Configurando servidores DNS, no muque

http://www.hardware.com.br/tutoriais/servidores-dns/

(Configurando o bind: venha direto para cá).

Introdução

A poucos dias, publiquei um artigo explicando como funciona o sistema DNS e o processo de registro de domínios que serve como uma boa introdução ao tutorial de hoje. Se você já leu o artigo, sinta-se à vontade para pular para a segunda página:

Com frequência, ouvimos dizer que o sistema de DNS é a maior base de dados do mundo. Sob certos aspectos, realmente é, mas existe uma diferença fundamental entre o DNS e um sistema de banco de dados tradicional (como um servidor MySQL usado por um servidor Web, por exemplo), que é o fato do DNS ser uma base de dados distribuída.

No topo da cadeia, temos os root servers, 14 servidores espalhados pelo mundo que têm como função responder a todas as requisições de resolução de domínio. Eles são seguidos por diversas camadas de servidores, que culminam nos servidores diretamente responsáveis por cada domínio.

Um nome de domínio é lido da direita para a esquerda. Temos os domínios primários (chamados de top level domains, ou TLD's), como .com, .net, .info, .cc, .biz, etc., e, em seguida, os domínios secundários (country code TLD's, ou ccTLD's), que recebem o prefixo de cada país, como .com.br ou .net.br. Nesse caso, o "com" é um subdomínio do domínio "br".

Embora normalmente ele seja omitido, todo nome de domínio termina na verdade com um ponto, que representa o domínio raiz, de responsabilidade dos root servers. Quando um dos root servers recebe um pedido de resolução de domínio, ele encaminha a requisição aos servidores da entidade responsável pelo TLD (como ".com") ou pelo ccTLD (como ".com.br") do qual ele faz parte. Eles, por sua vez, encaminham a requisição ao servidor DNS responsável pelo domínio, que finalmente envia a resposta ao cliente.

Ao acessar o endereço "www.gdhn.com.br", o cliente começaria enviando a requisição ao servidor DNS informado na configuração da rede (o DNS do provedor). A menos que tenha a informação em cache, o servidor consulta um dos root servers, perguntando: "quem é o servidor responsável pelo domínio gdhn.com.br?".

O root server gentilmente responde que não sabe, mas verifica qual é o servidor responsável pelos domínios ".br" (o registro.br) e orienta o cliente a refazer a pergunta, dessa vez a um dos servidores da entidade correspondente. O processo pode envolver mais um ou dois servidores, mas eventualmente o cliente chega ao servidor DNS do responsável pelo site (informado ao registrar o domínio) que finalmente fornece o endereço IP do servidor ao cliente:

Assim como no caso do "com", que é um subdomínio do "br" de responsabilidade do Registro.br, você pode criar subdomínios, como "www.gdhn.com.br" ou "ftp.gdhn.com.br" livremente. Estes subdomínios podem apontar para seu próprio servidor, para um servidor separado, ou mesmo serem usados como aliases para outros domínios. Dentro da sua zona, ou seja, do seu domínio, a autoridade é você.

Configurar o servidor DNS é uma etapa importante na configuração de qualquer servidor que vai disponibilizar serviços para a Internet, sobretudo hospedar sites, já que nenhum visitante vai querer acessar os sites hospedados através do endereço IP.

Registro de domínios

Assim como no caso das faixas de endereços IP, que são delegados pelas RIRs (Regional Internet Registries), como a ARIN (http://www.arin.net/) e a LACNIC (http://www.lacnic.net/pt/), os nomes de domínio são delegados através de entidades menores (com ou sem fins lucrativos), chamadas de "domain name registrars" (ou simplesmente "**registrars**"), que coordenam o registro, a delegação e a disputa de domínios. Embora o valor anual de manutenção de cada domínio seja relativamente baixo, o enorme volume de domínios registrados faz com que o registro de domínios seja um negócio que movimenta muito dinheiro.

Os requisitos para registrar domínios variam de acordo com o registrar. Para os TDLs, ou seja, os domínios primários genéricos, como ".com", ".net", ".org" e outros, não existe muita burocracia; basta escolher uma empresa de registro e pagar.

Você pode encontrar uma lista dos registrars oficialmente reconhecidos pela ICANN no:
http://www.icann.org/registrars/accredited-list.html

O maior é o Godaddy (http://godaddy.com), que cobra US$ 9.99 por ano, por domínio .com (com valores diferentes para outros prefixos), seguido pelo Enom (http://www.enom.com/). Existem também algumas empresas nacionais registradas, como a Locaweb (http://locaweb.com.br). Essas empresas concorrem entre si, o que faz com que os preços variem. Os registros de domínio são oferecidos como se fossem um produto, com direito a descontos e promoções:

Você pode ver estatísticas com relação ao volume de domínios TLD registrados, prefixos mais populares e outros detalhes no: http://www.domaintools.com/internet-statistics/

O ranking dos registrars (baseado no volume de domínios registrados por cada um) está disponível no: http://www.domaintools.com/internet-statistics/registrar-stats.html

Além das empresas listadas na página da Internic, que são os registrars primários, existem inúmeras empresas menores que entram como prestadores de serviço, intermediando o registro, como no caso dos provedores de acesso e de empresas como a Brasnic (http://brasnic.com).

Normalmente, elas cobram mais caro, já que precisam registrar o domínio junto a um dos registrars primários, repassando o valor cobrado por ele, e ainda ganharem alguma coisa. O registro de um domínio .com, que custa US$ 9.99 no Godady (e até 6.99 em outros registrars menores) custa US$ 12.00 na Brasnic, por exemplo.

