**Mastering pg\_hba.conf with Maximum Possible Settings**

**Objective**

* Understand how PostgreSQL uses pg\_hba.conf for authentication
* Practice different authentication methods (trust, md5, scram-sha-256, peer, ident, ldap, cert)
* Configure access by host, IP, network, and database
* Test settings and troubleshoot login scenarios

**Location of pg\_hba.conf**

On **Windows**, the file is typically found in:

C:\Program Files\PostgreSQL\<version>\data\pg\_hba.conf

**Structure of pg\_hba.conf Entries**

Each line has the following format:

<connection\_type> <database> <user> <address> <auth\_method> [options]

**Exercise 1: Local Trusted Connection (Linux only)**

local all all trust

**Explanation:**

* local means Unix socket
* No password required
* Not supported on Windows

Test:

psql -U postgres -d mydb

**Exercise 2: Host Connection with MD5 Password**

host all all 127.0.0.1/32 md5

host all all ::1/128 md5

**Explanation:**

* Requires password (hashed with MD5)
* IPv4 and IPv6 loopback

Test:

psql -U testuser -h 127.0.0.1 -d testdb

**Exercise 3: SCRAM-SHA-256 Authentication**

Set in postgresql.conf:

password\_encryption = scram-sha-256

Create a user with SCRAM password:

CREATE ROLE secure\_user LOGIN PASSWORD 'securepass';

Then, in pg\_hba.conf:

host all secure\_user 127.0.0.1/32 scram-sha-256

**Exercise 4: Peer Authentication (Linux only)**

local all all peer

**Explanation:**

* Maps the OS username to PostgreSQL username
* Only works on local Unix socket
* Not supported on Windows

**Exercise 5: LDAP Authentication**

Example configuration:

host all all 0.0.0.0/0 ldap ldapserver=ldap.example.com

**Explanation:**

* Uses LDAP server to authenticate users

**Exercise 6: Certificate Authentication**

Requires SSL to be enabled in postgresql.conf:

ssl = on

In pg\_hba.conf:

hostssl all cert\_user 192.168.1.0/24 cert

**Also required:**

* Client certificate signed by a trusted CA
* ssl\_cert\_file, ssl\_key\_file, and ssl\_ca\_file properly set

**Exercise 7: Ident Authentication (Linux only)**

host all all 127.0.0.1/32 ident

Used with pg\_ident.conf, where you map OS usernames to database usernames:

**Example pg\_ident.conf:**

# MAPNAME SYSTEM-USERNAME PG-USERNAME

localmap linuxuser pguser

**Exercise 8: Restrict by Database**

host testdb all 0.0.0.0/0 md5

Only allows access to testdb, all users, from any IP using md5.

**Exercise 9: Restrict by IP and Subnet**

host all all 10.10.10.0/24 md5

Allows access only from the 10.10.10.0/24 subnet.

**Exercise 10: Reject All External Connections**

Add this at the bottom:

host all all 0.0.0.0/0 reject

Prevents outside access.

**Exercise 11: Role-Based Rule**

host all readonly\_user 192.168.0.0/16 md5

Grants access to one role (readonly\_user) from a specific network.