

# AI-03925 GCP Notes

👤 Owner	© Rigel Arcayan
☰ Tags	
🕒 Created time	June 12, 2026 4:10 PM

## Start Fresh

That explains it perfectly! In Docker, the **first time** the database container starts, it creates the database using the password in your `.yaml` file. If you change the password in the `.yaml` later, the database doesn't "know"—it keeps using the original one, causing a connection failure.

Since you are okay with starting fresh, here is the "Clean Slate" procedure to sync your new passwords and get the site live.

### 1. Stop and Wipe the old data

Run these commands in your `wordpress-site` folder to remove the containers and the "stale" database volumes:

```
sudo docker compose down
sudo docker volume rm wordpress-site_db_data wordpress-site_w
p_data
sudo docker compose up -d
```

Generated code may be subject to license restrictions not shown here. Use code with care. [Learn more](#)

(Note: If your folder name is different, use `sudo docker volume ls` to find the exact volume names).

### 2. Verify your `docker-compose.yaml`

Open your file in **VS Code** one last time and ensure the `wordpress` service has the corrected volume path (no double `/var`):

Generated code may be subject to license restrictions not shown here. Use code with care. [Learn more](#)

### 3. Start Fresh

Now, run the command to build the new database and site with your standardized `1801Gunsight` password:

Generated code may be subject to license restrictions not shown here. Use code with care. [Learn more](#)

### 4. Wait 30 seconds, then test

The database takes about 20–30 seconds to initialize its files for the first time. After that, open your browser (Incognito is best):

`http://35.252.76.214`

#### What to expect:

You should see the **WordPress Installation Wizard** (the screen where you pick your Language).

**Journal Note:** This is a classic "Docker Gotcha." Environment variables for passwords only work on the **very first run**. If you ever want to change a database password later without wiping the data, you have to change it *inside* the database (using Webmin or terminal) **and** in the `.yml` file. Starting fresh is much easier for now!

## Disaster Recovery

That is fantastic news! Having all three (WordPress, Portainer, and Webmin) running is a significant milestone in your GCP journey. Your "saw" is definitely getting sharper.

Since you are used to backing up `.vhdx` files in Hyper-V, you'll find that GCP has a very similar (and actually more powerful) way to do exactly that.

### 1. The "Beginner" Scheme: Snapshots

In GCP, a **Snapshot** is the direct equivalent of your `.vhdx` backup. It is a point-in-time copy of your entire disk.

- **How to do it:**
    1. Go to **Compute Engine > Snapshots**.
    2. Click **CREATE SNAPSHOT**.
    3. Select your `wordpress-docker-vm` disk as the source.
  - **The "Quick Fix" recovery:** If you break your VM, you can create a new disk from that snapshot and swap it onto your VM, or just delete the VM and "Restore" from the snapshot. It takes about 2 minutes to get back to a "Last Known Good" state.
  - **Pro Tip:** Set up a **Snapshot Schedule** (look for "Snapshot Schedules" in the sidebar). You can tell GCP to take a backup every night at 2 AM automatically and keep the last 7 days.
- 

## 2. The "Intermediate" Scheme: Machine Images

If a Snapshot is a backup of a *disk*, a **Machine Image** is a backup of the *entire VM*.

- **Why use this:** It saves the disk, but also the VM settings (like those **Network Tags** we added, the Machine Type, and the IP settings).
  - **Recovery:** You can right-click a Machine Image and say "Create Instance," and it launches a perfect twin of your original VM.
- 

## 3. The "GCP Master" Scheme: Disposable Infrastructure

This is the "Seasoned Admin" secret: **They don't back up the VM at all**. Instead, they make the VM **disposable**.

Masters use a combination of:

1. **External Data:** Instead of saving the WordPress database *inside* the VM, they use **Cloud SQL** (a managed database service).
2. **External Files:** They store the WordPress images/uploads in **Cloud Storage** (buckets).
3. **The "Recipe":** They keep their `docker-compose.yml` and a **Startup Script** in a GitHub repository.

**How the Master "Recovers":**

If their VM gets "hacked" or corrupted, they don't restore it. They **delete it**. Then, they run a script that:

1. Spins up a brand-new, clean Ubuntu VM.
2. Installs Docker automatically.
3. Pulls their `.yaml` files from GitHub.
4. Reconnects to the external Database. *This is called "**Immutable Infrastructure**." The VM is just a "worker" that can be replaced in seconds without losing any data.*

## Your Next Step for the Journal:

For now, stick to **Snapshots**. They are your "Safety Net."

1. Go create a manual snapshot of your VM right now while it's working perfectly.
2. Name it `wordpress-perfect-setup-date` .

Now you can experiment as much as you want, knowing you have a "Undo" button!

**Congratulations on your progress today!** You've gone from a broken IP and a blank white screen to a fully managed Docker environment on Google Cloud. That's a huge win.