

M2R Network Edition

Administrator's Guide

Server Installation · Configuration · Network Setup
License Management · Backup & Recovery · Security
Performance Tuning · Troubleshooting

1. Introduction

About M2R Network Edition

M2R Network Edition is a multi-user credit reporting application that enables multiple users to simultaneously access and manage Metro 2® credit data through a centralized PostgreSQL database server.

Key Features

- **Multi-User Access:** 10+ concurrent users
- **Record Locking:** Prevents conflicting edits
- **Audit Logging:** Complete change history
- **Multi-Company Support:** Manage multiple companies in one database. (*Multi-Bureau License tier only*)
- **Centralized Data:** All data stored on server
- **Role-Based Security:** PostgreSQL user permissions

Architecture Overview

M2R Network Edition uses client-server architecture. Multiple client workstations connect over TCP/IP (Port 5432) to a central network server running PostgreSQL 16 with the M2R database. All data is stored centrally on the server, and clients access it using the PostgreSQL wire protocol.

Note

This guide is intended for system administrators responsible for server setup, IT professionals managing M2R Network deployment, database administrators overseeing PostgreSQL instances, and support staff troubleshooting installation or configuration issues.

2. System Requirements

Server Requirements

Minimum Configuration

- Windows Server 2019/2022 or Windows 10/11 Pro (64-bit)
- 8 GB RAM
- 50 GB available disk space
- 2 CPU cores
- Static IP address or stable hostname
- Network connectivity to all client machines

Recommended Configuration

- Windows Server 2022 (64-bit)
- 16 GB RAM (for 10+ concurrent users)
- 100 GB SSD storage
- 4+ CPU cores
- Gigabit Ethernet
- RAID 1/5 for data redundancy
- UPS for power protection

Client Requirements

Minimum Configuration

- Windows 10 (64-bit) version 1809 or later
- Windows 11 (any version)
- 4 GB RAM
- 500 MB available disk space
- Network connectivity to server

Recommended Configuration

- Windows 11
- 8 GB RAM
- SSD storage
- Stable network connection (LAN preferred over Wi-Fi)

Network Requirements

- TCP/IP network connectivity between all clients and server
- Port 5432 accessible from client machines to server
- Minimum bandwidth: 1 Mbps per concurrent user
- Recommended: 100 Mbps LAN connection
- VPN required for remote/internet access

Database Requirements

- PostgreSQL 16 (included with server installer)
- Minimum database size: 100 MB (empty)
- Growth estimate: ~500 KB per 1,000 accounts with full segment data

3. Server Installation

Pre-Installation Checklist

1. Administrator Access: Ensure you have local administrator privileges on the server
2. Antivirus: Temporarily disable antivirus during installation
3. Windows Updates: Apply all critical Windows updates
4. Backup: Back up any existing PostgreSQL data if upgrading
5. Firewall: Note current firewall configuration
6. Network: Configure server with static IP address

Installation Steps

Step 3-1: Run Server Installer

1. Download M2R_Server_Setup_x.x.x.exe
2. Right-click the installer → Run as Administrator
3. If you get a User Account Control popup window click Yes

Step 3-2: Accept License Agreement

1. Read the license agreement
2. Check "I accept the agreement"
3. Click Next

Step 3-3: Select Installation Directory

1. Default: C:\Program Files\M2Reporter Server
2. Ensure the selected drive has at least 50 GB free space
3. Click Next

Step 3-4: Select Components

1. Components to install:
 - ✓ PostgreSQL 16 Database Server (required)
 - ✓ M2R Server Setup Tool (required)
 - ✓ Documentation (recommended)
2. Click Next.

Step 3-5: PostgreSQL Installation

The installer will automatically check if PostgreSQL 16 is already installed, install it silently if not found, configure the PostgreSQL service to start automatically, and set the default superuser password to “postgres” (you should change this after setup).

Important

This step may take 5–10 minutes. Do not interrupt the installation process.

