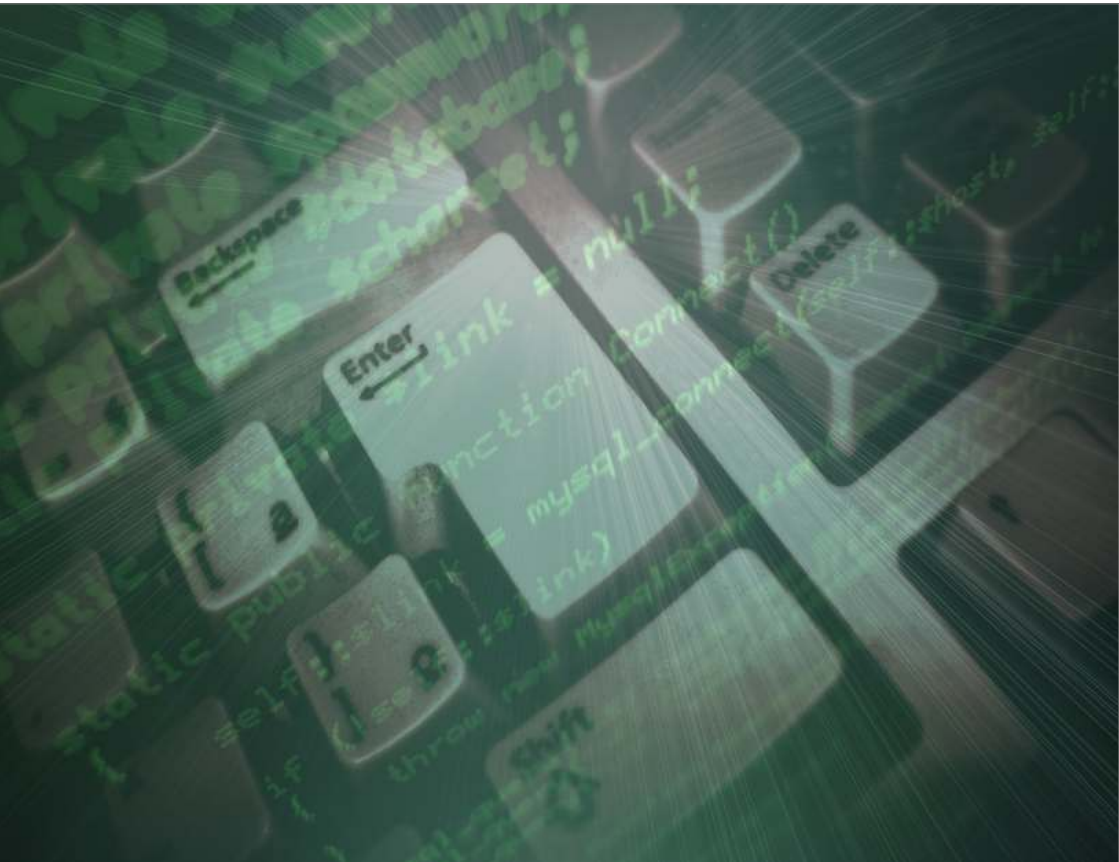


ISSN 1677-9274

Instalação e Configuração do PostgreSQL no Linux via código-fonte





*Empresa Brasileira de Pesquisa Agropecuária
Embrapa Informática Agropecuária
Ministério da Agricultura, Pecuária e Abastecimento*

ISSN 1677-9274
Agosto, 2009

Documentos 90

Instalação e Configuração do PostgreSQL no Linux via código-fonte

Evandro Porto de Souza

Embrapa Informática Agropecuária
Campinas, SP
2009

Embrapa Informática Agropecuária
Área de Comunicação e Negócios (ACN)

Av. André Tosello, 209

Cidade Universitária "Zeferino Vaz" – Barão Geraldo

Caixa Postal 6041

13083-970 – Campinas, SP

Telefone (19) 3211-5700 – Fax (19) 3211-5754

URL: <http://www.cnptia.embrapa.br>

e-mail: sac@cnptia.embrapa.br

Comitê de Publicações

Kleber Xavier Sampaio de Souza (presidente)

