usenet News
- seis micro computadores utilizados pelos funcionários da empresa

4.3.1 Arquivo Mestre: /etc/named.boot

O coração de qualquer servidor DNS é o arquivo */etc/named.boot.* Neste arquivo são descritos os domínios para os quais o servidor é primário ou secundário, o diretório onde os arquivos contendo as informações sobre

as zonas se encontram, quais servidores estão autorizados a transferir as zonas, entre outras informações.

Analisemos então o arquivo */etc/named.boot* do servidor de nomes do provedor *NetRoad:*

```
directory                                       /usr/local/named
primary    netroad.com.br                       p/netroad.db
primary    netmasters.com.br                    p/netmasters.db
primary    20.200.200.IN-ADDR.ARPA              p/200.200.20.0.db
secondary  netwizards.com.br        222.222.22.22 s/netwizards.db
secondary  21.200.200.IN-ADDR.ARPA  222.222.22.22 s/200.200.21.0.db
cache      .                                    named.root
primary    0.0.127.IN-ADDR.ARPA                 127.0.0.db
```

A primeira linha, contendo a diretiva *directory,* indica o diretório base onde se encontram todos os arquivos de configuração do servidor de nomes. O arquivo *named.root* por exemplo, referenciado pela diretiva *cache,* encontra-se na realidade no diretório */usr/local/named/named.root.*

A diretiva seguinte, na segunda linha, indica que o servidor de nomes é responsável pelo domínio *netroad.com.br.* O arquivo localizado em */usr/local/named/p/netroad.db,* contém as informações sobre todos os ccomputadores do provedor conectados à Internet. O nome *netroad.db* é totalmente opcional, ficando a critério do administrador DNS. A localização no subdiretório ".", é feita para sinalizar mais claramente ao administrador do domínio que se trata de um domínio para o qual se é primário. O sufixo *db* é também uma convenção comum na Internet e significa *data base*.

4.3.2 Descrição de Zona: netroad.db

Passemos agora a analisar o conteúdo do arquivo *netroad.db:*

```
netroad.com.br. IN   SOA ns.netroad.com.br. dnsmaster.netroad.com.br. (
                  1998122103 ; Serial
                  10800      ; Refresh
```

```
                1800       ; Retry
                3600000    ; Expire
                259200 )   ; Minimum
;
; Definição dos Servidores Primário e Secundário do Domínio NetRoad.com.BR
;
netroad.com.br.         10800 IN   NS       ns.netroad.com.br.
ns.netroad.com.br.      10800 IN   A        200.200.20.1
netroad.com.br.         10800 IN   NS       ns.netwizards.com.br.
;
; Definição dos Servidores de Email Primário e Secundário
;
netroad.com.br.         10800 IN   MX    10 mail.netroad.com.br.
netroad.com.br.         10800 IN   MX    20 mail.netwizards.com.br.
;
; Definição dos servidores Web, FTP, News
;
www.netroad.com.br.     10800 IN   CNAME ns.netroad.com.br.
ftp.netroad.com.br.     10800 IN   CNAME ns.netroad.com.br.
news.netroad.com.br.    10800 IN   A     200.200.20.2
;
; Definição dos microcomputadores de trabalho do provedor
;
pc01.netroad.com.br.    10800 IN   A     200.200.20.3
pc02.netroad.com.br.    10800 IN   A     200.200.20.4
pc03.netroad.com.br.    10800 IN   A     200.200.20.5
pc04.netroad.com.br.    10800 IN   A     200.200.20.6
pc05.netroad.com.br.    10800 IN   A     200.200.20.7
pc06.netroad.com.br.    10800 IN   A     200.200.20.8
;
; Definição do Roteador e de suas oito portas assíncronas
;
async01.netroad.com.br. 10800 IN   A     200.200.20.65
async02.netroad.com.br. 10800 IN   A     200.200.20.66
async03.netroad.com.br. 10800 IN   A     200.200.20.67
async04.netroad.com.br. 10800 IN   A     200.200.20.68
async05.netroad.com.br. 10800 IN   A     200.200.20.69
async06.netroad.com.br. 10800 IN   A     200.200.20.70
async07.netroad.com.br. 10800 IN   A     200.200.20.71
async08.netroad.com.br. 10800 IN   A     200.200.20.72
```

Cada uma das linhas do arquivo netroad.db é o que se chama de *Resource Record* (RR), ou traduzindo, Registro de Recurso. O arquivo *netroad.db* é o que se denomina de zona. Esta zona contém os registros descritivos do domínio *netroad.com.br.*

No arquivo *netroad.db*, todos os registros são do tipo INternet (IN). Os registros INternet, por sua vez, são de vários tipos. SOA (*Start Of Authority*), NS (*Name Server*), A (Address), MX (*Mail eXchanger*) e CNAME (*Canonical Name*).