Step 3-6: Complete Installation

1. Finish installation wizard prompts
2. When installation is complete, check "Configure M2Reporter Server" and click Finish.

4. Initial Configuration

After installation, you must complete the M2R Network Server Setup to configure the database.

If the setup does not open go to **Start Menu** → M2Reporter Server → M2R Server Setup
Or run: C:\Program Files\M2Reporter Server\M2Reporter_Server_Setup.exe

Configuration Wizard

Step 4-1: PostgreSQL Admin Connection

Enter the PostgreSQL superuser credentials below:

Field	Value	Notes
Host	localhost	Use localhost for local server
Port	5432	Default PostgreSQL port
Database Name	m2reporter	Use default
Admin User	postgres	PostgreSQL superuser
Admin Password	postgres	Default password (change after setup)

Continued on the next page.

PostgreSQL Connection

Host:
Port:

Database:

PostgreSQL Admin (for initial setup)

Admin User:
Password:

Click Test Connection to verify. You should see “Connection successful!”

Note

If connection fails verify PostgreSQL service is running, check firewall isn't blocking port 5432 locally, and verify the password is correct.

Step 4-2: Create M2R Database

Configure the application database:

Field	Value	Recommendation
Username	m2Reporter	Use default
Database Password	(choose strong password)	Min 8 chars, mixed case, numbers, symbols
Confirm	Enter chosen password again	Click Show button to verify data

M2R Database User

Username:
Password:

Confirm:

Continued on the next page.

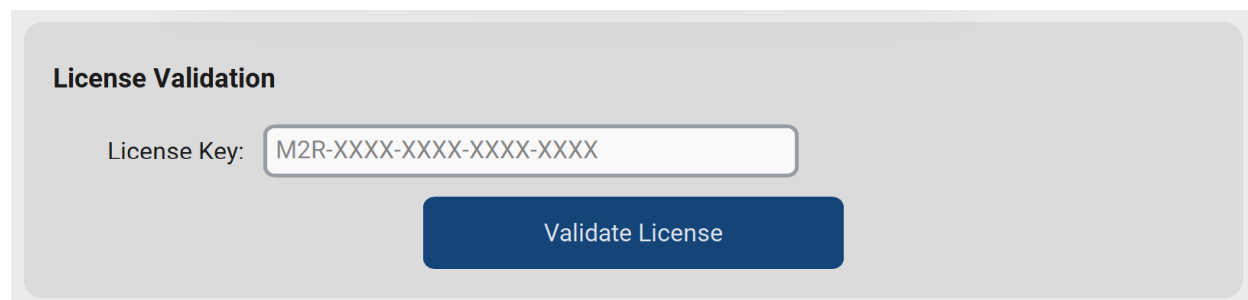
⚠ Critical

Write down the database password! You will need to provide it to all client users. Communicate the password separately via encrypted email or secure messaging — it is NOT included in the client config file.

Click Create Database & User. The wizard will create the database, create the M2R user, and grant necessary privileges. You should see "...configured successfully!"

Step 4-3: Validate License

Enter your M2R Network Edition license key in the format M2R-XXXX-XXXX-XXXX-XXXX and click Validate License.



License Validation

License Key:

Validate License

💡 Note

Step 3 performs an online validity check only. License data is stored in the database during Step 4 (Initialize Schema).

The system will connect to the license validation server, verify the license key format (M2R-XXXX-XXXX-XXXX-XXXX), verify the license key is recognized and active, and check the license type and expiration.

License information displayed includes: License Type (Network Edition), Max Concurrent Users (typically 10), Expiration Date, and Company Name.

💡 Note

If validation fails: check internet connectivity, verify the license key is entered correctly in M2R-XXXX-XXXX-XXXX-XXXX format (use copy/paste), ensure the license hasn't expired, or contact help@m2reporter.com.

Step 4-4: Initialize Database Schema

Click Initialize Schema. This creates 16 database tables, sets up foreign key relationships, creates indexes for performance, creates a default company record, and stores validated license information in the license_info table (if Step 3 was completed).

