

Phpmyadmin vhost setup

I want to add phpmyadmin with a vhost in my apache server. I don't want to install phpmyadmin in my system. Can I do it without installation?

Short answer: yes. You don't have to "install" phpMyAdmin via apt at all. Two clean ways:

Option A — "Drop-in" deploy (no system package)

1. Make a directory for it (anywhere Apache can serve):

```
bash

sudo mkdir -p /var/www/pma
cd /var/www/pma
# grab the latest release archive manually downloaded or via wget/curl
# e.g., phpMyAdmin-*-all-languages.zip or .tar.gz
sudo tar -xf ~/Downloads/phpMyAdmin-*-all-languages.tar.gz --strip-components=1
sudo chown -R www-data:www-data /var/www/pma
```

2. Create minimal config:

```
bash

sudo tee /var/www/pma/config.inc.php >/dev/null <<'PHP'
<?php
$config['blowfish_secret'] = 'put-32+random-chars-here_____'; //
required_for cookie auth
$config['TempDir'] = __DIR__ . '/tmp';
$config['Servers'][1]['auth_type'] = 'cookie';
PHP
sudo mkdir /var/www/pma/tmp && sudo chown www-data:www-data /var/www/pma/tmp
```

3. Make a dedicated Apache vhost (so it's isolated from other sites):

```
apache
```

```
# /etc/apache2/sites-available/pma.conf
<VirtualHost *:80>
    ServerName pma.local
    DocumentRoot /var/www/pma

    <Directory /var/www/pma>
        AllowOverride All
        Require all granted
    </Directory>

    # Optional: extra hardening — allow only your LAN/IP
    #<Directory /var/www/pma>
    #    Require ip 192.168.0.0/16
    #</Directory>

    ErrorLog ${APACHE_LOG_DIR}/pma_error.log
    CustomLog ${APACHE_LOG_DIR}/pma_access.log combined
</VirtualHost>
```

4. Enable and reload:

```
bash

echo "127.0.0.1 pma.local" | sudo tee -a /etc/hosts
sudo a2ensite pma.conf
sudo systemctl reload apache2
```

5. Visit <http://pma.local> and log in with your MySQL credentials.

Notes:

- This doesn't "install" anything system-wide; it's just static files served by Apache using your existing PHP.
- Make sure PHP has required extensions (mysqli, mbstring, json, session, openssl, zip).
- For internet exposure, use HTTPS + IP allowlist or basic auth. Otherwise keep it local only.

Option B — Docker behind Apache (also no host install)

If you prefer zero files on the host:

```
bash

docker run -d --name pma \
-e PMA_HOST=127.0.0.1 \
-p 127.0.0.1:9000:80 \
phpmyadmin:latest
```

Then add an Apache vhost that reverse-proxies to the container:

```
apache

# /etc/apache2/sites-available/pma.conf
<VirtualHost *:80>
    ServerName pma.local
    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:9000/
    ProxyPassReverse / http://127.0.0.1:9000/
    ErrorLog ${APACHE_LOG_DIR}/pma_error.log
    CustomLog ${APACHE_LOG_DIR}/pma_access.log combined
</VirtualHost>
```

Enable needed modules and site:

```
bash

sudo a2enmod proxy proxy_http
sudo a2ensite pma.conf
sudo systemctl reload apache2
```

Want me to tailor the vhost to your exact Apache/PHP versions and existing sites (ports, SSL, access rules)? If you paste your current vhost layout, I'll slot phpMyAdmin in neatly.

↳ 2. Create minimal config: bash CopyEdit

what does config do, can this step be omitted?

Great question.

Short version: that tiny `config.inc.php` sets a couple things phpMyAdmin **can't safely guess**, mainly the encryption key for cookie-based logins. If you skip it, phpMyAdmin will usually **refuse to log you in** (or nag you hard), so it's not a step you can just drop—unless you set the same values another way.

Here's what those lines do:

- `$cfg['blowfish_secret'] = '...';`

Required for `auth_type = 'cookie'` (the default). phpMyAdmin uses this secret to encrypt/decrypt your login cookie.

- If missing → you'll get an error and can't log in with cookie auth.
- You can generate any 32+ random characters.

- `$cfg['TempDir'] = __DIR__ . '/tmp';`

A writable temp folder phpMyAdmin uses for things like big imports/exports, cache, and some plugins.

- If missing/not writable → basic browsing may still work, but some features break or throw warnings.

- `$cfg['Servers'][1]['auth_type'] = 'cookie';`

Makes the login screen use cookies (safest for shared hosts). You could choose `http` or `config`, but `cookie` is recommended.

So, can you omit the config step?

- **No, not if you're using cookie auth** (default). You must provide a `blowfish_secret` —either via:
 - a minimal `config.inc.php` (what I showed),
 - the built-in **Setup Wizard** (`/setup`) to generate the file,
 - or, in Docker, by passing env vars and letting the image generate config.
- **Yes, if you switch to** `auth_type = 'http'` (Apache Basic Auth) and don't mind the browser's login prompt. Even then, I still recommend setting `TempDir`, and you'll likely want to define the server host.