O Registro SOA (Start of Authority)

Comecemos analisando o registro SOA:

```
netroad.com.br. IN SOA ns.netroad.com.br. dnsmaster.netroad.com.br. (
                   1999122103 ; Serial
                   10800      ; Refresh
                   1800       ; Retry
                   3600000    ; Expire
                   86400)    ; Minimum
```

Este registro SOA está disposto em várias linhas para facilidade de leitura, mas na verdade é apenas um registro. Observe a abertura de parenteses no final da primeira linha e o fechamento na última linha. Os parenteses permitem que o registro se extenda por várias linhas.

O primeiro valor, *netroad.com.br.*, indica o domínio ao qual as informações do registro SOA se aplicam. Em seguida temos o valor IN, indicando que este é um registro do tipo INternet. Temos então a indicação do tipo de registro Internet, SOA, seguida pelo nome do computador onde esta zona reside, o computador ns.netroad.com.br, seguida pelo endereço eletrônico do administrador desta zona, *dnsmaster@netroad.com.br.* Note bem que no endereço eletrônico, o caracter "." foi substituído pelo caracter ".". Isto se dá devido ao fato de que o caracter "." tem um significado especial, que será abordado mais à frente.

Todos os valores da segunda linha em diante, são utilizados por servidores secundários do domínio *netroad.com.br*. É norma seguida pela maior parte das autoridades que cuidam do registro de domínios na Internet que

cada domínio possua no mínimo dois servidores de nomes atendendo pelo domínio registrado. No Brasil o registro de um domínio somente é aceito se já existirem dois servidores de nomes configurados e fornecendo informações corretas sobre este domínio. O servidor primário é aquele onde os arquivos de configuração são criados e mantidos pelo administrador DNS. O servidor secundário realiza uma cópia destas dados via rede e os grava em seu disco rígido. Os campos que passaremos a discutir agora servem para estabelecer este sincronismo entre servidores primários e secundários.

O primeiro deles, o valor

```
1999122103 ; Serial
```

indica a versão do mapa. Todos servidores secundários, como veremos em breve, são configurados para entrar em contato com o servidor primário regularmente para verificar se houveram mudanças nos mapas descritivos das zonas para qual atende. Os arquivos não são verificados integralmente, verifica-se apenas o número da versão do mapa. Caso a versão do mapa existente no servidor primário seja maior do que a versão que o servidor secundário possui, é então realizada uma transferência de zona, pois isto indica que os dados foram alterados. O servidor secundário solicita então ao servidor primário a transferência dos dados desta zona.

A forma como este número é escrito não segue nenhuma norma fixa. A convenção mostrada aqui, por sinal bastante comum, é utilizada para conveniência do administrador. Os quatro primeiros dígitos, 1999, indicam o ano, os quatro dígitos seguintes indicam o mês, dezembro, e o dia, 21, em que os dados desta zona foram alterados. Os próximos dois dígitos, 03, indicam que esta foi a terceira modificação realizada no dia. Resumindo, este mapa foi modificado três vezes no dia 21 de dezembro de 1999. Esta certamente é uma informação bastante útil para o administrador DNS.

O próximo valor

```
10800       ; Refresh
```

indica de quanto em quanto tempo o servidor secundário deve contactar o servidor primário para verificar se houveram mudança nos dados do domínio para o qual é secundário. Este valor é expresso em segundos, no nosso caso 10.800 ou 3 horas. A cada três horas o servidor secundário do domínio netroad.com.br entrará em contato com o servidor primário e verificará se o número da versão do mapa que possui está ou não atualizada. Caso não esteja, é solicitado ao servidor primário a transferência da zona.