Initialize Schema

Creates all database tables, indexes, and the default company record.

Initialize Schema

Step 4-5: Generate Client Configuration

Click Generate Client Config and save the file as M2R_client_config.json. Save it to the server at C:\Program Files\M2R Server\client_configs\ and share it with all users via email, network share, or USB.

Generate Client Config

Save the client connection file to distribute to workstations.

Generate Client Config

The client config file contains the server host IP, port, database name, and username. The password is NOT included for security reasons.

Step 4-6: Configuration Complete

The Status Log will display confirmation that the database was created successfully, the license was validated, the schema was initialized, and the client config was generated. Click Finish.

Post-Installation Verification

Verify PostgreSQL Service

1. Press Win + R, type services.msc, press Enter
2. Locate service: postgresql-x64-16
3. Verify Status is Running and Startup Type is Automatic

Verify File Installation

Check that these files exist:

- C:\Program Files\M2R Server\M2R_Server_Setup.exe
- C:\Program Files\M2R Server\README_SERVER.txt
- C:\Program Files\M2R Server\License.txt
- C:\Program Files\PostgreSQL\16\bin\psql.exe

5. Network Setup

To allow client connections, you must configure the server firewall and PostgreSQL network settings.

Windows Firewall Configuration

Method 5-1: Automatic Rule Creation (Recommended)

Open PowerShell as Administrator and run:

```
New-NetFirewallRule -DisplayName "M2R PostgreSQL" `
-Direction Inbound `
-Protocol TCP `
-LocalPort 5432 `
-Action Allow `
-Profile Domain,Private
```

Method 5-2: Manual Rule Creation

1. Open Windows Firewall with Advanced Security (Win + R, type wf.msc)
2. Click Inbound Rules → New Rule
3. Rule Type: Port → Next
4. Protocol: TCP, Specific local ports: 5432 → Next
5. Action: Allow the connection → Next
6. Profile: Check Domain and Private (uncheck Public for security) → Next
7. Name: M2Reporter PostgreSQL → Finish

PostgreSQL Network Configuration

Allow Network Connections

PostgreSQL must be configured to accept connections from client machines.

The server installer automatically configures `pg_hba.conf` to allow client connections using `scram-sha-256` authentication. No manual configuration is required.

To verify or modify this, open the file `C:\Program Files\PostgreSQL\16\data\pg_hba.conf` and add this line before the IPv4 local connections section:

```
host      M2Reprter      M2Reporter      0.0.0.0/0      md5
```

This allows the M2R user to connect from any IP address using password authentication.

Note

Better Security: Restrict connections to your network subnet instead: `host M2Reporter M2Reporter 192.168.1.0/24 md5`

Save the file and restart the PostgreSQL service:

```
Restart-Service postgresql-x64-16
```

Verify PostgreSQL is Listening

Open Command Prompt and run:

```
netstat -an | findstr :5432
```

You should see `TCP 0.0.0.0:5432` in a `LISTENING` state.

Remote Access Configuration

VPN (Recommended): Set up a VPN server (Windows VPN, OpenVPN, etc.). Clients connect via VPN, then access the server on the VPN IP. This is the most secure option.

Port Forwarding (Not Recommended): Forward an external port to the server's local IP:5432. Use strong passwords, consider changing the PostgreSQL port, and use dynamic DNS if your public IP changes. This method has significant security risks.

Test Connection from Client

From a client workstation, open Command Prompt and test network connectivity:

```
ping <server-ip-address>
telnet <server-ip-address> 5432
```

If telnet connects, the port is accessible. If the connection is refused, check the firewall configuration.

6. License Management

License Types

License	Concurrent Users	Description
Standard	10 users	Small teams and offices
Professional	25 users	Mid-size organizations
Enterprise	Unlimited	Large organizations

License States

State	Days Past Expiry	Client Behavior
Valid	N/A	Full access to all features
Grace Period	0–30 days	Full access + expiry warning banner
Read-Only	30–60 days	Can view and export, cannot edit
Expired	60+ days	No access until license renewed

Viewing License Status

Run the M2R Server Setup tool and click Check License Status. This displays the license key (M2R-XXXX-XXXX-XXXX-XXXX), status, type, max users, expiration date, and days remaining.

