

# Odoo AWS Troubleshooting Guide

Quick solutions to common problems

---

## Emergency Quick Fixes

### Server Not Responding?

```
# Restart everything
ssh -i MyTestApp-KeyPair.pem ec2-user@56.228.2.47
sudo systemctl restart postgresql
sudo systemctl restart odoo
```

### Odoo Won't Start?

```
# Check what's wrong
sudo journalctl -u odoo -n 50

# Common fix: Clear lock file
sudo rm -f /home/ec2-user/odoo18/.odoo.pid

# Try starting again
sudo systemctl start odoo
```

### Can't Connect to Server?

1. Check AWS Security Group allows your IP
  2. Verify SSH key permissions
  3. Try alternate connection: `ssh -i key.pem ec2-user@56.228.2.47`
- 

## Diagnostic Commands

### Check Everything is Running

```
# Odoo status
sudo systemctl status odoo

# PostgreSQL status
sudo systemctl status postgresql

# Check Odoo process
ps aux | grep odoo

# Check port 8069 is listening
sudo netstat -tlnp | grep 8069
```

## View Logs

```
# Last 100 lines of Odoo logs
tail -n 100 /home/ec2-user/.odoo/odoo-server.log

# Follow logs in real-time
tail -f /home/ec2-user/.odoo/odoo-server.log

# System logs for Odoo service
sudo journalctl -u odoo -n 100

# PostgreSQL logs
sudo tail /var/lib/pgsql/data/log/postgresql-*.log
```

## Check System Resources

```
# CPU and memory usage
top

# Press 'q' to exit

# Memory details
free -h

# Disk space
df -h

# Who's logged in
who
```

---

## ✖ Common Problems & Solutions

### 1. "Connection Refused" on Port 8069

#### Symptoms:

- Browser shows "Connection refused"
- Can't access <http://56.228.2.47:8069>

#### Diagnosis:

```
# Check if Odoo is running
sudo systemctl status odoo

# Check if port 8069 is open
sudo netstat -tlnp | grep 8069
```

#### Solutions:

##### A. Odoo not running:

```
sudo systemctl start odoo
sudo systemctl status odoo
```

#### B. Port in use by another process:

```
# Find process using port 8069
sudo lsof -i :8069

# Kill that process
sudo kill -9 PROCESS_ID

# Start Odoo
sudo systemctl start odoo
```

#### C. AWS Security Group:

- Go to AWS Console → EC2 → Security Groups
  - Find "launch-wizard-1"
  - Verify port 8069 is open for your IP
- 

## 2. "Permission Denied" SSH Connection

#### Symptoms:

- Can't connect via SSH
- "Permission denied (publickey)" error

#### Solutions:

##### A. Key file permissions (Windows):

1. Right-click `MyTestApp-KeyPair.pem`
2. Properties → Security → Advanced
3. Disable inheritance
4. Remove all users except your account
5. Give yourself Full Control

##### B. Wrong key path:

```
# Verify file exists
dir "C:\path\to\MyTestApp-KeyPair.pem"

# Use full path in SSH command
ssh -i "C:\full\path\to\MyTestApp-KeyPair.pem" ec2-user@56.228.2.47
```

##### C. Security Group doesn't allow your IP:

- AWS Console → EC2 → Security Groups
  - Edit "launch-wizard-1"
  - Add your IP to port 22 (SSH)
-

### 3. Odoo Shows "500 Internal Server Error"

#### Symptoms:

- Browser shows generic error page
- Odoo loaded but crashed on request

#### Diagnosis:

```
# Check recent errors in log  
tail -n 100 /home/ec2-user/.odoo/odoo-server.log | grep -i error
```

#### Common Causes:

##### A. Database not initialized:

```
# Stop Odoo  
sudo systemctl stop odoo  
  
# Initialize database  
cd /home/ec2-user/odoo18  
/usr/bin/python3.11 odoo-bin -c /home/ec2-user/.odoo/odoo.conf -d odoo18 -i base --stop-after-init  
  
# Start Odoo  
sudo systemctl start odoo
```

##### B. Python error in code:

```
# Check logs for traceback  
tail -n 200 /home/ec2-user/.odoo/odoo-server.log  
  
# Fix the Python error in code  
# Then restart  
sudo systemctl restart odoo
```

##### C. Missing Python dependency:

```
# Install missing package (example)  
sudo pip3.11 install package-name  
  
# Restart Odoo  
sudo systemctl restart odoo
```

---

### 4. "Module Not Found" Error

#### Symptoms:

- Odoo logs show "ModuleNotFoundError"
- Custom module won't load

## Solutions:

### A. Module path not correct:

```
# Verify module exists
ls -la /home/ec2-user/odoo18/addons/module_name

# Check addons path in config
cat /home/ec2-user/.odoo/odoo.conf | grep addons_path
```

### B. Module not properly structured:

```
# Module must have __manifest__.py
ls -la /home/ec2-user/odoo18/addons/module_name/__manifest__.py

# Check manifest syntax
python3.11 -c "import ast; ast.parse(open('/home/ec2-user/odoo18/addons/module_name/__manifest__.py').read())"
```

