<https://support.plesk.com/hc/en-us/articles/115003321434-How-to-enable-remote-access-to-PostgreSQL-server-on-a-Plesk-server->

<https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es/Configurar_acessos>

Answer

**Note:** the instructions below were tested on a Plesk server, however, they can be applicable to a non-Plesk server as well keeping in mind possible different path to the files.

To enable remote access to PostgreSQL server:

1. Connect to the PostgreSQL server [via SSH](https://support.plesk.com/hc/en-us/articles/115000172834).
2. Get location of postgresql.conf file by executing command (it should be something like /var/lib/pgsql/data/postgresql.conf):

# find / -name "postgresql.conf"

1. [Open **postgresql.conf** file](https://support.plesk.com/hc/en-us/articles/360001084114) and add the following line to the end:

listen\_addresses = '\*'

1. Add the following line to the end of /var/lib/pgsql/data/pg\_hba.conf file:

host all all **203.0.113.2/32 md5**

* + **203.0.113.2/32** is the remote IP from which connection is allowed. If you want to allow connection from any IP specify **0.0.0.0/0** .
  + **md5** is the authentication method, which requires the client to supply a double-MD5-hashed password for authentication.

Note: In the above configuration user john.doe from database example1 has access to database example2 and vice versa. To allow user john.doe remotely connect to example1 database only, specify directives as following in the end of /var/lib/pgsql/data/pg\_hba.conf file:  
host samerole all 203.0.113.2/32 md5

As for other authentication methods refer to [PostgreSQL documentation](https://www.postgresql.org/docs/9.6/static/auth-pg-hba-conf.html).

1. Restart PostgreSQL server to apply the changes:

# service postgresql restart

# PostgreSQL Prático/Configurações/Configurar acessos

< [PostgreSQL Prático](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico" \o "PostgreSQL Prático)‎ | [Configurações](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es)

* [**Configurações**](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es)
* [Configurações](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es)
* [Configurações Diversas](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es/Configura%C3%A7%C3%B5es_Diversas)

[Saltar para a navegação](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es/Configurar_acessos#mw-head)[Saltar para a pesquisa](https://pt.wikibooks.org/wiki/PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es/Configurar_acessos#p-search)

## 11.1 - pg\_hba.conf[[editar](https://pt.wikibooks.org/w/index.php?title=PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es/Configurar_acessos&veaction=edit&section=1) | [editar código-fonte](https://pt.wikibooks.org/w/index.php?title=PostgreSQL_Pr%C3%A1tico/Configura%C3%A7%C3%B5es/Configurar_acessos&action=edit&section=1)]

Ao instalar o PostgreSQL 8.1.4 via fontes ele cria (e alerta) o arquivo pg\_hba.conf com autenticação do tipo trust (sem senha mas somente para conexão local).

Para autenticar exigindo um dos tipos com senha, devemos antes, ainda no trust, alterar os usuários adicionando senha:

ALTER ROLE nomeuser WITH PASSWORD 'senhadopg';

Somente então devemos alterar o pg\_hba.conf para pedir senha e restartar o PostgreSQL.

As configurações principais são feitas nos arquivos pg\_hba.conf e postgresql.conf. Se instalado através dos fontes ficam no subdiretório data de instalação do PostgreSQL, normalmente em /usr/local/pgsql. Se instalado via binários da distribuição vai variar com a distribuição. No Slackware estão no diretório /usr/share/postgresql.

O pg\_hba.conf controla que máquinas terão acesso ao PostgreSQL e a autenticação dessas máquinas clientes (sem autenticação ou através de outras formas, trust, md5, crypt, etc).

O pg\_hba.conf é muito rico e podemos controlar o acesso pelo IP, pela máscara, pelo banco, pelo usuário, pelo método (trust, md5, password, etc).

**Resumo sobre o pg\_hba.conf**

Arquivo de configuração da autenticação dos clientes do PostgreSQL

Este arquivo controla:

- Quais hosts têm permissão de conectar

- Como os clientes são autenticados

- Nomes dos usuários que podem usar

- Quais bancos eles podem acessar

Os registros deste arquivo têm as formas seguintes:

1. local DATABASE USER METHOD [OPTION]

Este método de conexão é para uso somente através sockets de domínio UNIX

Sem um registro tipo local essa conexão é negada

1. host DATABASE USER CIDR-ADDRESS METHOD [OPTION]

Conexão via TCP/IP. Este tipo de conexão só é possível quando o valor do parâmetro listen\_address (IP) é adequado no postgresql.conf. Por default somente localhost é permitido. São permitidas conexões com ou sem SSL.

1. hostssl DATABASE USER CIDR-ADDRESS METHOD [OPTION]

Semelhante ao host, sendo que neste usa-se criptografia SSL.

1. hostnossl DATABASE USER CIDR-ADDRESS METHOD [OPTION]

Apenas para conexões que não usam SSL.