Updating License

1. Obtain new license key from M2R sales
2. Run M2R Server Setup tool
3. Click Update License button
4. Enter new license key
5. Click Validate

The new license is effective immediately for all connected clients.

License Renewal Timeline

30 days before expiration: Clients see warning “License expires in 30 days.” Admin receives email reminder (if configured).

After expiration: Clients enter grace period. Can continue working normally but a warning banner is displayed on all screens.

30 days after expiration: Clients enter read-only mode. Can view data, generate reports, and export, but cannot create, edit, or delete records.

60 days after expiration: Clients fully locked out. Must contact admin to renew license.

License Troubleshooting

“License validation failed”: Cause: No internet connectivity on server. Solution: Connect server to internet temporarily and validate license.

“License expired” but renewal not recognized: Cause: Old license cached on clients. Solution: Have clients restart the application; the new license will load.

“Maximum users exceeded”: Cause: More than the allowed number of concurrent connections. Solution: Upgrade license or disconnect idle users.

7. User Management

PostgreSQL User Management

User accounts are managed at the PostgreSQL level. M2R Network Edition uses PostgreSQL authentication. Each client user logs in with a shared username (M2R) and the database password you set during configuration.

For audit logging, clients are identified by their client machine name and the user's Windows username, which is stored in the audit_log table.

Create Read-Only Users (Optional)

1. Open pgAdmin 4 (from Start Menu)
2. Connect to PostgreSQL server (localhost)
3. Right-click Login/Group Roles → Create → Login/Group Role
4. General tab: Name: M2R_readonly
5. Definition tab: Password: (choose strong password)
6. Privileges tab: Can login: Yes, all others: No
7. Click Save

Then grant read-only access to the database:

```
GRANT CONNECT ON DATABASE M2R TO M2R_readonly;
GRANT USAGE ON SCHEMA public TO M2R_readonly;
GRANT SELECT ON ALL TABLES IN SCHEMA public TO M2R_readonly;
```

Change Database Password

1. Open pgAdmin 4
2. Navigate to Login/Group Roles → M2R
3. Right-click → Properties
4. Definition tab → Enter new password
5. Click Save
6. Update all client config files with the new password
7. Notify all users of the password change

View Active Connections

Open pgAdmin 4 and navigate to Databases → M2R, then click the Dashboard tab. The Sessions section shows each user's name, application name, client IP address, connection start time, and current activity.

Terminate User Connection

If a user connection is hung or needs to be disconnected, find the session in pgAdmin's Dashboard → Sessions list, right-click, and select Terminate. Alternatively, use this SQL command:

```
SELECT pg_terminate_backend(pid)
FROM pg_stat_activity
WHERE datname = 'M2R'
AND pid <> pg_backend_pid();
```

8. Backup & Recovery

Critical

Regular backups are essential to prevent data loss. Follow the recommended backup strategy below.

Backup Strategy

Frequency	Type	Retention
Daily	Full database backup (automated)	Keep 30 days
Weekly	Backup verification test	Keep 90 days
Monthly	Offsite backup copy	Keep 1 year

Manual Backup

Using pg_dump (Command Line)

1. Open Command Prompt as Administrator
2. Navigate to: `cd "C:\Program Files\PostgreSQL\16\bin"`
3. Run the backup command (see below)
4. Enter database password when prompted

```
pg_dump -U M2R -F c -f "C:\Backups\M2R_YYYYMMDD.dump" M2R
```

Using pgAdmin 4 (GUI)

1. Open pgAdmin 4
2. Navigate to Databases → M2R
3. Right-click → Backup
4. Filename: C:\Backups\M2R_YYYYMMDD.dump
5. Format: Custom, Encoding: UTF8
6. Click Backup

Automated Backups

Create a backup script at C:\Scripts\backup_M2R.bat with the pg_dump command and a forfiles command to delete backups older than 30 days. Then configure Windows Task Scheduler to run this script daily at 2:00 AM with highest privileges, running whether the user is logged on or not.