O registro de domínios .br é menos caótico, pois eles são controlados por uma única entidade, o Registro.br (http://registro.br), uma entidade sem fins lucrativos. A taxa de registro é (enquanto escrevo) de R$ 30 anuais por domínio registrado, mas existem algumas exigências adicionais.

Para registrar um domínio ".com.br", por exemplo era, até pouco tempo, necessário ter uma empresa aberta em território nacional, para registrar um domínio ".net.br" é necessário ter uma empresa dentro do ramo de telecomunicações e assim por diante. Pessoas físicas (residentes no Brasil, ou que possuam um contato no Brasil) podem registrar apenas domínios específicos, como o "nom.br", "blog.br", "flog.br" e outros. Em primeiro de maio de 2008 entrou em vigor uma nova norma, que flexibilizou o registro dos domínios ".com.br", liberando o registro para pessoas físicas, desde que com um CPF válido.

Existem ainda outros detalhes interessantes, como o fato de empresas estrangeiras poderem fazer o registro apenas através de um procurador. Você pode ver mais detalhes no FAQ: http://registro.br/faq/index.html

Note que o registro de domínios inclui apenas o cadastramento do domínio e o encaminhamento das requisições aos seus servidores DNS, informados durante o registro. Em muitos casos, são oferecidos serviços adicionais, como a exibição de uma página "em construção" (placeholder), a configuração dos servidores DNS para você, ou

mesmo a hospedagem do site. Entretanto, estes são serviços adicionais, que variam de acordo com a empresa de registro escolhida.

Uma prática muito comum é registrar domínios em que você tenha interesse, mas que não pretenda usar de imediato, mostrando uma página genérica, contendo um "em construção" ou alguns links de anúncios. Esta prática é chamada de "**domain parking**" (reserva de domínios, ou estacionamento de domínios) e é bastante difundida, já que sai mais barato registrar um domínio antecipadamente do que ter que disputá-lo mais tarde. Existem também casos de empresas que deliberadamente registram um grande volume de domínios contendo palavras ou frases populares, com o objetivo de vendê-los mais tarde, ou simplesmente lucrar com cliques de visitantes que acessam os endereços por acidente.

Existem também casos de registros de domínios contendo marcas, ou palavras similares a marcas, com objetivo de enganar os visitantes (encaminhando-os a outras páginas) e/ou lesar ou extorquir os proprietários da marca. Esta prática é chamada de "**cybersquatting**" (grilagem de domínios) e é ilegal na maioria dos países, incluindo o Brasil.

Embora seja um processo demorado, é possível disputar a posse de um domínio registrado, o que se aplica em casos em que você é o detentor de uma marca registrada, ou é o proprietário de um site que esteja sendo lesado por um domínio similar, registrado com o propósito de roubar visitantes.

Para os domínios primários, o processo é chamado de **UDRP** (Uniform Domain-Name Dispute-Resolution Policy), cujos detalhes estão disponíveis no: http://www.icann.org/udrp/udrp.htm

Para os domínios ccTLDs, ou seja, os domínios com código de país, que são responsabilidade de entidades separadas, o processo varia. Algumas entidades aceitam a aplicação do UDRP, outras aplicam conjuntos particulares de regras, enquanto outras simplesmente não possuem uma política definida, se limitando a acatar decisões judiciais.

Atualmente (junho de 2008) o Registro.br ainda faz parte da terceira categoria, mas existem negociações com relação à adoção do UDRP. Você pode ver algumas cartas nesse sentido, trocadas entre os responsáveis pelo Registro.br e a ICANN, disponíveis no:
http://www.icann.org/cctlds/br/br-icann-letters-10may07.pdf

DNS e virtual hosting

Com poucas exceções, ao registrar um domínio você precisa fornecer o endereço de dois servidores DNS (primário e secundário), para onde serão encaminhadas as consultas referentes ao seu domínio. O segundo servidor é exigido para fins de redundância, garantindo que as requisições continuem a ser respondidas mesmo que o primeiro servidor esteja fora do ar.

Uma opção muito usada para o segundo DNS é pedir para que algum amigo, que também possua um servidor dedicado, seja seu DNS secundário. Ele precisará apenas adicionar a configuração do seu domínio na configuração do DNS, o que é rápido e indolor. Se você administra dois servidores diferentes, pode também configurar o

servidor B para ser o DNS secundário dos domínios hospedados no servidor A e vice-versa.

Ao locar um servidor dedicado, é comum que você receba dois ou mais endereços IP's válidos. Originalmente, seu servidor vai estar configurado para usar apenas um deles, mas você pode ativar o segundo (mesmo que o servidor possua apenas uma placa de rede) usando o ifconfig, como veremos em detalhes a seguir. Isso permite que o mesmo servidor seja usado simultaneamente como DNS primário e secundário, eliminando a necessidade de um segundo servidor.

Continuando, ao registrar um domínio, você passa a ter autoridade sobre ele e pode criar subdomínios da forma como quiser, como "fulano.meunome.com.br" ou "vendas.minhaempresa.com". Veja o caso dos serviços de hospedagem gratuita de blogs e sites, como o blogger, o wordpress e tantos outros, que, em muitos casos, criam milhões de subdomínios diferentes para as páginas hospedadas.

Resolver um nome de domínio (ou seja, percorrer todo o caminho necessário para descobrir o IP do servidor responsável, começando com a requisição enviada aos root servers) é uma operação que pode demorar vários segundos, por isso os servidores DNS armazenam um cache de domínios já resolvidos, minimizando o número de requisições. É por isso que quando você faz alguma mudança na configuração do domínio, demora algumas horas para que ela se replique. Isso explica também casos onde você não consegue acessar um determinado site usando o DNS do provedor (que está desatualizado), mas consegue usando um DNS local, ou outro servidor qualquer.