1. DATABASE can be "all", "sameuser", "samerole", a database name, or
2. a comma-separated list thereof.

DATABASE pode ser:

- "all"

- "sameuser" (para quando o usuário tiver o mesmo nome do banco)

- "samerole" (para quando o usuário for membro de uma role (papel) com o mesmo nome de um banco)

- um nome de banco ou

- uma lista separados por vírgula

USER pode ser:

- "all" (para qualquer usuário)

- um nome de usuário

- um nome de grupo prefixado por "+"

- uma lista separados por vírgula

Tanto no campo DATABASE quanto no USER podemos também escrever um nome de arquivo prefixado com "@" para incluir nomes em um arquivo separado

CIDR-ADDRESS especifica uma faixa de IPs. Através de um endereço de IP e de uma máscara que é um inteiro (entre 0 e 32 para IPV4 ou de 128 para IPV6),que especifica o número de bits significativos da máscara.

Alternativamente podemos escrever o IP separado da máscara em colunas diferentes para especificar o conjunto de hosts.

Exemplos (IPV4):

172.20.143.89/32 para um único host

172.20.143.0/24 para uma rede

O campo CIDR-ADDRESS somente se aplica aos registros host, hostssl e hostnossl.

IP-address e IP-mask

Estes campos podem ser utilizados como alternativa à notação CIDR-ADDRESS. ao invés

de especificar o comprimento da máscara, a atual máscara é especificada em uma coluna separada.

**Exemplo:**

255.0.0.0 representa uma máscara CIDR em IPV4 com comprimenro 8

255.255.255.255 representa uma máscara CIDR com comprimenro 32

172.20.143.89 255.255.255.255

Estes campos somente se aplica aos registros host, hostssl e hostnossl.

METHOD pode ser "trust", "reject", "md5", "crypt", "password",

1. "krb5", "ident", ou "pam"

trust - permite conexão a qualquer um incondicionalmente (sem senha)

reject - rejeita conexão incondicionalmente para o user/host especificado

crypt - recomendada somente para versões inferiores a 7.2. Atualmente recomenda-se md5

krb5 - somente disponível para conexões via TCP/IP

ident - Obtém o nome do usuário do sistema operacional. Para conexões TCP/IP contacta o servidor ident no cliente. Para conexões locais, recebendo este do sistema operacional.

pam - usando o serviço PAM (Pluggable Authentication Modules) do sistema operacional

password - envia senha em texto claro

md5 - deve ser preferido, pois envia senhas criptografadas

1. TYPE DATABASE USER CIDR-ADDRESS METHOD
2. IPv4 local connections:

host all all 127.0.0.1/32 md5

1. IPv6 local connections:
2. host all all ::1/128 md5

Alerta: este arquivo é examinado sequencialmente para cada tentativa de conexão.

**A ordem dos registros é significativa.**

Caso um registro que venha primeiro entre em conflito com outro que vem em seguida, o primeiro será executado e o segundo, não.

Exemplos:

1. TYPE DATABASE USER CIDR-ADDRESS METHOD

local all all md5

local all all trust

Acima todas as conexões locais exigirão senha md5.

local all all trust

local all all md5

Já este acima aceitará todas as conexões locais incondicionalmente (sem senha)

A segunda linha será ignorada.

Veja vários exemplos do pg\_hba.conf e mais detlahes no capítulo 20 do manual oficial do PostgreSQL em

Observe que conexões locais não usam o campo CIDR-ADDRESS.

**Exemplos no pg\_hba.conf:**

Conexão local via socket

local all all md5

Conexão local via TCP/IP

host all all 127.0.0.1/32 md5

Conexão local via TCP/IP com máscara separada

host all all 127.0.0.1 255.255.255.255 md5

Conexão para uma rede (ident sameuser) via TCP/IP

host banco usuario 192.168.93.0/24 ident sameuser

Conexão rejeitada via TCP/IP para o IP, usuário e banco

host banco usuario 192.168.93.1/32 reject

**ident**

Obtém o nome de usuário do sistema operacional do cliente (para conexões TCP/IP fazendo contato com o servidor de identificação no cliente, para conexões locais obtendo a partir do sistema operacional) e verifica se o usuário possui permissão para se conectar como o usuário de banco de dados solicitado consultando o mapa especificado após a palavra chave ident.

Exemplo para Ubuntu (facilmente adaptável para outras distros e SOs).

Se instalado pelo repositório:

- sudo gedit /etc/postgresql/8.1/main/postgresql.conf

Altere:

1. listen\_addresses = 'localhost'

para:

listen\_addresses = '\*'

Aqui também existem outras importantes configurações, como datestyle = 'sql, dmy' # Datas no estilo brasileiro dd/mm/aaaa

client\_encoding = latin1 # suporte à nossa acentuação

- sudo gedit /etc/postgresql/8.1/main/pg\_hba.conf

1. TIPO BANCO USUÁRIO CIDR-ADDRESS MÉTODO
2. "local" é para domínios Unix somente com conexões via socket
3. não requer IP

local all all md5

1. Conexões locais via IPv4:

host all all 127.0.0.1/32 md5

1. Conexões locais via IPv6:

host all all ::1/128 md5

1. Conexões para todos os IPs de uma subrede (10.0.0.0)
2. TYPE DATABASE USER IP-ADDRESS IP-MASK METHOD

host all all 10.43.0.0 255.0.0.0 md5

Opção para permitir acesso através de toda a internet:

1. Aceitar toda a Internet, exceto 200.217.23.234
2. TYPE DATABASE USER IP-ADDRESS IP-MASK METHOD

host all all 200.217.23.234 255.255.255.255 reject

host all all 0.0.0.0 0.0.0.0 md5