Restore from Backup

Full Database Restore

Important

This will overwrite all current data! Disconnect all clients before proceeding.

1. Stop M2R Server Setup tool (if running)
2. Disconnect all clients
3. Open Command Prompt as Administrator
4. Navigate to PostgreSQL bin directory
5. Drop existing database: `psql -U postgres -c "DROP DATABASE M2R;"`
6. Recreate database: `psql -U postgres -c "CREATE DATABASE M2R OWNER M2R;"`
7. Restore from backup: `pg_restore -U M2R -d M2R -c "C:\Backups\backup_file.dump"`
8. Verify restoration by checking record counts

Backup Verification

Perform a monthly verification test by restoring the backup to a test database (M2R_test), running verification queries to check record counts and recent data, and then dropping the test database after successful verification.

Disaster Recovery

If the server hard drive fails: install a new server or repair the existing one, install the Windows Server OS, run the M2R Server installer, complete the server setup wizard, restore from the most recent backup, verify data integrity, and reconnect clients.

Recovery Time Objective (RTO): 2–4 hours

Recovery Point Objective (RPO): 24 hours (with daily backups)

9. Monitoring & Maintenance

Daily Monitoring

Check PostgreSQL Service Status

```
Get-Service postgresql-x64-16
```

Should show Running.

Check Disk Space

```
Get-PSDrive C | Select-Object Used,Free
```

Alert if free space is less than 10 GB.

Check Recent Backup

```
Get-ChildItem C:\Backups\*.dump |  
  Sort-Object LastWriteTime -Descending |  
  Select-Object -First 1
```

Verify backup was created within the last 24 hours.

Weekly Monitoring

Review PostgreSQL Logs

Location: C:\Program Files\PostgreSQL\16\data\log\

Check for connection failures, authentication errors, slow queries, and database errors. Open the most recent log file and search for ERROR or FATAL.

Database Size Monitoring

```
SELECT pg_size_pretty(pg_database_size('M2R')) AS database_size;
```

Track growth over time. Alert if the growth rate is unexpected.

Connection Pool Status

```
SELECT count(*) AS active_connections  
FROM pg_stat_activity  
WHERE datname = 'M2R';
```

Normal: 0–10 connections. Warning: 10–15 connections (approaching limit). Alert: 15+ connections (may impact performance).

Monthly Maintenance

Vacuum Database

Run VACUUM ANALYZE during off-hours to reclaim storage and update statistics. Alternatively, enable autovacuum in postgresql.conf (recommended).

Reindex Database

Run REINDEX DATABASE M2R during a maintenance window (may take 5–30 minutes). This rebuilds indexes for optimal performance.

Review Audit Log

Query the audit_log table for the past 30 days, grouped by action. Look for unusual patterns such as excessive deletes, failed login attempts, or off-hours activity.

Quarterly Maintenance

- **Check for PostgreSQL Updates:** Visit postgresql.org/download, review release notes, backup database before upgrading, and test thoroughly after updating.
- **Review License Expiration:** Query the license_info table. Renew license if expiry is less than 60 days.
- **Audit User Access:** Query pg_stat_activity for distinct client addresses and verify all IP addresses are authorized.
- **Test Backup Restoration:** Perform a full backup verification test (see Backup Verification section).

10. Troubleshooting

Cannot Connect to Server

Symptom: Server Setup tool shows "Connection failed."

1. Is PostgreSQL running? Run `Get-Service postgresql-x64-16` in PowerShell. If stopped, start it with `Start-Service postgresql-x64-16`.
2. Is PostgreSQL listening on port 5432? Run `netstat -an | findstr :5432`. If not LISTENING, check postgresql.conf for `listen_addresses = '*'` and `port = 5432`.
3. Correct password? Try resetting with `psql -U postgres` followed by `\password postgres`.
4. Firewall blocking? Temporarily disable Windows Firewall and test.