Em seguida temos o valor

```
1800        ; Retry
```

que indica quanto tempo o servidor secundário deve aguardar para tentar novamente uma conexão com o servidor primário quando houver uma falha de comunicação. Como configurado, o servidor secundário deve contactar o servidor primário a cada três horas. Caso uma destas conexões falhe, uma nova tentativa deve ser feita dentro de 1.800 segundos, ou 30 minutos. As tentativas se repetem até que seja estabelecida uma conexão.

Depois de 3.600.000 segundos (3600000 ; Expire), ou 41 dias, o servidor secundário desiste então de contactar o servidor primário, e expira todos os dados relativos ao domínio cujo servidor primário está fora do ar. Todos os arquivos relativos à zona expirada são apagadas e deste momento em diante o servidor secundário não mais responde perguntas sobre este domínio.

O próximo valor

```
86400)      ; Minimum
```

indica o valor mínimo, ou TTL (*Time to Live*) aplicado aos registros (RR) deste arquivo. No nosso exemplo este valor é de 86.400 segundos ou um

dia. Todas as informações fornecidas pelos servidores (primários ou secundários) do domínio *netroad.com.br* a outros servidores será mantida no cache dos servidores que solicitaram esta informação por apenas um dia. Após 24 horas a informação é expirada e removida do cache. Solicitações posteriores devem novamente ser obtidas junto aos servidores do domínio *netroad.com.br*.

Os valores utilizados são utilizados apenas a título de ilustração. O certo é que cada administrador determine os valores mais adequados para sua situação específica. Todos os valores devem ser analisados e configurados em conformidade com o que for mais adequado. O valor do TTL, em nosso exemplo de 86.400 segundos, certamente não é o mais adequado em ambientes onde as informações forem mais estáveis. Talvez um valor mais adequado seja uma semana ou até mais. Novamente, em um ambiente onde os dados são estáveis, o período de atualização (refresh) pode ser maior que três horas. O valor de um dia ou até mesmo mais tempo pode ser adequado. Não existe uma receita fixa, tudo depende do bom senso do administrador DNS. Não utilize o *copy-paste* como receita de bom senso.

O Registro NS (Name Server)

Em seguida aparece o registro do tipo NS (NameServer). O registro NS contém, à direita, o nome do domínio, em seguida a indicação de que se trata de um registro do tipo INternet. O valor 10.800 indica o valor em segundos do TTL deste registro. Em outras palavras, estes registros possuem prazo de validade de três horas (muito pouco,não?). Um TTL deste valor nunca deveria ser usado indistintamente em todos os registros. Um valor de três horas para um registro somente deve ser usado em casos especiais. Especialmente em registros do tipo NS este valor deve ser mais alto. Afinal de contas, os servidores de nomes para um domínio devem ser estáveis e sujeitos a pouquíssimas modificações. Consequentemente, um TTL de valor 2592000 (30 dias), não é exagerado.

```
;
; Definição dos Servidores Primário e Secundário do Domínio NetRoad.com.BR
;
netroad.com.br.         10800 IN    NS      ns.netroad.com.br.
ns.netroad.com.br.      10800 IN    A       200.200.20.1
netroad.com.br.         10800 IN    NS      ns.netwizards.com.br.
```

Analisemos então o primeiro registro. À esquerda encontra-se o nome do registro, *netroad.com.br*. No lado direito encontra-se o nome do servidor que atende a este domínio. Assim temos que todas perguntas relativas ao domínio *netroad.com.br* são respondidas pelo computador cujo nome é *ns.netroad.com.br.* Mas temos então um problema. Se todas as perguntas sobre computadores pertencentes ao domínio *netroad.com.br* devem ser respondidas pelo computador *ns.netroad.com.br,* como então achar o computador *ns.netroad.com.br,* que faz parte do mesmo domínio pelo qual responde? Esta é uma típica situação do que veio primeiro, o ovo ou a galinha. Este problema é resolvido acrescentando-se em seguida um registro do tipo A (Address) que informa o número IP do servidor de nomes do domínio *netroad.com.br* (em nosso exemplo o endereço IP é 200.200.20.1). Este tipo de registro é conhecido como registro cola (*glue record*). Este registro é necessário sempre que o servidor de nomes encontra-se dentro do domínio sobre o qual é autoridade.