Um único servidor pode ser configurado para responder por inúmeros domínios, assim como um único servidor web pode hospedar vários sites. As duas configurações acabam intimamente ligadas, já que é justamente a presença de vários domínios, ou vários subdomínios que permite ao servidor web entregar a página apropriada ao cliente.

Imagine o caso de um servidor que hospeda 10.000 sites. Na configuração do Apache, especificamos o domínio e o diretório local correspondente a cada site, como em:

```
<VirtualHost *>
ServerAdmin webmaster@gdhn.com.br
ServerName www.gdhn.com.br
ServerAlias gdhn.com.br
DocumentRoot /var/www/gdhn
</VirtualHost>
```

A idéia aqui é que o visitante digita o nome de domínio do site no navegador e, ao receber a requisição, o Apache se encarrega de enviá-lo ao diretório correto. Isso é possível porque o domínio a ser acessado é uma informação que faz parte da requisição enviada pelo cliente.

Depois de resolver o domínio e obter o endereço do servidor do site, o cliente não pedirá "me envie a página index do site", mas sim "me envie a página index do site www.gdhn.com.br". O fato do domínio ser incluído na requisição permite que o Apache verifique a configuração e forneça os arquivos do diretório correto.

Se seu servidor estiver hospedando subdomínios, ou seja, endereços como "www.fulano.gdhn.com.br", "www.ciclano.gdhn.com.br", etc., como fazem serviços de hospedagem, a configuração continua basicamente a mesma: você especifica o subdomínio do cliente na configuração do VirtualHost do Apache e também na configuração do servidor DNS.

Uma observação importante é que, para o Apache, o domínio "www.fulano.gdhn.com.br" é diferente de apenas "fulano.gdhn.com.br". A linha "ServerAlias" na configuração serve justamente para permitir que o site seja acessado tanto com o www quanto sem, de acordo com o gosto do freguês.

Muitos sites grandes acabam ficando acessíveis apenas com (ou apenas sem) o "www", simplesmente por que os administradores deixam de criar os aliases necessários na configuração do DNS, ou deixam de configurar o servidor web para responder pelos dois endereços, como no caso do site da claro. Ao acessar o "www.claro.com.br" acesso o site normal da operadora, mas ao tentar o "http://claro.com" obtenho uma mensagem de erro "Invalid Hostname", gerada pelo servidor IIS que hospeda o site. Este é um exemplo de erro que você vai querer evitar na configuração dos seus servidores:

Voltando à explicação inicial, como os servidores de registro de domínio lêem as URLs de trás para a frente, todos os acessos a subdomínios dentro do "gdhn.com.br" serão enviados para o servidor DNS responsável pelo domínio e daí para o servidor Apache. Chegamos então ao prato principal, que é a configuração do servidor DNS propriamente dito.

Configurando o Bind

O servidor DNS mais usado no Linux é o Bind, que aprenderemos a configurar aqui. Não existe problema em instalá-lo no mesmo servidor onde foi instalado o Apache e os demais serviços, embora, do ponto de vista da segurança, o ideal seja utilizar servidores separados (ou usar uma máquina virtual) ou usar um chroot, onde o Bind roda dentro de um diretório separado, sem acesso aos demais arquivos do sistema. Veremos como configurar o Bind para operar dentro do chroot mais adiante, por enquanto vamos nos concentrar na configuração propriamente dita.

O Bind é um software open-source, desenvolvido pelo ISC (Internet Software Consortium) e é largamente utilizado não apenas no Linux, mas em sistemas Unix em geral, incluindo o FreeBSD e o Mac OS X, ou seja, praticamente todos os sistemas operacionais com exceção do Windows.

Instalação

Para instalar o Bind, procure pelo pacote "bind" ou "bind9" no gerenciador de pacotes da distribuição usada. Nas distribuições derivadas do Debian, o pacote "bind" instala o Bind 8, enquanto o pacote "bind9" instala o Bind 9, que é a versão recomendada:

```
# apt-get install bind9
```

No CentOS, no Fedora e no Mandriva o pacote "bind" instala diretamente a versão mais recente, sem opção de escolher entre instalar o Bind 8 ou 9:

```
# yum install bind
```

ou:

```
# urpmi bind
```

No Slackware você encontra o pacote dentro da pasta "n" do primeiro CD. Ao instalar, verifique a versão incluída na distribuição. Use sempre o Bind 8 ou 9; nunca o Bind 4, que está em desuso.

O Bind passou diretamente da versão 4 para a versão 8, de forma a acompanhar a numeração de versões usada no Sendmail. Com isso, as versões 5, 6 e 7 não existiram e o Bind 8 equivale à quinta versão do software.

Se você está instalando o Bind em um servidor local, configure as estações para utilizarem o endereço IP do servidor como DNS primário e você verá que eles já serão capazes de navegar normalmente, sem precisar mais do DNS do provedor. Esta é a configuração padrão do Bind, onde ele trabalha como um servidor DNS de cache, encaminhando as requisições para os root servers, sem responder nada diretamente. A configuração que veremos aqui é necessária para que ele passe a responder pelos seus domínios, assumindo a função de servidor autoritativo.