If you want the absolute minimum file that unblocks login, this works:

php

```
<?php
$cfg['blowfish_secret'] = 'use-a-unique-32+-char-random-string-here_____';
```

```
$cfg['Servers'][1]['auth_type'] = 'cookie';  
$cfg['TempDir'] = __DIR__ . '/tmp';
```

Create the `tmp` folder and make it writable by Apache:

```
bash  
  
mkdir /var/www/pma/tmp  
chown www-data:www-data /var/www/pma/tmp
```

If you'd rather avoid creating the file by hand, I can walk you through using `/setup` to generate it, or switch you to `auth_type = 'http'` and harden the vhost. Your call!

↳ `sudo tar -xf ~/Downloads/phpMyAdmin-*-all-languages.tar.gz --strip-components=1`

this command is giving me this error:
tar: /home/itminan/Downloads/phpMyAdmin-*-all-languages.tar.gz: Cannot open: No such file or directory
tar: Error is not recoverable: exiting now

That error means the file you're trying to extract simply isn't there —

`~/Downloads/phpMyAdmin-*-all-languages.tar.gz` is just a placeholder path, not an actual file on your system yet.

What's happening:

- `*` is a shell wildcard — it matches a real filename.
- If there's nothing in `~/Downloads` that matches `phpMyAdmin-*-all-languages.tar.gz`, `tar` complains.

How to fix it

1 Actually download phpMyAdmin first

You can grab the latest release from the official site:

```
bash
```

```
wget https://files.phpmyadmin.net/phpMyAdmin/5.2.1/phpMyAdmin-5.2.1-all-languages.tar.gz
```

Example for `/var/www/pma`:

```
sudo mkdir -p /var/www/pma
sudo tar -xf ~/Downloads/phpMyAdmin-5.2.1-all-languages.tar.gz -C /var/www/pma --strip-components=1
```

```
bash

sudo chown -R www-data:www-data /var/www/pma
```

how to edit the config file, explain it more please.

Got it — let me walk you through **exactly** how to create and edit the phpMyAdmin `config.inc.php` file step-by-step, and why each part matters.

1 Where the file goes

- phpMyAdmin looks for `config.inc.php` in its root folder.
- If you extracted phpMyAdmin to:

```
bash

/var/www/pma
```

then the config file path will be:

```
swift

/var/www/pma/config.inc.php
```

2 Create the file

Open a terminal and run:

```
bash

sudo nano /var/www/pma/config.inc.php
```

(Replace `nano` with `vim` or any editor you like.)

3 Paste this template

```
php

<?php
/**
 * phpMyAdmin configuration
 */

// 1. A secret passphrase for encrypting cookies.
// Must be 32+ characters, random, no quotes inside.
```

```
$cfg['blowfish_secret'] = 'Bf9#vF2LIsQz@T7eWcN0kMp4Hr^1yXu$';
```

```
// 2. Directory for temporary files (must be writable by Apache).
```

```
$cfg['TempDir'] = __DIR__ . '/tmp';
```

```
// 3. Database server connection settings.
```

```
// [1] means "first server".
```

```
$cfg['Servers'][1]['auth_type'] = 'cookie'; // cookie-based login
```

```
$cfg['Servers'][1]['host'] = '127.0.0.1'; // MySQL host
```

```
$cfg['Servers'][1]['compress'] = false;
```

```
$cfg['Servers'][1]['AllowNoPassword'] = false;
```

4 Generate a secure blowfish secret

Run:

```
bash
```

```
openssl rand -base64 48
```

This will output a random string — copy it into `$cfg['blowfish_secret']` (replace my example string).

5 Create the temp directory

```
bash
```

```
sudo mkdir /var/www/pma/tmp
```

```
sudo chown www-data:www-data /var/www/pma/tmp
```

```
sudo chmod 755 /var/www/pma/tmp
```

- `www-data` is the Apache user on Ubuntu/Debian.
- This folder is where phpMyAdmin will store cache files and large imports.

6 Why each part is important

Setting	Purpose	If Missing
<code>\$cfg['blowfish_secret']</code>	Encrypts/decrypts cookies for login sessions	Cookie login won't work; phpMyAdmin will throw an error
<code>\$cfg['TempDir']</code>	Stores temporary data (cache, imports)	Some features break, large imports may fail
<code>\$cfg['Servers'][1]['auth_type']</code>	Chooses login method (cookie is safest)	Defaults may be insecure or cause login issues
<code>\$cfg['Servers'][1]['host']</code>	Tells phpMyAdmin which MySQL/MariaDB to connect to	Defaults to <code>localhost</code> , may fail if DB is remote

7 Save & exit

- In `nano`: press `CTRL+O` → `ENTER` → `CTRL+X`.

If you want, I can also give you a **ready-made config** that works for 99% of local Apache+MySQL setups, so you just paste it and go.

Do you want me to prepare that?