Infrastrucuture

- Public API
  - (DEPRECATED)Dolt Server Documentation
- Pterodactyl
  - Updating Panel Troubleshooting

Public API

Public API

(DEPRECATED)Dolt Server Documentation

Dolt Database Server Setup Guide

This document outlines the complete process for setting up a Dolt database server that automatically synchronizes with DoltHub and provides MySQL-compatible access for applications like NocoDB.

System Overview

- **Database Platform**: Dolt (git for data)
- **Remote Repository**: DoltHub (fb-api-public)
- **Local Database Path**: /dolt/dbs/fb-api-public
- **Connection Type**: MySQL-compatible (port 3306)
- **Updates**: Automatically pulled hourly from DoltHub

Initial Installation

Install Dolt

```
# Linux
sudo bash -c 'curl -L https://github.com/dolthub/dolt/releases/latest/download/install.sh | bash'

# macOS
brew install dolt
```

Clone the Database

```
# Create directory if it doesn't exist
sudo mkdir -p /dolt/dbs
sudo chmod 777 /dolt/dbs

# Clone the repository
cd /dolt/dbs
dolt clone https://doltremoteapi.dolthub.com/fullbuff/fb-api-public
```

User Management

Connect to the Dolt SQL Shell

```
# Connect to the database using Dolt's SQL client
cd /dolt/dbs/fb-api-public
dolt -u root -p "" sql
```

Create a User for Applications

In the SQL shell:

```
-- Create a user with a password
CREATE USER 'nocodb'@'%' IDENTIFIED BY 'securepassword';

-- Grant read permissions
GRANT SELECT ON *.* TO 'nocodb'@'%';

-- Apply changes
FLUSH PRIVILEGES;

-- Verify user creation
SELECT User, Host FROM mysql.user WHERE User = 'nocodb';
```

Automated Service Setup

Create Update Script

Create a script to pull updates from DoltHub:

```
sudo nano /usr/local/bin/update-dolt.sh
```

Script content:

```
#!/bin/bash

# Log file for output
LOG_FILE="/var/log/dolt-update.log"

# Full path to dolt executable
DOLT_PATH=$(which dolt)

# Function to log messages
log() {
    echo "$(date '+%Y-%m-%d %H:%M:%S') - $1" >> "$LOG_FILE"
}

# Check if dolt is available
if [ -z "$DOLT_PATH" ]; then
    log "ERROR: dolt executable not found in PATH"
    exit 1
fi

log "Using dolt at: $DOLT_PATH"

# Check if database directory exists
if [ ! -d "/dolt/dbs/fb-api-public" ]; then
    log "ERROR: Database directory /dolt/dbs/fb-api-public does not exist"
    exit 1
fi

# Change to the database directory
cd /dolt/dbs/fb-api-public || {
    log "ERROR: Failed to change to database directory"
    exit 1
}

# Pull updates from main branch
```

```
log "Starting pull from origin main"
$DOLT_PATH pull origin main >> "$LOG_FILE" 2>&1
RESULT=$?

if [ $RESULT -eq 0 ]; then
    log "Successfully pulled updates"
else
    log "Failed to pull updates, exit code: $RESULT"
fi

exit $RESULT
```

Make the script executable:

```
sudo chmod +x /usr/local/bin/update-dolt.sh
```

Create Systemd Services

Create the Dolt Server service:

```
sudo nano /etc/systemd/system/dolt-server.service
```

Content:

```
[Unit]
Description=Dolt SQL Server
After=network.target

[Service]
Type=simple
User=root
WorkingDirectory=/dolt/dbs/fb-api-public
ExecStart=/usr/local/bin/dolt sql-server --host 0.0.0.0 --port 3306
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target
```

Create the update service:

```
sudo nano /etc/systemd/system/dolt-update.service
```

Content:

```
[Unit]
Description=Dolt Database Update
After=network.target
Before=dolt-server.service

[Service]
Type=oneshot
User=root
ExecStart=/usr/local/bin/update-dolt.sh
TimeoutStartSec=300

[Install]
WantedBy=multi-user.target
```

Configure Cron Job for Regular Updates

Set up a cron job to run the update script hourly:

```
sudo crontab -e
```

Add:

```
0 * * * * /usr/local/bin/update-dolt.sh
```

Enable and Start Services

```
sudo systemctl daemon-reload
sudo systemctl enable dolt-update.service
sudo systemctl enable dolt-server.service
sudo systemctl start dolt-update.service
sudo systemctl start dolt-server.service
```

Connecting to the Database

From the Command Line

```
# Connect as the root user
dolt -u root -p "" sql

# Connect as the application user
dolt -u nocodb -p "securepassword" sql
```

From Application (e.g., NocoDB)

Connection details:

- Host: Server IP address
- Port: 3306
- Username: nocodb
- Password: securepassword
- Database: fb_api_public

Troubleshooting

Check Service Status

```
sudo systemctl status dolt-server.service
sudo systemctl status dolt-update.service
```

View Update Logs

```
cat /var/log/dolt-update.log
```

Manual Database Operations

```
cd /dolt/dbs/fb-api-public

# Check status
dolt status
```

```
# View remote configuration
dolt remote -v

# Manually pull updates
dolt pull origin main
```

Restart Services

```
sudo systemctl restart dolt-update.service
sudo systemctl restart dolt-server.service
```

Notes

- The server runs in read-write mode to allow updates from DoltHub
- Local changes will be overwritten by the scheduled pull
- All permanent changes should be made through DoltHub's PR process
- The database is automatically updated hourly and on system boot
- Both services are configured to start automatically on system boot
- User authentication is handled with a password for security

This configuration provides a robust, automatically updating Dolt database server that maintains synchronization with DoltHub while providing MySQL-compatible access for applications.