O principal arquivo de configuração do Bind é o "/etc/bind/named.conf" (em versões antigas, o arquivo pode ser simplesmente "/etc/named.conf"). Como comentei, por padrão o Bind já vem configurado para trabalhar como um servidor DNS de cache, que pode ser usado tanto localmente quanto por outros PCs da rede local. Dentro do arquivo de configuração, você encontrará entradas similares a essas:

```
zone "." {
type hint;
file "/etc/bind/db.root";
};

zone "localhost" {
type master;
file "/etc/bind/db.local";
};

zone "127.in-addr.arpa" {
type master;
file "/etc/bind/db.127";
};

zone "0.in-addr.arpa" {
type master;
file "/etc/bind/db.0";
};

zone "255.in-addr.arpa" {
type master;
```

```
file "/etc/bind/db.255";
};
```

Como pode ver, cada uma das seções indica a localização de um arquivo, onde vai a configuração referente a ela. Por exemplo, na primeira seção ("`zone "."`") é indicado o arquivo "`/etc/bind/db.root`", que contém os endereços dos 14 root servers, que o Bind contactará na hora de resolver os domínios.

Esta configuração vem incluída por padrão e não deve ser alterada, a menos que você saiba bem o que está fazendo. O que fazemos ao configurar o servidor DNS é incluir novas zonas (ou seja, novas seções de configuração), contendo os domínios que desejamos configurar.

Uma exceção fica por conta do CentOS 5 onde (acompanhando a mudança feita no RHEL 5) os arquivos de configuração do Bind não são instalados junto com o pacote. Nele, é necessário que você gere a configuração inicial copiando os modelos de configuração que estão dentro da pasta "`/usr/share/doc/bind-9.?.?/sample/`" para as pastas apropriadas, como em:

```
# cp -r /usr/share/doc/bind-9.?.?/sample/etc/* /etc/
# cp -r /usr/share/doc/bind-9.?.?/sample/var/named/* /var/named/
```

O serviço referente ao Bind pode se chamar "bind" ou "named", de acordo com a distribuição. Nos derivados do Debian você controla o serviço através do comando "/etc/init.d/bind9" (ou "/etc/init.d/bind" para a versão 8), enquanto nas distribuições derivadas do Red Hat utilizamos o comando "service named", como em:

```
# /etc/init.d/bind9 restart
```

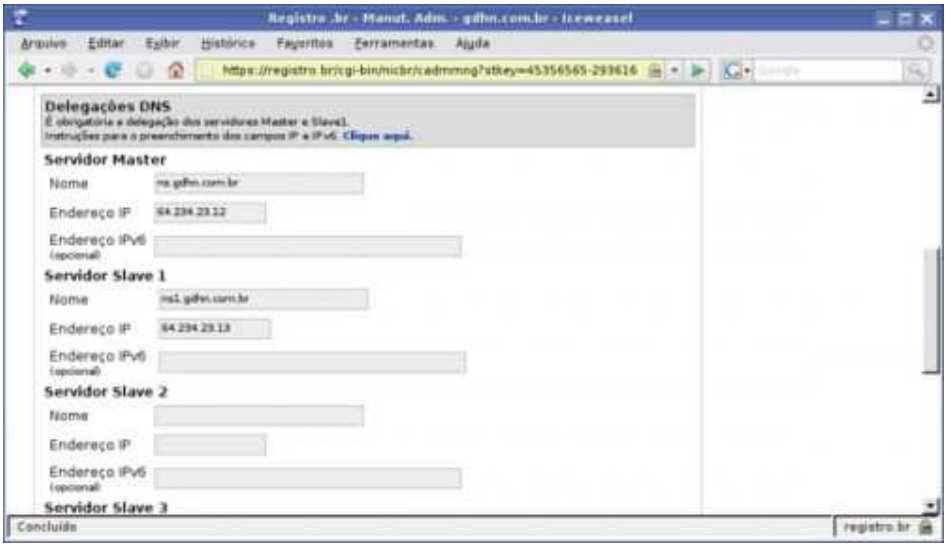
ou:

```
# service named restartc
```

Adicionando os domínios

Com o Bind instalado, o próximo passo é configurar o serviço para responder pelos domínios que você registrou. Vamos usar como exemplo o domínio "**gdhn.com.br**".

Como ele é um domínio .br, ele é registrado através do http://registro.br. Depois de pagar e fornecer os dados da empresa e do responsável pelo domínio, é necessário fornecer os endereços dos dois endereços (nomes e endereços IP) dos dois servidores DNS responsáveis pelo seu domínio, como no screenshot abaixo:



É através dessa interface de gerenciamento que você pode alterar a configuração do seu domínio, alterar os endereços dos servidores DNS (caso você migre seu site para outro servidor, ou outra empresa de hospedagem, por exemplo), ou mesmo transferí-lo para outra pessoa. A senha definida ao criar sua conta é essencial, já que qualquer um que tenha acesso às suas informações de login poderia se apoderar do seu domínio (é possível recuperá-lo depois caso você seja o responsável legal pela empresa, mas o processo é demorado).

É possível registrar o domínio usando um plano ADSL empresarial, ou outra modalidade de conexão onde você tenha um endereço IP fixo. Não seria recomendável hospedar o site da sua empresa (ou nem mesmo seu site pessoal) em um servidor ligado a um link ADSL, pois o acesso dos visitantes seria muito ruim, mas, de qualquer forma, na falta de um servidor dedicado, você pode montar um servidor de internet doméstico para fins de estudo ou de testes. A dica para conseguir registrar o domínio tendo um único endereço IP é utilizar um modem discado (ou qualquer outro tipo de conexão temporária) para obter um segundo endereço e assim conseguir efetuar o processo de registro do domínio.

Não faria muito sentido tentar registrar um domínio usando uma conexão doméstica, com IP dinâmico, já que a entrada ficaria desatualizada assim que o IP mudasse, fazendo com que o domínio deixasse de funcionar. A única solução viável nesse caso é utilizar um serviço de DNS dinâmico (no-ip.com, dyndns.com, etc.).