Marcia Izabel Fugisawa Souza

Martha Delphino Bambini

Sílvia Maria Fonseca Silveira Massruhá

Stanley Robson de Medeiros Oliveira

Suzilei Almeida Carneiro (secretária)

Supervisão editorial: *Suzilei Almeida Carneiro*

Normalização bibliográfica: *Maria Goretti Gurgel Praxedes*

Revisão de texto: *Adriana Farah Gonzalez*

Editoração eletrônica: *Área de Comunicação e Negócios (ACN)*

Suplentes

Goran Neshich

Leandro Henrique Mendonça de Oliveira

1ª. edição on-line - 2008

Todos os direitos reservados.

A reprodução não-autorizada desta publicação, no todo ou em parte, constitui violação dos direitos autorais (Lei nº 9.610).

Souza, Evandro Porto

Instalação e configuração do PostgreSQL no Linux via código-fonte / Evandro Porto de Souza.- Campinas: Embrapa Informática Agropecuária, 2009.

19 p. il. - (Documentos / Embrapa Informática Agropecuária; 90)

ISSN 1677-9274

1. Sistema de gerenciamento de bancos de dados. 2. SGBD. 3. PostgreSQL.
4. Instalação via código-fonte. I. Título. II. Série.

CDD 005.756 21st ed

Autor

Evandro Porto de Souza

Bacharel em Ciência da Computação,
Analista B - Administração de Banco de Dados da
Embrapa Informática Agropecuária
Av. André Tosello, 209, Barão Geraldo
Caixa Postal 6041 - 13083-970 - Campinas, SP
Telefone: 19-3211-5772
e-mail: evandro@cnptia.embrapa.br

Apresentação

O uso e a adoção do Sistema Gerenciador de Banco de Dados (SGBD) PostgreSQL tem crescido progressivamente por todo o mundo, haja vista a sua capacidade, desempenho, estabilidade, robustez e adicionando a tudo isso, o fato de se tratar de uma ferramenta *open source*, ou seja, de código livre.

Este tutorial tem como objetivo identificar e descrever os principais passos e procedimentos necessários para realizar a instalação do SGBD PostgreSQL para o sistema operacional Linux, por meio da compilação de seu código-fonte, bem como, realizar a configuração necessária ao seu adequado funcionamento.

Esta tarefa pode trazer ganhos significativos quanto ao desempenho, bem como, dar maior versatilidade na manutenção e segurança do SGBD PostgreSQL.

Logo, espera-se com este documento, ter em mãos um guia que facilite, contribua e oriente os usuários interessados, em como proceder para uma instalação personalizada e adequada.

Eduardo Delgado Assad
Chefe-Geral

Sumário

Introdução.....	07
Instalação e Configuração do PostgreSQL via código-fonte.....	08
Instalar pacotes requeridos (obrigatórios).....	08
Instalar pacotes sugeridos (opcionais).....	09
Criar usuário no Linux.....	10
Buscar e preparar arquivo fonte.....	10
Compilar e instalar PostgreSQL.....	11
Configurações pós-instalação.....	13
Criação do Cluster de Banco de Dados.....	15
Auto-inicialização do Servidor.....	16
Conclusão.....	18
Referências Bibliográficas.....	19

Instalação e Configuração do PostgreSQL no Linux via código-fonte

Evandro Porto de Souza

Introdução

Apesar de exigir maior grau de dificuldade do que a instalação via pacotes, a instalação via código-fonte oferece uma infinidade de recursos em relação à primeira, tais como: definir a estrutura de diretórios para binários, bibliotecas, módulos, manuais e demais arquivos do SGBD conforme o desejado, o que facilita a organização, localização e manutenção do PostgreSQL; ativar suporte a um determinado idioma, para tradução de mensagens; construir suporte a determinadas linguagens ou módulos; definir alguns parâmetros padrões; além de conceber maior controle e segurança nas atualizações de versões do SGBD, evitando assim, atualizações automáticas indesejáveis por meio do sistema operacional e a conseqüente perda de bases de dados; entre outros.