### C. Module needs upgrade:

- Go to Odoo UI → Apps
- Search for module
- Click "Upgrade" button

---

## 5. Git Push Failed - Authentication

### Symptoms:

- `git push` fails with authentication error
- "Authentication failed for GitHub"

### Solutions:

#### A. Token expired:

```
# Generate new token at: https://github.com/settings/tokens
# Update remote URL with new token
cd /home/ec2-user/odoo18
git remote set-url origin https://NEW_TOKEN@github.com/karem505/odoo.git

# Test
git push origin development
```

#### B. Wrong remote URL:

```
# Check current remote
git remote -v

# Should show token in URL
```

```
# If not, set it:
git remote set-url origin https://TOKEN@github.com/karem505/odoo.git
```

---

## 6. Database Connection Error

### Symptoms:

- Odoo logs show "psycopg2.OperationalError"
- Can't connect to database

### Diagnosis:

```
# Check PostgreSQL is running
sudo systemctl status postgresql

# Try connecting manually
psql -U ec2-user -d odoo18 -c "SELECT version();"

```

### Solutions:

#### A. PostgreSQL not running:

```
sudo systemctl start postgresql
sudo systemctl restart odoo

```

#### B. Authentication issue:

```
# Check pg_hba.conf
sudo cat /var/lib/pgsql/data/pg_hba.conf

# Should have lines with "trust" for local connections
# If not, fix it:
sudo sed -i 's/ident/trust/g' /var/lib/pgsql/data/pg_hba.conf

# Restart PostgreSQL
sudo systemctl restart postgresql
sudo systemctl restart odoo

```

#### C. Database doesn't exist:

```
# List databases
sudo -u postgres psql -c "\l"

# Create if missing
sudo -u postgres createdb odoo18 -O ec2-user

```

---

## 7. Merge Conflicts in Git

### Symptoms:

- `git pull` fails with "CONFLICT"
- Can't merge branches

**Solution:**

```
# Pull with conflicts
git pull origin development

# Git will mark conflict files
# Open conflicting files in VS Code

# Look for conflict markers:
# <<<<<< HEAD
# Your changes
# =====
# Their changes
# >>>>>> branch-name

# Edit file to resolve
# Remove markers, keep desired code

# Mark as resolved
git add filename

# Complete merge
git commit -m "fix: resolve merge conflicts"

# Push
git push origin branch-name
```

**Alternative - Accept theirs:**

```
git pull origin development --strategy-option theirs
```

**Alternative - Start over:**

```
# Abort merge
git merge --abort

# Reset to remote
git fetch origin
git reset --hard origin/development
```

---

## 8. VS Code Can't Connect via Remote SSH

**Symptoms:**

- VS Code Remote SSH fails
- "Could not establish connection"

**Solutions:**

#### A. SSH works from terminal but not VS Code:

1. Close VS Code completely
2. Delete: `C:\Users\USERNAME\.ssh\known_hosts`
3. Restart VS Code
4. Try connecting again

**B. Fix SSH config:** Press `Ctrl+Shift+P` → "Remote-SSH: Open SSH Configuration File"

Verify config:

```
Host odoo-aws
  HostName ec2-56-228-2-47.eu-north-1.compute.amazonaws.com
  User ec2-user
  IdentityFile C:\path\to\MyTestApp-KeyPair.pem
  StrictHostKeyChecking no
```

#### C. Reset Remote SSH:

```
Ctrl+Shift+P → "Remote-SSH: Kill VS Code Server on Host"
```

Then reconnect.

---

## 9. Out of Disk Space

#### Symptoms:

- "No space left on device"
- Odoo won't start
- Can't write files

#### Diagnosis:

```
# Check disk usage
df -h

# Find large directories
du -sh /home/ec2-user/* | sort -h

# Find large files
find /home/ec2-user -type f -size +100M -exec ls -lh {} \;
```

#### Solutions:

##### A. Clean log files:

```
# Check log size
ls -lh /home/ec2-user/.odoo/odoo-server.log

# Truncate log (keep last 1000 lines)
tail -n 1000 /home/ec2-user/.odoo/odoo-server.log > /tmp/odoo.log
sudo mv /tmp/odoo.log /home/ec2-user/.odoo/odoo-server.log
```



```
# Or rotate logs
sudo logrotate -f /etc/logrotate.conf
```

#### B. Clean old backups:

```
# List backups
ls -lh /tmp/odoo18_backup_*

# Remove old backups (keep last 5)
ls -t /tmp/odoo18_backup_* | tail -n +6 | xargs rm -f
```

#### C. Clean Docker/temp files (if any):

```
# Clean yum cache
sudo yum clean all

# Clean tmp
sudo rm -rf /tmp/*
```

---

## 10. Module Update Failed

#### Symptoms:

- Module won't update
- Changes don't appear
- Odoo shows old version

#### Solutions:

##### A. Update via command line:

```
cd /home/ec2-user/odoo18
sudo systemctl stop odoo

# Update specific module
/usr/bin/python3.11 odoo-bin -c /home/ec2-user/.odoo/odoo.conf -d odoo18 -u module_name --stop-after-init

sudo systemctl start odoo
```