Continuando, a checagem dos servidores DNS é feita durante a fase final do registro, de forma que, para concluir o registro do domínio, seu DNS já deve estar funcionando. Vamos então à configuração do Bind.

Comece adicionando as seguintes linhas no final do arquivo "**/etc/bind/named.conf**" do servidor (sem modificar as demais):

```
zone "gdhn.com.br" IN {
type master;
file "/etc/bind/db.gdhn";
allow-transfer { 64.234.23.12; };
};
```

Na configuração, o "**zone "gdhn.com.br**" na primeira linha indica o domínio que estamos configurando, como registrado no Registro.br.

O "file "**/etc/bind/db.gdhn**" especifica o arquivo onde vai a configuração desse domínio. Na verdade, você pode salvar esse arquivo em qualquer lugar, muita gente usa a pasta "/var/named". Aqui estou seguindo o padrão do Debian, colocando os arquivos dentro da pasta "/etc/bind", junto com os demais arquivos de configuração do Bind.

Estas linhas dizem que o servidor é o responsável pelo domínio "gdhn.com.br" (type master;) e que sempre que receber uma requisição vai responder de acordo com o especificado no arquivo db.gdhn (file "/etc/bind/db.gdhn";).

A linha "allow-transfer" indica quais servidores terão permissão para realizar transferências de zona. Ao utilizar um servidor DNS secundário, a linha conteria o endereço IP do segundo servidor, que seria o único autorizado a realizar as transferências. Se você estiver utilizando um único servidor, utilize o próprio endereço IP do servidor (como no exemplo). Não deixe de utilizar a opção, caso contrário qualquer servidor poderá realizar transferências, o que permite obter diversas informações sobre seu domínio, que poderão então serem utilizadas para formular ataques.

No caso do **Debian**, é recomendado que você use o arquivo "**/etc/bind/named.conf.local**", que é processado como se fosse parte do named.conf principal. No arquivo named.conf do Debian, você encontra as seguintes linhas:

```
// If you are just adding zones, please do that in
/etc/bind/named.conf.local
include "/etc/bind/named.conf.local";
```

A existência desse arquivo separado visa separar a configuração geral do servidor e a configuração dos domínios, minimizando a possibilidade de erros, mas, na verdade, o efeito de editar qualquer um dos dois arquivos é o mesmo.

Em seguida vem a parte principal, que é adicionar a configuração do domínio no arquivo "**/etc/bind/db.gdhn**", que foi citado na configuração. Este é um exemplo de configuração:

```
@ IN SOA servidor.gdhn.com.br. hostmaster.gdhn.com.br. (
2008061645 3H 15M 1W 1D )
NS servidor.gdhn.com.br.
IN MX 10 servidor.gdhn.com.br.
gdhn.com.br. A 64.234.23.12
www A 64.234.23.12
ftp A 64.234.23.12
smtp A 64.234.23.12
```

Nesse arquivo a formatação é especialmente importante. Você pode usar espaços e tabs (ambos têm o mesmo efeito) para organizar as opções, mas existem algumas regras. As linhas "IN SOA" até "IN MX" precisam ficar justificadas (como no exemplo) e você não pode esquecer dos espaços entre as opções. Caso queira incluir comentários, use ";" ao invés de "#", como em outros arquivos.

Pesquisando no Google, você pode encontrar inúmeros templates como esse, mas é difícil encontrar alguma explicação clara de cada uma das opções. Isso faz com que configurar um servidor DNS pareça muito mais complicado do que realmente é. Vamos então a uma descrição detalhada de cada um dos campos, começando pela primeira linha:

```
@ IN SOA servidor.gdhn.com.br. hostmaster.gdhn.com.br. (
```

A "@" na primeira linha indica a origem do domínio e, ao mesmo tempo, o início da configuração. Ela é sempre usada, assim como em um endereço de e-mail.

O "IN" é abreviação de "internet" e o "SOA" de "Start of autority". Em seguida vem o nome do seu servidor (que você checa usando o comando "hostname"), seguido do e-mail de contato do administrador (você).

Note que, no caso do e-mail, temos a conta separada do domínio por um ponto, e não por uma "@". O mais comum é criar uma conta chamada "hostmaster", mas isso não é uma regra. Você poderia usar "fulaninho.meudominio.com.br", por exemplo. A principal observação é que você não deve usar um e-mail contendo pontos na porção antes da arroba.

Note também que existe um ponto depois do "servidor.gdhn.com.br" e do "hostmaster.gdhn.com.br", que faz parte da configuração. Como citei no início, na realidade todos os nomes de domínio terminam com um ponto; em muitas situações o ponto é omitido (como acessar um site através do navegador), mas ele é obrigatório dentro da configuração do Bind.

O ponto se refere ao domínio raiz, de responsabilidade dos rootservers. No exemplo, nosso servidor é o responsável pelo domínio "gdhn", que faz parte do domínio ".com.br", que, por sua vez, faz parte do domínio raiz. Lembre-se que os domínios são lidos da direita para a esquerda, de forma que, ao resolver o domínio, o cliente leria: raiz . br . com . gdhn.

A linha diz algo como: na internet, o servidor "servidor" responde pelo domínio "gdhn.com.br" e o e-mail do responsável pelo domínio é "hostmaster@gdhn.com.br".

A primeira linha termina com um parênteses, que indica o início da configuração do domínio. Temos então:

```
2008061645 3H 15M 1W 1D )
```

O "2008061645" é o valor de sincronismo, que permite que o servidor DNS secundário se mantenha sincronizado com o principal, detectando alterações na configuração. O servidor DNS checa periodicamente as informações disponibilizadas pelo servidor primário e realiza uma atualização sempre que percebe que o número de sincronismo do servidor é mais alto que o seu (do servidor secundário).