O domínio *netroad.com.br* possui dois servidores de nomes: *ns.netroad.com.br* e *ns.netwizards.com.br.* Observe que o segundo servidor de nomes não possui um registro "cola" associado. Isto porque o computador *ns.netwizards.com.br* não pertence ao domínio *netroad.com.br*. A inclusão do registro "cola" neste caso é não apenas desnecessária como também desaconselhável. Desnecessária e desaconselhável porque os dados relativos ao domínio do segundo servidor de nomes, *netwizards.com.br,* são gerenciados por outras pessoas. O número IP do servidor de nomes pode mudar. Caso a modificação não seja comunicada aos administradores do domínio *netroad.com.br* vários transtornos podem ocorrer. Informações incorretas serão passadas aos servidores de nomes que fizerem consultas relativas

ao domínio *netroad.com.br.* Pior ainda, em caso de falha do servidor de nomes primário, o secundário não será encontrado e todos os computadores deste domínio ficarão virtualmente isolados. Registros "cola" devem ser utilizados apenas quando estritamente necessário. Não se ganha nada utilizando-se estes tipos de registro indiscriminadamente.

Registros MX (Mail Exchanger)

```
;
; Definição dos Servidores de Email Primário e Secundário
;
netroad.com.br.         10800 IN   MX    10 mail.netroad.com.br.
netroad.com.br.         10800 IN   MX    20 mail.netwizards.com.br.
;
```

Os registros do tipo MX (*Mail Exchanger*) provêm a interação entre o DNS e o correio eletrônico. Novamente, à direita temos o nome do domínio (*netroad.com.br*), o TTL, tipo do registro (INternet), o tipo de registro (MX), a precedência e o nome do servidor de mensagens (*mail.netroad.com.br*). Temos então que o domínio *netroad.com.br* é atendido por dois servidores: *mail.netroad.com.br* e *mail.netwizards.com.br.* A precedência é um número que indica qual servidor tem prioridade no encaminhamento das mensagens. Quanto menor este número maior a prioridade. Mensagens enviadas para qualquer computador dentro do domínio *netroad.com.br* são preferencialmente encaminhadas para o computador *mail.netroad.com.br.* Se este servidor estiver indisponível por algum motivo, as mensagens são então encaminhadas ao servidor MX secundário, o computador *mail.netwizards.com.br.*

Registros A (Address)

Este é o tipo de registro mais frequentemente utilizado e realiza o mapeamento entre endereços IP (Addresses) e nomes.

Registros CNAME (Canonical Name)

Estes registros servem para atribuir diversos nomes diferentes a um mesmo número IP. Em nosso exemplo os nomes *www.netroad.com.br* e *ftp.netroad.com.br* direcionam para um mesmo computador, *ns.netroad.com.br,* cujo número IP é 200.200.20.1.

Um erro bastante comum é apontar, em um registro do tipo CNAME para outro registro do tipo CNAME.

Ainda em nosso exemplo, a configuração

```
www.netroad.com.br.     10800 IN   CNAME ftp.netroad.com.br.
ftp.netroad.com.br.     10800 IN   CNAME ns.netroad.com.br.
```

é inválida, visto que *www.netroad.com.br* está apontando para *ftp.netroad.com.br,* que não é um nome canônico e sim um apelido (alias). Todos os registros CNAME, em nosso exemplo, devem obrigatoriamente apontar para *ns.netroad.com.br* que é o nome verdadeiro (o que é definido com um registro do tipo A).

4.3.3 Descrição de Zona Reversa: 200.200.21.0.db

O nosso provedor de acesso fictício recebeu uma classe C, 200.200.21.0, com 254 endereços, para endereçar os computadores de sua rede e de seus clientes. O mapeamento reverso realiza a tradução de nomes em números IP. Este mapeamento é indicado através de registros do tipo PTR (Domain Name Pointer).