Para este trabalho, foi utilizado o sistema operacional Ubuntu Server 8.04.1, contudo, este tutorial é totalmente aplicável e compatível a outras versões Ubuntu, bem como a outras distribuições Linux, respeitando, é claro, as particularidades de algumas distribuições, como a estrutura de diretórios destes. Quanto ao SGBD PostgreSQL, foi adotado a versão 8.3.4, lançada em Setembro/2008, contudo, este tutorial se aplica também à versões inferiores 8.x.x deste SGBD.

Para auxiliar o entendimento deste tutorial, segue abaixo, na Tabela 1, uma legenda com a descrição da simbologia empregada.

Tabela 1. Legenda da simbologia utilizada.

<i>Símbolo</i>	<i>Nome</i>	<i>Descrição</i>
#	Cerquilha	Login de usuário "root"
\$	Cifrão	Login de usuário comum
--	Dois traços	Comentário em linha de comando

Instalação e Configuração do PostgreSQL via código-fonte

Para facilitar o entendimento segue, logo abaixo, as principais etapas que serão realizadas durante este procedimento:

- Instalar pacotes requeridos (obrigatórios);
- Instalar pacotes sugeridos (opcionais);
- Criar usuário no Linux;
- Buscar e preparar arquivo fonte;
- Compilar e instalar PostgreSQL;
- Configurações pós-instalação;
- Criação do Cluster de Banco de Dados; e
- Auto-inicialização do Servidor.

Para as instalações realizadas em distribuições Linux com interfaces gráficas (como KDE ou Gnome), deve-se abrir um programa de terminal para linha de comando ou console shell, conforme a figura 1.



Fig. 1 - Console shell pronto para edição.

Em geral, grande parte das tarefas desta atividade devem ser executadas pelo usuário "root". Para isso, deve-se, caso o usuário tenha permissões de "sudo", executar comando conforme apresentado na Figura 2, ou caso contrário, executar a linha de comando "su - root", lembrando que na segunda opção é necessário o conhecimento da senha deste usuário.



Fig. 2 - Login do usuário "root".

Instalar pacotes requeridos (obrigatórios)

Uma vez conectado como usuário "root", a tarefa seguinte consiste em realizar a instalação dos pacotes exigidos para a compilação do código-fonte, vide Figura 3.

```
# apt-get install gcc g++ build-essential gzip bzip2 libreadline5-dev zlib1g-dev
```

Fig. 3 - Instalação de pacotes obrigatórios

A Tabela 2 descreve os pacotes acima mencionados.

Tabela 2. Descrição dos pacotes obrigatórios.

<i>Pacote</i>	<i>Descrição</i>
gcc / g++	Compiladores C/C++ da GNU
build-essential	Pacote contendo as ferramentas necessárias para compilação nas distribuições Debian
gzip / bzip2	Utilitários para compactação e descompactação
libreadline5	Biblioteca de suporte a históricos de comando no console
zlib1g-dev	Biblioteca de compressão

Instalar pacotes sugeridos (opcionais)

O passo seguinte é a instalação dos pacotes sugeridos, vide Figura 4, os quais não são obrigatórios para a compilação padrão do código-fonte, porém, são necessários para as configurações adotadas neste tutorial, conforme a descrição na tabela 3.

```
# apt-get install gettext openssl libssl-dev
```

Fig. 4 - Instalação de pacotes opcionais.

A Tabela 3 descreve os pacotes acima mencionados.

Tabela 3. Descrição dos pacotes opcionais.

<i>Pacote</i>	<i>Descrição</i>
gettext	Pacote de ferramentas para ativar suporte ao Idioma Nativo (NLS ¹). Mostrar mensagens em outra língua diferente do inglês (exemplo: português)
openssl	Pacote para permitir autenticação ou criptografia utilizando protocolo SSL ²
libssl-dev	Biblioteca de desenvolvimento para cabeçalhos de arquivos e documentação do protocolo ssl, exigida para compilar o PostgreSQL com OpenSSL.