Não existe uma regra específica para a formatação do número de sincronismo; você pode simplesmente usar um número de 10 dígitos aleatório e aumentá-lo a cada mudança na configuração.

Uma convenção popular é usar a data da última alteração (como em: 20080616) e um número de dois dígitos qualquer, aumentando sequencialmente a cada mudança. Ela acaba sendo útil, pois faz com que você nunca se esqueça de atualizar o número de sincronismo ao alterar a configuração do servidor. :)

Sempre que editar a configuração ou sempre que configurar um servidor DNS a partir de um template qualquer, lembre-se de atualizar o número de sincronismo, usando a data atual ou outro número definido por você. Uma observação é que o número no servidor primário deve ser sempre superior ao número no servidor secundário, caso contrário a atualização nunca é disparada. Veja que a convenção de usar a data evita esse problema, já que ela faz com que o servidor que foi atualizado por último tenha sempre o número mais alto.

Continuando, os quatro campos seguintes (3H 15M 1W 1D) orientam o servidor DNS secundário (caso você tenha um). O primeiro campo indica o tempo que o servidor aguarda entre as atualizações (3 horas). Caso ele perceba que o servidor principal está fora do ar, ele tenta fazer uma transferência de zona, ou seja, tenta assumir a

responsabilidade sob o domínio. Caso a transferência falhe e o servidor principal continue fora do ar, ele aguarda o tempo especificado no segundo campo (15 minutos) e tenta novamente.

O terceiro campo indica o tempo máximo que ele pode responder pelo domínio, antes que as informações expirem (1 semana, tempo mais do que suficiente para você arrumar o servidor principal ;) e o tempo mínimo antes de devolver o domínio para o servidor principal quando ele retornar (1 dia). Se você acha que uma semana é pouco tempo, você pode aumentar o valor, usando, por exemplo, "4W" (4 semanas). Com isso você tem mais tempo para restaurar o servidor primário em caso de catástrofe.

Esses são os valores padrão, por isso não existem muitos motivos para alterá-los. A transferência do domínio para o DNS secundário é sempre uma operação demorada, por causa do cache feito pelos diversos servidores DNS espalhados pelo mundo: demora de um a dois dias até que todos atualizem suas tabelas de endereços. A principal prioridade deve ser evitar que o servidor principal fique indisponível em primeiro lugar.

Muitos preferem especificar esses valores em segundos. Uma configuração muito comum é separar os valores por linha, incluindo comentários, como em:

```
2008061645; serial
10800; refresh, seconds
900; retry, seconds
604800; expire, seconds
86400 ); minimum, seconds
```

O resultado é exatamente o mesmo. A única diferença é que você vai acabar digitando várias linhas a mais.

As duas linhas a seguir concluem a seção inicial:

```
NS servidor.gdhn.com.br.
IN MX 10 servidor.gdhn.com.br.
```

A linha "NS" (Name Server) diz quem são os servidores DNS responsáveis pelo domínio. Ao usar apenas um servidor DNS, você simplesmente repete o nome do servidor, seguido pelo domínio, como adicionamos na primeira linha. Caso você esteja usando dois servidores, então você precisa declarar ambos, como em:

```
NS servidor.gdhn.com.br.
NS ns2.gdhn.com.br.
```

A linha "IN MX" (Mail Exchangers) é necessária sempre que você pretende usar um servidor de e-mails (você pode escolher entre usar o Postfix, Qmail, Sendmail ou outro MTA). Aqui estou simplesmente usando a mesma máquina para tudo, por isso novamente citei o "servidor.gdhn.com.br", que acumula mais esta função. Assim como no caso do DNS, você pode especificar um servidor de e-mails secundário, que passa a receber os e-mails caso seu servidor principal saia fora do ar. Nesse caso, você adiciona uma segunda linha, como em:

```
IN MX 10 servidor.gdhn.com.br.
IN MX 20 outroservidor.outrodominio.com.br.
```

Os números indicam a prioridade de cada servidor. O servidor da primeira linha tem prioridade 10, por isso é o primário. O segundo tem prioridade 20 e por isso só assume em casos de problemas com o primário. Usar um segundo servidor de e-mails, em um domínio separado, adiciona uma camada extra de redundância e evita que você perca e-mails caso seu servidor fique temporariamente fora do ar.

Em grandes redes, é possível adicionar um volume muito maior de servidores, de forma a garantir a operação do serviço em qualquer situação. É possível até mesmo terceirizar o serviço, adicionando os servidores de uma empresa externa. Ao usar o Google apps for your domain (https://www.google.com/a/), que permite criar contas do Gmail com o domínio do seu site, por exemplo, você é orientado a adicionar as seguintes linhas na configuração:

```
IN MX 1 ASPMX.L.GOOGLE.COM.
IN MX 5 ALT1.ASPMX.L.GOOGLE.COM.
IN MX 5 ALT2.ASPMX.L.GOOGLE.COM.
IN MX 10 ASPMX2.GOOGLEMAIL.COM.
IN MX 10 ASPMX3.GOOGLEMAIL.COM.
IN MX 10 ASPMX4.GOOGLEMAIL.COM.
IN MX 10 ASPMX5.GOOGLEMAIL.COM.
```

Com isso, os e-mails são recebidos pelos servidores do Google (veja que são usados nada menos do que 7 servidores diferentes), que os armazenam nas contas apropriadas. O servidor DNS do seu site passa a apenas encaminhar as requisições.