##### B. Update via UI:

1. Odoo → Apps → Remove "Apps" filter
2. Search module name
3. Click "Upgrade"

##### C. Force full update:

```
cd /home/ec2-user/odoo18
sudo systemctl stop odoo
```

```
# Update all modules
/usr/bin/python3.11 odoo-bin -c /home/ec2-user/.odoo/odoo.conf -d odoo18 -u all --stop-after-init

sudo systemctl start odoo
```

## Advanced Debugging

### Enable Debug Mode in Odoo

**Method 1: URL Add** `?debug=1` to URL:

```
http://56.228.2.47:8069/web?debug=1
```

**Method 2: Activate Developer Mode** Settings → Activate Developer Mode

#### Debug Features:

- View technical information
- Edit views
- See Python code
- Access developer menu

### Verbose Logging

Edit config for more logs:

```
nano /home/ec2-user/.odoo/odoo.conf

# Add these lines:
log_level = debug
log_db = True
log_db_level = debug

# Restart
sudo systemctl restart odoo
```

### Python Debugger (pdb)

Add to Python code:

```
import pdb; pdb.set_trace()
```

Then run Odoo in foreground:

```
sudo systemctl stop odoo
cd /home/ec2-user/odoo18
/usr/bin/python3.11 odoo-bin -c /home/ec2-user/.odoo/odoo.conf
```

### Database Query Analysis

```
# Connect to database
psql -U ec2-user -d odoo18

# List tables
\dt

# Check table structure
\d tablename

# View slow queries
SELECT * FROM pg_stat_statements ORDER BY total_time DESC LIMIT 10;

# Exit
\q
```

---

## Performance Issues

### Odoo is Slow

#### Check System Resources:

```
top
free -m
df -h
```

#### Check Database:

```
# Database size
sudo -u postgres psql odoo18 -c "SELECT pg_size_pretty(pg_database_size('odoo18'));"

# Vacuum database
sudo -u postgres psql odoo18 -c "VACUUM ANALYZE;"
```

#### Optimize:

```
# Restart services
sudo systemctl restart postgresql
sudo systemctl restart odoo

# Clear browser cache
# Check network speed
ping 56.228.2.47
```

---

## Critical Issues

### Complete System Recovery

If everything is broken:

```
# 1. Stop services
sudo systemctl stop odoo
sudo systemctl stop postgresql

# 2. Check system resources
df -h # Disk space
free -m # Memory
top # CPU

# 3. Check logs
journalctl -xe

# 4. Restart PostgreSQL
sudo systemctl start postgresql
sudo systemctl status postgresql

# 5. Start Odoo
sudo systemctl start odoo

# 6. Monitor logs
tail -f /home/ec2-user/.odoo/odoo-server.log
```

## Database Recovery

If database is corrupted:

```
# Stop Odoo
sudo systemctl stop odoo

# Backup current database
sudo -u postgres pg_dump odoo18 > /tmp/odoo18_broken_$(date +%Y%m%d).sql

# Try repair
sudo -u postgres psql odoo18 -c "REINDEX DATABASE odoo18;"
sudo -u postgres psql odoo18 -c "VACUUM FULL;"

# If needed, restore from backup
# sudo -u postgres dropdb odoo18
# sudo -u postgres createdb odoo18 -O ec2-user
# sudo -u postgres psql odoo18 < /tmp/odoo18_backup_YYYYMMDD_HHMMSS.sql

# Start Odoo
sudo systemctl start odoo
```

---

## When to Ask for Help

Ask for help if:

- Issue persists after trying solutions
- Data loss risk

- Security concern
- Unfamiliar with solution steps
- Need production rollback

**Contact:**

- Team Lead: @karem505
- Check documentation first
- Prepare error messages and logs



## Reporting Issues

When reporting issues, include:

```
**Problem**: Brief description
**When**: Date/time it started
**Impact**: Users affected / features broken
**What I tried**: Steps you already took
**Error messages**: Copy from logs
**Screenshots**: If helpful

**System Info**:
- Odoo version: 18.0
- Recent changes: Deployment, config change, etc.
- Logs: (attach relevant log snippet)
```



## Prevention Checklist

Prevent issues by:

- ☐ Regular backups (daily)
- ☐ Monitor disk space (weekly)
- ☐ Review logs (daily)
- ☐ Test before deploying
- ☐ Follow deployment checklist
- ☐ Keep documentation updated
- ☐ Rotate credentials (quarterly)
- ☐ Update system packages (monthly)



## Learning Resources

- **Odoo Forum**: <https://www.odoo.com/forum>
- **PostgreSQL Docs**: <https://www.postgresql.org/docs/>
- **Git Documentation**: <https://git-scm.com/doc>
- **AWS EC2 Troubleshooting**: <https://docs.aws.amazon.com/AWSEC2/>

---

**Remember:** Most issues can be solved by:

1. Checking logs
2. Restarting services

3. Verifying configuration

4. Testing step by step

**Stay calm, debug systematically, and document what you learn!**

---

**Last Updated:** 2025-10-22 **Maintained By:** @karem505