Criar usuário no Linux

A próxima etapa é a criação do usuário/grupo “postgres” que trata-se do proprietário dos diretórios, arquivos e serviços do servidor PostgreSQL, conforme apresentado na Figura 5. O comando solicitará o fornecimento e a confirmação da senha deste usuário, além de solicitar o preenchimento de informações adicionais acerca do usuário, entretanto, tal preenchimento é opcional, basta acionar o botão “Enter” até finalizar o comando. Não é necessário criar o grupo “postgres” manualmente, uma vez que o comando “adduser” cria o grupo de mesmo nome do usuário automaticamente.

```
# adduser postgres
```

Fig. 5 - Criar usuário no linux com diretório HOME.

Por padrão, será criado o diretório HOME em “/home/postgres”. Caso queira adotar outro diretório como HOME, utilize a sintaxe apresentada na Figura 6, onde “/diretorio/” é o caminho o qual deseja criar o diretório HOME.

```
# adduser --home /diretorio/ --no-create-home postgres
```

Fig. 6 - Criar usuário no linux adotando outro diretório HOME.

Buscar e preparar arquivo fonte

Nesta etapa será realizada a busca em um repositório mais próximo ou que esteja disponível para que seja efetuado o *download* do arquivo fonte, vide Figuras 7 e 8. Para isso, pode ser utilizado os comandos “wget”, “ftp” ou “sftp”. Neste tutorial, foi adotado o comando “wget”, onde “/usr/src/” corresponde ao caminho que será salvo o arquivo. O segundo parâmetro refere-se a URL que hospeda tal arquivo. Os principais repositórios do PostgreSQL estão hospedados nos link's: <ftp://ftp2.br.postgresql.org/postgresql/source/> (nacional) e <http://www.postgresql.org/ftp/source/> (oficial). Os repositórios acima, disponibilizam os arquivos em 02 (dois) formatos bastante utilizados: bzip2 (*.bz2) e gzip (*.gz). Neste tutorial optou-se pelo formato bzip2 (*.bz2) por este possuir arquivos com tamanho menor do que o gzip.

```
# wget -P /usr/src/ ftp://ftp2.br.postgresql.org/postgresql/source/v8.3.4/postgresql-8.3.4.tar.bz2
```

Fig. 7 - *Download* do arquivo fonte do PostgreSQL em bzip2.

```
➤ wget -P /usr/src/ ftp://ftp2.br.postgresql.org/postgresql/source/v8.3.4/postgresql-8.3.4.tar.gz
```

Fig. 8 - Download do arquivo fonte do PostgreSQL em grip.

Realizado o *download* do arquivo fonte, o passo seguinte é acessar o diretório onde o arquivo foi salvo (vide Figura 9) e descompactá-lo. Caso tenha realizado o *download* do arquivo no formato bzip2, execute o comando conforme a Figura 10, ou caso o arquivo esteja no formato gzip, descompacte conforme ilustrado na Figura 11.

```
➤ cd /usr/src/
```

Fig 9 - Acessar diretório "source".

```
➤ tar jxvf postgresql-8.3.4.tar.bz2
```

Fig 10 - Descompactar arquivo fonte em bzip2.

```
➤ tar zxvf postgresql-8.3.4.tar.gz
```

Fig 11 - Descompactar arquivo fonte em gzip.

Compilar e instalar PostgreSQL

Antes de iniciar a compilação do código-fonte é necessário criar a estrutura de diretórios dos arquivos do PostgreSQL, em outras palavras, definir o caminho de instalação do SGBD. Visando centralizar e facilitar a localização dos arquivos, neste tutorial a estrutura de diretórios será criada no diretório HOME do usuário Linux "postgres", conforme ilustrado nas Figuras 12 e 13. Entretanto, não se trata de uma regra, o PostgreSQL pode ser instalado em outro diretório, de acordo com a preferência do usuário.

```
➤ mkdir /home/postgres/postgresql
```

Fig. 12 - Criação do diretório "/postgresql."

```
# mkdir /home/postgres/postgresql/8.3
```

Fig. 13 - Criação do diretório “/8.3”.

