give me the complete check list

Here is a complete checklist to set up and connect securely to your PostgreSQL database on an AWS EC2 Linux instance, covering key steps from server setup to troubleshooting:

Complete Checklist for AWS EC2 PostgreSQL Setup & Connection

1. EC2 Instance Setup

- Ensure the instance is **running** and has a **public IPv4 address**.
- Confirm your instance OS is supported by PostgreSQL (e.g., Amazon Linux 2, Ubuntu).
- Connect to your instance via SSH:
  - Use correct `.pem` key with proper permissions.
  - Handle path quoting properly on Windows (e.g., `"C:\path\to\key.pem"`).

2. Security Group Configuration

- Identify the security group attached to your instance.
- Check **Inbound Rules**:
  - Allow SSH (port 22) from your IP address.
  - Allow PostgreSQL (port 5432) from your IP address or allowed networks.
- Check **Outbound Rules**:
  - Open outbound traffic (usually open by default).

3. PostgreSQL Installation & Service

- Install PostgreSQL if not installed:
  - Amazon Linux example:

    ```
    sudo amazon-linux-extras install postgresql14
    sudo yum install postgresql-server postgresql-contrib
    ```

  - Ubuntu example:

    ```
    sudo apt update
    sudo apt install postgresql postgresql-contrib
    ```

- Initialize and start PostgreSQL service:

```
sudo systemctl enable postgresql
sudo systemctl start postgresql
```

- Check service status:

  ```
  sudo systemctl status postgresql
  ```

4. PostgreSQL Configuration

- Edit `postgresql.conf` to allow remote connections:

  ```
  listen_addresses = '*'
  ```

- Edit `pg_hba.conf` to allow your IP address/subnet to connect:

  ```
  host    all             all             your.ip.address/32          md5
  ```

- Restart PostgreSQL after config changes:

  ```
  sudo systemctl restart postgresql
  ```

5. User and Database Setup

- Create PostgreSQL user and database if needed:

  ```
  sudo -u postgres createuser yourusername
  sudo -u postgres createdb yourdbname
  ```

- Set password for the user:

  ```
  sudo -u postgres psql
  \password yourusername
  \q
  ```

6. Verify Local Connection on EC2

- Test connection locally:

  ```
  psql -U yourusername -d yourdbname
  ```

7. Test Remote Connection

- On your local machine, test SSH connection:

  ```
  ssh -i "path/to/key.pem" ec2-user@your.ec2.ip.address
  ```

- Test PostgreSQL connection from your local machine or client software (e.g., pgAdmin) using:

    - Hostname: EC2 public IP
    - Port: 5432
    - Username/password as configured
- Troubleshoot based on error messages:
    - Timeout: Security groups or firewall blocks
    - Authentication error: Check PostgreSQL user and password
    - Connection refused: PostgreSQL not running or not listening on IP

8. Troubleshooting Common Issues

- SSH Timeout:
    - Check security group SSH rules
    - Verify local network/firewall allows outbound port 22
- PostgreSQL Timeout:
    - Confirm port 5432 open in security group
    - Check PostgreSQL listening on all IPs
- No socket or server connection error locally:
    - Ensure PostgreSQL service is running
    - Check logs for errors (`sudo journalctl -u postgresql`)
- Permission denied on `.pem` key files (Windows):
    - Use proper quoting on file path or move the key to a simple path without spaces

This checklist should guide you through setting up, configuring, and connecting your PostgreSQL server on AWS EC2 with troubleshooting tips included.

If you want, I can provide commands/scripts for any specific step. Would you like that?