Uma observação é que o protocolo SMTP prevê falhas nos links entre os servidores, de forma que, caso o servidor de e-mail principal esteja inativo, o emissor tenta contactá-lo durante um longo período (o default são 3 dias) antes de tentar o servidor secundário. Com isso, usar vários servidores não resolve o problema completamente, já que em caso de falha no servidor primário, as mensagens passarão a chegar com um grande atraso.

Outra observação é que os endereços dos domínios da configuração fornecida pelo Google aparecem em letras maiúsculas, mas, apesar disso, a configuração funciona sem problemas. Isso acontece porque, diferente de outras configurações, os nomes de domínio são case-insensitive, de forma que tanto faz escrevê-los usando letras minúsculas ou maiúsculas.

Depois dessas linhas iniciais, temos a parte mais importante, em que você especifica o endereço IP do servidor e pode cadastrar subdomínios, como em:

```
gdhn.com.br. A 64.234.23.12
www A 64.234.23.12
ftp A 64.234.23.12
smtp A 64.234.23.12
```

Nesse exemplo, incluí também três subdomínios, o "www", "ftp" e "smtp", ambos relacionados ao IP do servidor. Isso permite que os visitantes digitem "www.gdhn.com.br" ou "ftp.gdhn.com.br" no navegador. Ao trabalhar com subdomínios, você pode relacioná-los com IP's ou domínios diferentes. Por exemplo, muitos portais possuem vários subdomínios, como "www1", "www2" e "www3", onde cada um é um servidor diferente, configurados para dividir os acessos.

Ao trabalhar com dois servidores DNS, adicione também uma entrada para o servidor secundário, especificando o nome do segundo servidor (ns2 no exemplo) e o endereço IP, como em:

```
gdhn.com.br. A 64.234.23.12
www A 64.234.23.12
ftp A 64.234.23.12
smtp A 64.234.23.12
ns2 A 64.234.23.13
```

Note o segundo DNS usa o IP .13, enquanto o servidor principal usa o .12, mas ambos estão dentro da mesma faixa de endereços.

DNS primário e secundário no mesmo servidor

Como comentei anteriormente, é possível fazer com que um único servidor dedicado (que disponha de dois ou mais endereços IP) atue simultaneamente como servidor DNS primário e secundário, evitando que você precise de um segundo servidor separado. Para isso, precisamos apenas criar um alias para a placa de rede. Se o segundo IP é o "64.234.23.13" e a placa de rede é a "eth0", o comando seria:

```
# ifconfig eth0:1 64.234.23.13
```

A partir daí, seu servidor passa a responder pelos dois endereços IP, e você pode usá-lo simultaneamente como DNS primário e secundário.

Naturalmente, ao fazer isso, você perde a redundância (que é o grande motivo de usar dois servidores DNS em primeiro lugar) mas isso nem sempre é um grande problema, já que se o servidor DNS está hospedado no mesmo servidor que seu site, não faz muita diferença ter dois servidores DNS, pois se o servidor principal cair, o site ficará fora do ar de qualquer forma.

Sites maiores possuem sistemas de redundância e, muitas vezes, servidores DNS separados, o que cria uma malha de segurança. É por isso que é muito raro a página de um portal ficar fora do ar, por exemplo.

Continuando, o comando do ifconfig não é permanente, de forma que você deve adicioná-lo a um dos arquivos de inicialização do sistema (como o "/etc/rc.local") para que a configuração torne-se permanente. Nas distribuições derivadas do Debian você pode adicionar o alias diretamente ao arquivo "/etc/network/interfaces", logo depois da configuração da interface principal, como em:

```
auto eth0:1
iface eth0:0 inet static
address 64.234.23.13
netmask 255.255.255.248
```

É necessário também especificar o segundo endereço na configuração da zona no Bind, adicionando uma linha "NS" adicional. Como no caso o mesmo servidor responde pelos dois endereços IP, você pode simplesmente inventar um nome fictício para o segundo endereço ao incluí-lo na configuração, como nesse exemplo, onde uso o nome "secundario":

```
@ IN SOA servidor.gdhn.com.br. hostmaster.gdhn.com.br. (
2008061645 3H 15M 1W 1D )
NS servidor.gdhn.com.br.
NS secundario.gdhn.com.br.
IN MX 10 servidor.gdhn.com.br.
gdhn.com.br. A 64.234.23.12
secundario A 64.234.23.13
```

O "A" é abreviação de "Address mapping" e é usado em entradas onde um domínio ou um subdomínio é relacionado a um endereço. Você pode também usar a diretiva "IN CNAME" para criar aliases, ou seja, subdomínios que atuam como apelidos para outros. Veja um exemplo:

```
gdhn.com.br. A 64.234.23.12
www A 64.234.23.12
ftp A 64.234.23.12
smtp A 64.234.23.12
ns2 A 64.234.23.13
joao A 200.123.23.2
maria IN CNAME joao
```

Nesse caso, o subdomínio "maria" é simplesmente um alias para o "joao", que aponta para um IP externo. Você pode criar quantos subdomínios de aliases precisar.