Uma vez criado o diretório principal do PostgreSQL, é preciso atribuir tal diretório e seus subdiretórios ao seu respectivo proprietário, o usuário Linux “postgres”, conforme apresentado na Figura 14.

```
# chown -R postgres:postgres /home/postgres/postgresql/
```

Fig. 14 - Atribuir proprietário “postgres” ao diretório “postgresql”.

Dando continuidade, é necessário acessar o diretório do arquivo fonte descompactado, conforme Figura 15.

```
# cd /usr/src/postgresql-8.3.4
```

Fig. 15 - Acessar diretório do arquivo fonte descompactado.

Iniciando o processo de compilação, deve-se configurar a árvore de fontes, conforme Figura 16, onde “--prefix=” é o parâmetro que indica o caminho onde será realizada a instalação dos binários, bibliotecas e demais arquivos do servidor PostgreSQL. As opções do comando “./configure” são opcionais, caso não seja informada nenhuma opção, será feita a configuração padrão, como por exemplo, a instalação desmembrada dos componentes do PostgreSQL, conforme a estrutura de diretório da distribuição Linux, o que dificulta a organização e localização de arquivos pelo usuário. Logo, é recomendado a adoção de no mínimo as opções da Figura 16 (para ver mais opções, consulte o manual PostgreSQL). A opção “--with-openssl” garante suporte às conexões criptografadas utilizando protocolo SSL e a opção “--enable-nls=pt_BR” ativa o Suporte ao Idioma Nativo para que as mensagens sejam apresentadas em português. Entretanto, caso seja adotada as opções “-with-openssl” e “--enable-nls=pt_BR”, mencionadas acima, deve-se obrigatoriamente instalar os pacotes do item 3.2.

```
# ./configure --prefix=/home/postgres/postgresql/8.3 --with-openssl --enable-nls=pt_BR
```

Fig. 16 - Configurar árvore de fontes.

A etapa seguinte é construir a árvore de fontes já configurada, conforme ilustrado na Figura 17.



```
• make
```

Fig. 17 - Construir árvore de fontes.

E finalmente a instalação árvore de fontes do PostgreSQL, vide Figura 18. Opcionalmente, após a instalação ,pode ser liberado espaço removendo da árvore de fontes ,os arquivos construídos ,vide Figura 19 .



```
• make install
```

Fig. 18 - Instalar árvore de fontes.

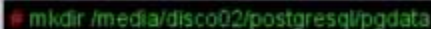


```
• make clean
```

Fig. 19 - Limpar arquivos construídos.

2.6. Configurações pós-instalação

Após finalizar a instalação, é necessário algumas configurações antes de se iniciar o *cluster* de banco de dados e criar um banco de dados propriamente dito. O primeiro passo é definir o caminho onde serão armazenados os arquivos físicos dos banco de dados. O ideal é não adotar o padrão do SGBD e sim definir um caminho, preferencialmente em um outro disco ou outra partição. Isso dará maior segurança aos dados e maior controle na hora de realizar manutenções e cópias de segurança. Logo, será criado o diretório “pgdata” no disco “disco02”, conforme apresentado na Figura 20. Caso não tenha outro disco, ou outra partição livre, o diretório deve ser criado no local de preferência do usuário.



```
• mkdir /media/disco02/postgresql/pgdata
```

Fig. 20 - Criação de diretório “pgdata”.

Uma vez criado o diretório “pgdata”, é preciso atribuir tal diretório ao seu respectivo proprietário, o usuário Linux “postgres”, conforme apresentado na Figura 21.



```
chown -R postgres postgres /media/disco02/postgresql/pgdata/
```

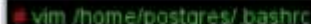
Fig. 21 - Atribuir proprietários “postgres” ao diretório “pgdata”.

O próximo passo consiste na configuração das variáveis de ambiente. Estas podem ser globais, criadas no arquivo “/etc/bash.bashrc” (vide Figura 22) ou no arquivo “.bashrc” do usuário Linux “postgres”, conforme ilustrado na Figura 23.



```
vim /etc/bash.bashrc
```

Fig. 22 - Abrir arquivo “.bashrc” global.



```
vim /home/postgres/.bashrc
```

Fig. 23 - Abrir arquivo “.bashrc” do usuário “postgres”.