Clients Cannot Connect

Symptom: Client shows "Connection to server failed."

1. Can client ping server? Run `ping <server-ip>`. If it fails, check cables, switches, and router.
2. Is port 5432 accessible? Run `telnet <server-ip> 5432`. If connection refused, check server firewall rules.
3. Correct credentials? Verify server IP, database name, username, and password in client config.
4. pg_hba.conf configured? Server must have: `host M2R M2R 0.0.0.0/0 md5`.
5. Test with psql from client: `psql -h <server-ip> -U M2R -d M2R`. If it connects, the network is OK.

Record Locked Errors

Symptom: User gets "Record is locked by another user."

Causes: Another user is editing the same record, a previous edit session didn't release its lock (crashed), or the lock timeout (30 minutes default) hasn't expired.

Solutions: Wait 30 minutes and retry. Check who has the lock by querying the record_locks table. Force release the lock (admin only) by deleting from record_locks. Release all expired locks with:

```
DELETE FROM record_locks WHERE locked_at < NOW() - INTERVAL '30 minutes';
```

Poor Performance

Symptom: Slow searches, timeouts, laggy UI.

1. Check server resources in Task Manager: CPU should be under 80%, RAM should have 2+ GB free, disk I/O should not be at 100%.
2. Check database size: if over 10 GB, consider archiving old data.
3. Check active connections: if over 20, performance may degrade.
4. Run VACUUM ANALYZE to reclaim storage and update statistics.
5. Check for long-running queries and cancel them with pg_cancel_backend(<pid>).

Database Corruption

Symptom: Errors like "invalid page header" or "index corrupted."

1. Restore from backup immediately (preferred solution).
2. If no backup, try reindexing: REINDEX DATABASE M2R;
3. Check disk for errors: chkdsk C: /F /R
4. Check PostgreSQL logs for hardware errors.

License Validation Fails

Symptom: "License validation failed" during setup.

1. Check internet connectivity: ping google.com
2. Firewall blocking HTTPS? Ensure port 443 outbound is allowed.
3. License key correct? Copy/paste from email to avoid typos. Ensure the format is M2R-XXXX-XXXX-XXXX-XXXX.
4. License expired? Check expiration date with M2R support.
5. Contact support: support@M2R.com. Provide license key, error message, and server internet status.

Import Failures

Symptom: Migration tool or import fails.

1. Source database accessible? Verify file path is correct.
2. PostgreSQL has disk space? Large imports need 2x data size in free space.
3. Timeout errors? Increase timeout in import tool settings.
4. Foreign key violations? Check logs for specific errors; may indicate data corruption.

11. Security Best Practices

Password Security

- **Strong Passwords:** Minimum 20 characters, mix uppercase, lowercase, numbers, and symbols.
- **Change Default Passwords:** Change the PostgreSQL "postgres" superuser password and the "M2R" database user password immediately after setup.
- **Password Storage:** Never store passwords in plain text. Use password managers (1Password, LastPass) and encrypt client config files.

Network Security

- **Use VPN for Remote Access:** Never expose PostgreSQL directly to the internet. Set up Windows VPN or third-party VPN (OpenVPN). Require VPN for all remote clients.
- **Firewall Rules:** Only allow port 5432 from trusted IPs. Block public access (use Private/Domain profiles only). Restrict to a specific subnet.
- **Change Default Port (Optional):** Edit postgresql.conf to use a non-standard port (e.g., 54320). Update firewall rules and client configs accordingly.

PostgreSQL Security

- **Restrict pg_hba.conf:** Don't use 0.0.0.0/0 (all IPs) in production. Use your specific subnet (e.g., 192.168.1.0/24).
- **Enable SSL/TLS (Advanced):** Encrypts data in transit. Requires an SSL certificate. See PostgreSQL SSL documentation for setup.
- **Regular Updates:** Apply PostgreSQL security patches, Windows security updates, and M2R application updates promptly.