Pterodactyl

Updating Panel Troubleshooting

Pterodactyl Panel Troubleshooting Guide

Introduction

This document outlines common issues encountered after updating Pterodactyl Panel and provides step-by-step solutions for diagnosing and fixing these problems. The guide is based on a real troubleshooting session where a Pterodactyl panel stopped loading after an update.

> Following the instructions in these pages should work most times.
> https://pterodactyl.io/panel/1.0/updating.html
> https://pterodactyl.io/guides/php_upgrade.html
>
> Blueprint docs if needed:
> https://blueprint.zip/docs/?page=getting-started/Installation
>
> Replace favicons in "**/var/www/pterodactyl/public/favicons**" with the ones found here
> https://mirror.fullbuff.gg/resources/favicons/

Common Issues After Pterodactyl Updates

1. Panel Fails to Load

The panel may fail to load after an update due to several possible causes:

- Missing PHP extensions
- PHP version mismatch

- Web server configuration issues
- Database connection problems
- Permission issues
- Maintenance mode not disabled
- Cache issues

2. Web Server Conflicts

One specific issue we encountered was having both Apache and NGINX trying to use port 80 simultaneously.

Diagnostic Steps

1. Check Panel Logs

First, examine the panel logs to identify specific errors:

```
tail -n 100 /var/www/pterodactyl/storage/logs/laravel-$(date +%F).log
```

Common error messages you might see:

- "could not find driver" (MySQL connection issue)
- "Class 'DOMDocument' not found" (Missing PHP XML extension)
- Database connection errors

2. Check Web Server Status

Verify that your web server is running:

```
systemctl status nginx
# or
systemctl status apache2
```

If you see errors like "bind() to 0.0.0.0:80 failed", it means something else is using port 80.

3. Check What's Using Port 80

To identify services using port 80:

```
lsof -i :80
# or
netstat -tuln | grep :80
```

4. Check Redis Status

Redis is needed for caching and queues:

```
systemctl status redis
# or
systemctl status redis-server
```

Common Solutions

1. Fix Missing PHP Extensions

Install required PHP extensions. Make sure to use your correct PHP version (in our case, 8.3):

```
apt update
apt install -y php8.3-mysql php8.3-xml php8.3-curl php8.3-gd php8.3-mbstring php8.3-zip
php8.3-bcmath php8.3-intl
```

2. Fix Web Server Conflicts

If both Apache and NGINX are trying to use port 80:

Option 1: Use NGINX (Recommended for Pterodactyl)

```
systemctl stop apache2
systemctl disable apache2
systemctl start nginx
systemctl enable nginx
```

Option 2: Use Apache

```
systemctl stop nginx
systemctl disable nginx
systemctl start apache2
```

```
systemctl enable apache2
```

3. Fix PHP-FPM Configuration

Make sure your web server is configured to use the correct PHP version:

1. Check your NGINX configuration:

```
nano /etc/nginx/sites-available/pterodactyl.conf
```

2. Look for the PHP-FPM socket path and update it to match your installed PHP version:

```
location ~ \.php$ {
    fastcgi_pass unix:/run/php/php8.3-fpm.sock;  # Adjust version number as needed
    # Other configuration...
}
```

3. Restart NGINX after making changes:

```
nginx -t  # Test the configuration
systemctl restart nginx
```

4. Fix Permissions

Set proper permissions on Pterodactyl files:

```
chown -R www-data:www-data /var/www/pterodactyl/*
chmod -R 755 /var/www/pterodactyl/storage/* /var/www/pterodactyl/bootstrap/cache/
```

5. Clear Cache

Clear various caches after updates:

```
cd /var/www/pterodactyl
php artisan config:clear
php artisan view:clear
php artisan cache:clear
```

6. Exit Maintenance Mode

If the panel was left in maintenance mode:

```
cd /var/www/pterodactyl
php artisan up
```

7. Restart Queue Worker

Always restart the queue worker after updates:

```
systemctl restart pteroq
```

Troubleshooting Third-party Extensions

If you have third-party extensions like Blueprint:

1. After fixing the core panel, reinstall the extension
2. Reinstall any dependencies required by the extension
3. Clear cache again after reinstalling extensions

If UI indicators show extensions as not fully installed despite working functionality, this might be a UI glitch that doesn't affect operations.

Preventative Measures for Future Updates

1. Always back up before updating
2. Check PHP version compatibility before updating
3. Make sure all required PHP extensions are installed
4. Follow the official update guide completely
5. Take note of any custom configurations or extensions you have installed
6. Monitor the logs during and after the update process

Conclusion

Most Pterodactyl panel issues after updates can be traced to missing PHP extensions, web server configuration mismatches, or permission problems. By systematically checking logs, verifying services, and applying the appropriate fixes, you can quickly restore functionality to your panel.