Uma vez dentro do arquivo “.bashrc”, basta inserir as linhas de comando apresentadas na Figura 24 e salvar o arquivo. Estas variáveis criadas facilitarão o manuseio com os utilitários do PostgreSQL, evitando assim a necessidade de se digitar o caminho dos binários ou do diretório de dados.



```
PATH=/home/postgres/postgresql/8.3/bin:$PATH
export PATH

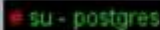
MANPATH=/home/postgres/postgresql/8.3/man
export MANPATH

PGDATA=/media/disco02/postgresql/pgdata
export PGDATA
```

Fig. 24 - Configurar variáveis de ambiente.

Criação do Cluster de Banco de Dados

Com todo o ambiente pronto, a etapa seguinte é a criação do cluster ou agrupamento de banco de dados. Essa tarefa deve ser executada somente pelo usuário Linux “postgres”, vide Figura 25, uma vez que somente este usuário tem tal permissão.



```
su - postgres
```

Fig. 25 - Conectar com usuário “postgres”.

Conectado com o usuário “postgres”, deve-se agora executar o comando “initdb” para inicializar o agrupamento do servidor PostgreSQL, conforme Figura 26. Na opção “-D” deste comando deve-se fornecer o caminho onde foi criado o diretório “pgdata”. Já a opção “-W” solicitará o preenchimento e a confirmação de uma senha para o usuário PostgreSQL “postgres”. Atenção para não confundir o usuário do Linux “postgres” (dono dos serviços do PostgreSQL), com o usuário do PostgreSQL “postgres” (superusuário do banco de dados). Apesar do mesmo nome são usuários diferentes e com senhas independentes.



```
initdb -D /media/disco02/postgresql/pgdata -W
```

Fig. 26 - Criar agrupamento do banco de dados.

Após a execução do “initdb”, é preciso inicializar o serviço do PostgreSQL. Isso pode ser feito por meio do utilitário “pg_ctl”, conforme Figura 27, onde será gerado um arquivo de log ou pode ser realizado simplesmente como apresentado na Figura 28, uma vez que as variáveis de ambiente já foram criadas. Logo, basta passar um dos parâmetros: start, stop e reload para inicializar, parar e recarregar o serviço do PostgreSQL respectivamente.



```
pg_ctl -D /media/disco02/postgresql/pgdata -l logfile start
```

Fig. 27 - Inicializar serviço do PostgreSQL com log.



```
pg_ctl start
```

Fig. 28 - Inicializar serviço do PostgreSQL.

Pronto. Seu servidor de banco de dados PostgreSQL já está funcionando. Caso deseje parar o serviço, vide Figura 29.

```
$ pg_ctl stop
```

Fig. 29 - Parar serviço do PostgreSQL.

Auto-inicialização do Servidor

Como já mencionado, seu servidor está pronto e em pleno funcionamento. Entretanto, a cada vez que seu computador for reiniciado, seja voluntariamente ou por motivo de força maior (como falta de energia elétrica ou oscilações na rede elétrica), será necessário inicializar o serviço manualmente, conforme ilustrado nas Figuras 27 e 28. Nesta etapa será apresentado um script para automatizar a inicialização do servidor PostgreSQL como serviço do Linux, para que isso ocorra de forma automática e transparente ao usuário. Esta tarefa deve ser executada como usuário “root”, conforme a Figura 30.

```
$ su - root
```

Fig. 30 - Conectar com usuário “root”.

Já conectado como usuário “root”, deve-se copiar o arquivo denominado “linux”, localizado no diretório do arquivo fonte do PostgreSQL, para o diretório “init.d”, vide Figura 31.

```
$ cp /usr/src/postgresql-8.3.4/contrib/start-scripts/linux /etc/init.d
```

Fig. 31 - Copiar script de inicialização.

