



Prompt Machine - Troubleshooting Guide

Common Issues & Solutions

1. Database Connection Failed

```
bash
```

```
# Error: "ECONNREFUSED" or "password authentication failed"
```

```
# Check 1: Test connection manually
```

```
psql -h sql.prompt-machine.com -U promptmachine_userbeta -d promptmachine_dbbeta
```

```
# Check 2: Verify .env values have no extra spaces
```

```
cat ~/prompt-machine/config/.env | grep DB_
```

```
# Check 3: Ensure PostgreSQL allows remote connections
```

```
# May need to whitelist your server IP with hosting provider
```

2. JWT Token Errors

```
bash
```

```
# Error: "JsonWebTokenError: invalid signature"
```

```
# Solution: Ensure JWT_SECRET matches across restarts
```

```
# Never change JWT_SECRET after users have logged in
```

```
# If you must change it, clear all sessions:
```

```
psql -c "DELETE FROM sessions;" -h sql.prompt-machine.com -U promptmachine_userbeta -d promptmachine_dbbeta
```

3. Port Already in Use

```
bash
```

```
# Error: "EADDRINUSE: address already in use :::3001"
```

```
# Find process using the port
```

```
sudo lsof -i :3001
```

```
# Kill the process
```

```
kill -9 <PID>
```

```
# Or use PM2 to manage properly
```

```
pm2 stop all
```

```
pm2 start ecosystem.config.js
```

4. Nginx 502 Bad Gateway

```
bash
```

```
# This means Nginx can't reach your Node.js app
```

```
# Check 1: Is the app running?
```

```
pm2 status
```

```
# Check 2: Correct port in Nginx config?
```

```
cat /etc/nginx/sites-available/api.prompt-machine.com
```

```
# Check 3: Test Nginx config
```

```
sudo nginx -t
```

```
# Fix: Restart everything
```

```
pm2 restart all
```

```
sudo systemctl reload nginx
```

5. SSL Certificate Issues

```
bash
```

```
# Error: "Challenge failed for domain"
```

```
# For wildcard certificates, you need DNS validation
```

```
# Easier solution: Generate certificates on-demand per subdomain
```

```
# Test with staging first
```

```
sudo certbot certonly --staging --webroot -w /var/www/html -d test.tool.prompt-machine.com
```

```
# If that works, remove --staging
```

```
sudo certbot certonly --webroot -w /var/www/html -d test.tool.prompt-machine.com
```

6. Claude API Errors

```
javascript
```

```
// Error: "Rate limit exceeded"
```

```
// Solution: Implement exponential backoff
```

```
const callClaudeWithRetry = async (prompt, retries = 3) => {  
  for (let i = 0; i < retries; i++) {  
    try {  
      return await claude.complete(prompt);  
    } catch (error) {  
      if (error.status === 429 && i < retries - 1) {  
        await new Promise(resolve => setTimeout(resolve, Math.pow(2, i) * 1000));  
      } else {  
        throw error;  
      }  
    }  
  }  
};
```

7. Subdomain Not Working

```
bash
```

Check 1: DNS propagation

`dig test.tool.prompt-machine.com`

Check 2: Nginx wildcard server block exists

`ls -la /etc/nginx/sites-enabled/`

Check 3: Your app handles subdomain routing

Add logging to your subdomain middleware

8. Memory/Performance Issues

bash

Monitor memory usage

`pm2 monit`

Check for memory leaks

Add to ecosystem.config.js:

`max_memory_restart: '500M'`

Enable cluster mode for better performance

`instances: 'max' # or number of CPUs`

`exec_mode: 'cluster'`

9. CORS Errors

javascript

// Error: "blocked by CORS policy"

// Fix: Ensure your CORS config includes all domains

`const corsOptions = {`

`origin: function (origin, callback) {`

// Log the origin to debug

`console.log('CORS request from:', origin);`

// Add your logic here

`callback(null, true);`

`}`

`};`

10. Database Migration Failed

```
bash
```

```
# Always backup first!
```

```
pg_dump -h sql.prompt-machine.com -U promptmachine_userbeta promptmachine_dbbeta > backup.sql
```

```
# Check current schema
```

```
psql -h sql.prompt-machine.com -U promptmachine_userbeta -d promptmachine_dbbeta -c "\dt"
```

```
# Run migrations one at a time
```

```
psql -h sql.prompt-machine.com -U promptmachine_userbeta -d promptmachine_dbbeta -f single-migration.sql
```

Debug Mode

Add this to your .env for better error messages:

```
bash
```

```
NODE_ENV=development
```

```
DEBUG=express:*
```

```
LOG_LEVEL=debug
```

Health Check Script

Create `scripts/health-check.sh`:

```
bash
```

```
#!/bin/bash
```

```
echo "=== Prompt Machine Health Check ==="
```

```
# Check API
```

```
curl -f http://localhost:3001/health || echo "❌ API is down"
```

```
# Check database
```

```
psql -h sql.prompt-machine.com -U promptmachine_userbeta -d promptmachine_dbbeta -c "SELECT 1" || echo "❌ Database connection failed"
```

```
# Check Nginx
```

```
sudo nginx -t || echo "❌ Nginx config error"
```

```
# Check PM2
```

```
pm2 status || echo "❌ PM2 issues"
```

```
# Check disk space
```

```
df -h | grep -E '^/dev/' || echo "❌ Check disk space"
```

Emergency Recovery

If everything breaks:

```
bash
```

```
# 1. Stop everything
```

```
pm2 stop all
```

```
sudo systemctl stop nginx
```

```
# 2. Check logs
```

```
tail -f ~/prompt-machine/logs/api-error.log
```

```
# 3. Start in debug mode
```

```
cd ~/prompt-machine/api
```

```
NODE_ENV=development node src/index.js
```

```
# 4. Fix issues, then restart normally
```

```
pm2 start ~/prompt-machine/ecosystem.config.js
```

```
sudo systemctl start nginx
```

Getting Help

1. **Check logs first** - They usually have the answer
2. **Use Claude Code** - "I'm getting error X when doing Y"
3. **Test in isolation** - Run components separately
4. **Start simple** - Get basic features working first

Remember: Most issues are configuration related, not code bugs!