Para especificar endereços IPV6, você usa a diretiva "AAAA" no lugar de "A", como em:

```
isac AAAA 2001:bce4:5641:3412:341:45ae:fe32:65
```

Concluindo, o nome do DNS secundário especificado na configuração (o "secundario" no meu exemplo) deve ser incluído no arquivo "**/etc/hosts**", de forma que o endereço IP fornecido no alias da placa de rede seja relacionado ao segundo nome, como em:

```
# /etc/hosts
127.0.0.1 localhost.localdomain localhost
64.234.23.12 servidor.gdhn.com.br servidor
64.234.23.13 secundario.gdhn.com.br secundario
```

Ao terminar, você pode testar a configuração do seu servidor DNS usando o comando **dig**. No Debian ele é instalado juntamente com o pacote "dnsutils". Faça uma busca pelo domínio, especificando o endereço IP do DNS que acabou de configurar, como em:

```
$ dig gdhn.com.br @64.234.23.12
```

Isso faz com que ele pergunte diretamente ao seu servidor, o que permite testar a configuração imediatamente, sem precisar esperar pela propagação do registro do domínio. Se tudo estiver correto, você verá algo como:

```
;; ANSWER SECTION:
gdhn.com.br. 86400 IN A 64.234.23.12

;; AUTHORITY SECTION:

gdhn.com.br. 86400 IN NS servidor.gdhn.com.br.
gdhn.com.br. 86400 IN NS secundario.gdhn.com.br.
```

Faça o mesmo com o IP do DNS secundário, como em:

```
$ dig gdhn.com.br @64.234.23.13
```

Ambos devem devolver a mesma resposta.

Para adicionar mais domínios, edite o arquivo "/etc/named.conf" (ou o "/etc/named.conf.local"), adicionando seções separadas (e especificando arquivos de configuração separados) para cada um dos domínios, como em:

```
zone "joao.com.br" IN {
type master;
file "/etc/bind/db.joao";
allow-transfer { 64.234.23.13; };
};
```

```
zone "maria.com.br" IN {
type master;
file "/etc/bind/db.maria";
allow-transfer { 64.234.23.13; };
};
```

Depois que você configurou o primeiro domínio, fica fácil adicionar domínios adicionais, pois você pode simplesmente gerar cópias do arquivo original, alterando apenas as opções que forem diferentes, tais como o nome do domínio e os subdomínios desejados. Aqui temos mais um exemplo de configuração:

```
@ IN SOA servidor.joao.com.br. contato.joao.com.br. (
2008051567 3H 15M 1W 1D )
NS servidor.joao.com.br.
NS ns2.joao.com.br.
IN MX 10 servidor.joao.com.br.
joao.com.br. A 64.234.23.12
www A 64.234.22.12
forum A 64.234.22.12
smtp A 64.234.22.12
ns2 A 64.234.23.13
downloads A 72.213.45.23
```

Veja que o subdomínio "downloads" aponta para um servidor diferente que, naturalmente, deve ter um servidor web configurado para responder pelo domínio.

Usando um servidor secundário separado

Vamos agora a uma configuração mais elaborada, usando dois servidores, configurados como master e slave. Como comentei, essa configuração se aplica apenas quando você realmente utiliza dois servidores separados e não em casos em que você tem um único servidor configurado para utilizar dois endereços IP. No exemplo, temos dois servidores, o primário utilizando o endereço "64.234.23.12" e o secundário o endereço "64.234.23.13".

O primeiro passo é a configuração no arquivo "/etc/bind/named.conf", ou "/etc/bind/named.conf.local", que é diferente nos dois servidores.

No servidor **primário**, a entrada deve conter a linha "allow-transfer", especificando o endereço do servidor secundário. Ela indica que o servidor secundário pode assumir a responsabilidade sobre o domínio em caso de problemas com o titular:

```
zone "gdhn.com.br" IN {
type master;
file "/etc/bind/db.gdhn";
allow-transfer { 64.234.23.13; };
};
```

Na configuração do servidor DNS **secundário**, mudam duas linhas, onde você especifica a posição hierárquica do servidor ("type slave" ao invés de "type master") e o endereço IP do servidor configurado como master:

```
zone "gdhn.com.br" IN {
type slave;
file "/etc/bind/db.gdhn";
```

```
masters { 64.234.23.12; };
};
```

Ao usar dois servidores separados como master e slave, é importante que a porta 53 TCP esteja aberta no firewall tanto para entrada quanto para saída. Embora o protocolo DNS utilize a porta 53 UDP para a resolução de nomes, a porta 53 TCP é usada durante as transferências de zona (Zone Transfers), ou seja, durante o processo de sincronismo entre os dois servidores. Sem acesso à porta 53 TCP do servidor principal, o servidor DNS secundário simplesmente não tem como desempenhar suas funções.

Continuando, temos a configuração do arquivo "/etc/bind/db.gdhn", que deve conter duas entradas "NS", uma para o servidor primário e outra para o servidor secundário, além de uma entrada "A" relacionando o nome do servidor secundário com o endereço utilizado por ele, como em:

```
@ IN SOA servidor.gdhn.com.br. hostmaster.gdhn.com.br. (
2008061645 3H 15M 1W 1D )
NS servidor.gdhn.com.br.
NS ns1.gdhn.com.br.
IN MX 10 servidor.gdhn.com.br.
gdhn.com.br. A 64.234.23.12
www A 64.234.23.12
ftp A 64.234.23.12
smtp A 64.234.23.12
ns2 A 64.234.23.13
```

Uma curiosidade é que a configuração precisa ser feita apenas no servidor primário, já que o servidor secundário se sincroniza periodicamente em relação a ele, realizando transferências de zona. O mesmo se aplica às atualizações no arquivo que, novamente, precisam ser feitas apenas no servidor primário. Ao terminar, não esqueça de reiniciar o serviço para que as alterações entrem em vigor.

Na segunda parte deste tutorial veremos detalhes sobre a configuração do DNS reverso, configuração de servidores DNS para redes locais e também sobre o uso do Bind dentro de um chroot para melhorar a segurança.

Confira a segunda parte em: http://www.hardware.com.br/tutoriais/servidores-dns_2/