Em seguida, renomeie o arquivo para “postgresql”, conforme figura 32.

```
$ mv /etc/init.d/linux /etc/init.d/postgresql
```

Fig. 32 - Renomear script de inicialização.

Após, abra o arquivo em modo de edição, conforme a Figura 33.

```
vim /etc/init.d/postgresql
```

Fig. 33 - Abrir script de inicialização.

Dentro do arquivo “postgresql”, altere as seguintes variáveis: “prefix”, inserindo o caminho de instalação do PostgreSQL; “PGDATA” inserindo o caminho do diretório de dados “pgdata”; e por fim a variável “PATH” acrescentando o valor definido na variável “prefix”, conforme ilustrado na Figura 34. Em seguida salve e feche o arquivo.

```
--Installation prefix
prefix=/home/postgres/postgresql/8.3

--Data directory
PGDATA="/media/disco02/postgresql/pgdata"

-- The path that is to be used for the script
PATH=/home/postgres/postgresql/8.3/bin:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin
```

Fig. 34 - Alterar variáveis do script de inicialização.

O próximo passo é atribuir a permissão de execução para o dono do arquivo, conforme Figura 35.

```
chmod u+x /etc/init.d/postgresql
```

Fig. 35 - Atribuir permissão de execução.

A seguir, acessar o diretório “init.d” (vide Figura 36) .

```
cd /etc/init.d/
```

Fig. 36 - Acessar diretório “init.d”.

Dentro do diretório “init.d”, deve se executar o comando “update-rc.d” para criação de link's simbólicos do “runlevel's” (mecanismo para inicializar e finalizar aplicações de serviços do Linux).

```
update-rc.d postgresql defaults
```

Fig. 37 - Criação de link's simbólicos do “runlevel's”.

Agora é necessário reiniciar o seu computador para validar a operação.

A terminal window with a black background and green text. The word "reboot" is displayed in a monospaced font.

Fig. 38 - Reiniciar o computador.

Após esse procedimento, seu servidor PostgreSQL será inicializado automaticamente sempre que o computador for reiniciado. Além disso, será também possível iniciar ou parar o serviço pelo diretório "init.d", conforme as figuras 39 e 40.

A terminal window with a black background and green text. The command "/etc/init.d/postgresql start" is displayed in a monospaced font.

Fig. 39 - Inicializar serviço pelo "init.d".

A terminal window with a black background and green text. The command "/etc/init.d/postgresql stop" is displayed in a monospaced font.

Fig. 40 - Parar serviço pelo "init.d".

Conclusão

Em síntese, este tutorial resume os principais procedimentos da compilação do código-fonte para instalação e configuração do SGBD PostgreSQL que a cada dia recebe mais adeptos, esperando assim que se tenha uma instalação segura e com o total controle por parte do administrador do banco de dados.

A adoção da instalação do PostgreSQL via código-fonte, assegura ainda uma maior comodidade quanto à manutenção, maior gerência quanto a atualização e melhor desempenho do SGBD.

Assim, este tutorial visa simplificar este processo de instalação, que comumente é evitado em virtude do desconhecimento de tais informações, e encorajar e difundir o uso dessa boa prática relacionada a Bancos de Dados.

Referências Bibliográficas

PEREIRA NETO, A. *PostgreSQL: Técnicas Avançadas: Versões Open Source 7.x – Soluções para Desenvolvedores e Administradores de Bancos de Dados*. São Paulo: Érica, 2003. 284p.

POSTGRESQL. *The world's most advanced open source database*. Disponível em: <<http://www.postgresql.org/>> Acesso em: 22 mar. 2007.

POSTGRESQLBRASIL. *Sítio da comunidade PostgreSQL brasileira*. Disponível em: <<http://www.postgresql.org.br/>> Acesso em: 10 fev. 2008.

THE POSTGRESQL GLOBAL DEVELOPMENT GROUP. *Documentação do PostgreSQL 8.0.0*. Tradução: Haley Pacheco de Oliveira. Rio de Janeiro, 2005. 1205p.



Informática Agropecuária

**Ministério da
Agricultura, Pecuária
e Abastecimento**