Physical Security

- **Server Room Access:** Lock the server room, log physical access, and install security cameras (recommended).
- **Backup Security:** Store backups on a separate drive, use offsite backup storage, and encrypt backup files with BitLocker.
- **Server Console:** Require login to access the server, set automatic screen lock to 5 minutes, and disable unnecessary accounts.

Audit & Compliance

- **Enable Audit Logging:** All user actions are logged automatically. Review logs monthly and retain per compliance requirements.
- **Access Reviews:** Conduct quarterly reviews of authorized users. Disable accounts for terminated employees and document all access changes.
- **Data Retention:** Define a retention policy (e.g., 7 years). Archive old data and securely delete when no longer needed.

Compliance

- **GDPR:** Right to erasure — ensure you can delete consumer data upon request.
- **CCPA:** Data access requests — ensure you can provide consumers with their data.
- **FCRA:** Credit reporting compliance — maintain accuracy and integrity of all reported data.

12. Performance Tuning

PostgreSQL Configuration

Edit C:\Program Files\PostgreSQL\16\data\postgresql.conf to optimize performance:

Memory Settings (for server with 16 GB RAM)

Setting	Recommended Value
shared_buffers	4GB (25% of total RAM)
effective_cache_size	12GB (75% of total RAM)
work_mem	64MB (per-operation memory)
maintenance_work_mem	512MB (for VACUUM, index creation)

Connection Settings

max_connections: 50 (default 100; reduce if not needed)

Write-Ahead Log (WAL)

wal_buffers: 16MB

checkpoint_completion_target: 0.9

Query Planner (SSD Optimization)

random_page_cost: 1.1 (default 4.0 for HDD)

effective_io_concurrency: 200 (default 1 for HDD)

 **Important:** After Changes: Restart the PostgreSQL service: Restart-Service postgresql-x64-16

Hardware Optimization

- **Use SSD Storage:** 10x faster than HDD for database operations. Reduces query latency and improves backup/restore speed.
- **Add More RAM:** PostgreSQL heavily relies on RAM for caching. Minimum 16 GB for 10+ users; 32 GB recommended for 20+ users.
- **Upgrade Network:** Gigabit Ethernet minimum. Avoid Wi-Fi for servers. Use quality network switches.

Application Optimization

Connection Pooling: Network clients use connection pooling automatically (10 max connections per client).

Batch Operations: For bulk imports, use database transactions. This is 10x faster than individual inserts.

Appendix

PostgreSQL Command Reference

Command	Description
psql -U M2R -d M2R	Connect to database
\l	List databases
\dt	List tables
\d table_name	Describe table
\q	Quit psql

Useful SQL Queries

Total account count:

```
SELECT COUNT(*) FROM base_segments WHERE company_id = 1;
```

Accounts by status:

```
SELECT account_status, COUNT(*) FROM base_segments GROUP BY account_status ORDER BY count DESC;
```

Recent changes:

```
SELECT * FROM audit_log ORDER BY timestamp DESC LIMIT 100;
```

Firewall Port Reference

Port	Protocol	Purpose
5432	TCP	PostgreSQL database connections
443	TCP	HTTPS (license validation)
80	TCP	HTTP (optional, for updates)

File Locations

Item	Location
PostgreSQL Config	C:\Program Files\PostgreSQL\16\data\postgresql.conf
Client Auth Config	C:\Program Files\PostgreSQL\16\data\pg_hba.conf
PostgreSQL Logs	C:\Program Files\PostgreSQL\16\data\log\
Backup Directory	C:\Backups\ (create if not exists)
Client Configs	C:\Program Files\M2R Server\client_configs\

Support Resources

- **Email:** help@m2reporter.com
- **Website:** www.m2reporter.com
- **Documentation:** [Help Center | M2 Reporter Metro 2® Credit Reporting Software](#)
- **PostgreSQL Docs:** <https://www.postgresql.org/docs/16/>

Document Version: 5.0.1 | Last Updated: January 2026
© 2026 M2R. All rights reserved.