Na configuração de nosso provedor, tal informação, como indicado em */etc/named.boot,* encontra-se descrita no arquivo *200.200.21.0.db*:

```
0.20.200.200.IN-ADDR.ARPA. IN SOA ns.netroad.com.br. dnsmaster.netroad.com.br. (
                  1998122103 ; Serial
```

```
                10800      ; Refresh
                1800       ; Retry
                3600000    ; Expire
                259200 )   ; Minimum
;
; Definição dos Servidores Primário e Secundário do Domínio NetRoad.com.BR
;
netroad.com.br.         10800 IN   NS       ns.netroad.com.br.
ns.netroad.com.br.      10800 IN   A        200.200.20.1
netroad.com.br.         10800 IN   NS       ns.netwizards.com.br.
;
; Definição dos microcomputadores de trabalho do provedor
;
3.20.200.200.IN-ADDR.ARPA. 10800 IN PTR pc01.netroad.com.br.
4.20.200.200.IN-ADDR.ARPA. 10800 IN PTR pc02.netroad.com.br.
5.20.200.200.IN-ADDR.ARPA. 10800 IN PTR pc03.netroad.com.br.
6.20.200.200.IN-ADDR.ARPA. 10800 IN PTR pc04.netroad.com.br.
7.20.200.200.IN-ADDR.ARPA. 10800 IN PTR pc05.netroad.com.br.
8.20.200.200.IN-ADDR.ARPA. 10800 IN PTR pc06.netroad.com.br.
;
; Definição do Roteador e de suas oito portas assíncronas
;
65.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async01.netroad.com.br.
66.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async02.netroad.com.br.
67.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async03.netroad.com.br.
68.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async04.netroad.com.br.
69.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async05.netroad.com.br.
70.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async06.netroad.com.br.
71.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async07.netroad.com.br.
72.20.200.200.IN-ADDR.ARPA. 10800 IN PTR async08.netroad.com.br.
```

4.3.4 Cache: named.root

Todo servidor DNS precisa possuir, para seu funcionamento correto, do nome dos servidores do domínio de mais alto nível, o domínio raiz (“.”), a partir dos quais obterá então informações sobre os servidores dos domínios de mais baixo nível (.com, .edu, .net, .gov, e os domínios regionais como .br, .ca, .jp e outros).

Esta lista pode ser obtida em *ftp://rs.internic.net/domain/named.root* A relação destes servidores pode mudar de tempos em tempos e é conveniente que o administrador de sistemas verifique periodicamente se o arquivo com

esta informação está atualizado.

Caso estes dados estejam incorretos o serviço DNS pode parar de funcionar.

O arquivo *named.root*, válido em junho de 2000, é o seguinte:

named.root

4.3.5 Loopback: 127.0.0.db

Todo servidor DNS necessita de uma entrada adicional para a interface *loopback*, identificada pelo endereço reservado 127.0.0.1. Todo computador possui este endereço reservado para tratar tráfego interno entre processos.

A rede de número 127, como já dissemos, é reservada, e seu primeiro endereço, 127.0.0.1, é utilizado por processos internos que queiram se comunicar. Todos os pacotes de comunicações deste tipo são enviados para a rede de número 127. Um servidor DNS que não configurasse esta interface funciona sem maiores problemas, porém todos os processos que se utilizarem do endereço 127.0.0.1 irão falhar, causando alguns transtornos.

Devido a tudo isto, nunca se esqueça de configurar um mapa para a interface *loopback*, como abaixo:

```
0.0.127.IN-ADDR.ARPA. IN SOA ns.netroad.com.br. dnsmaster.netroad.com.br. (
                  1          ; Serial
                  10800      ; Refresh
                  3600       ; Retry
                  604800     ; Expire
                  86400 )    ; Minimum

0.0.127.IN-ADDR.ARPA.   10800 IN NS  ns.netroad.com.br.
1.0.0.127.IN-ADDR.ARPA. 10800 IN PTR localhost.
```

4.3.6 Servidores Secundários

As linhas

```
secondary  netwizards.com.br        222.222.22.22 s/netwizards.db
secondary  21.200.200.IN-ADDR.ARPA  222.222.22.22 s/200.200.21.0.db
```

indicam que nosso servidor é também um servidor secundário dos domínios *netwizards.com.br* e *21.200.200.IN-ADDR.ARPA.* Para estes domínios não precisamos fazer absolutamente nada, visto que todos os dados serão transferidos do servidor primário destes domínios, identificado pelo número IP 222.222.22.22 e gravados no diretório */usr/local/named/s/netwizards.db* e */usr/local/named/s/200.200.21.0.db.*