

PowerDNS Server

Install PowerDNS + Recursor + Admin + MariaDB + DNSdist

Розгортаємо високопродуктивний безкоштовний авторитарний DNS-сервер для тримання власних зон доменів



All we do from sudo user!!!

```
sudo su
```

Install MariaDB

[Install MariaDB](#)

За бажанням можна встановити apache2 та phpMyAdmin для доступу до бази даних через веб панель керування

[Install apache2](#)

[Install phpMyAdmin](#)

Install PowerDNS server

В репозиторіях є зазвичай, застаріла версія, тому заглянемо на сайт <https://repo.powerdns.com/>

Там знаходимо стабільну версію програми і застосовуємо зміни в списку репозиторіїв

```
sh -c 'echo "deb [arch=amd64] http://repo.powerdns.com/debian bullseye-auth-46 main" >
/etc/apt/sources.list.d/pdns.list'
```

```
tee /etc/apt/preferences.d/pdns<<EOF
Package: pdns-*
Pin: origin repo.powerdns.com
Pin-Priority: 600
EOF
```

```
curl https://repo.powerdns.com/FD380FBB-pub.asc | sudo gpg --no-default-keyring --keyring gnupg-
ring:/etc/apt/trusted.gpg.d/powerdns.gpg --import
chmod 644 /etc/apt/trusted.gpg.d/*
```

```
apt-get update
apt-get install -y pdns-server pdns-backend-mysql
cp -r /etc/powerdns/ /etc/powerdns.orig/
```

```
tee /etc/powerdns/pdns.conf<<EOF
setgid=pdns
setuid=pdns
local-address=127.0.0.1
local-port=53
include-dir=/etc/powerdns/pdns.d
launch=
EOF
```

prepeare mysql db

```
mysql -u root -p
```

```
CREATE DATABASE powerdns;
GRANT ALL ON powerdns.* TO 'powerdns'@'localhost' IDENTIFIED BY 'strongpassword';
```

```
FLUSH PRIVILEGES;  
quit;
```

Якщо це нове встановлення, то створимо структуру бази даних

```
mysql powerdns < /usr/share/pdns-backend-mysql/schema/schema.mysql.sql
```

Інакше, відновимо структуру з вашої резервної копії

```
mysql powerdns < /tmp/backup_powerdns.sql
```

Після відновлення з резервної копії, перевіряємо наявність змін структури

```
ls /usr/share/pdns-backend-mysql/schema/ | grep -E -i '^[0-9]'  
"3.4.0_to_4.1.0_schema.mysql.sql  
4.1.0_to_4.2.0_schema.mysql.sql  
4.2.0_to_4.3.0_schema.mysql.sql"
```

Якщо вони є, застосовуємо всі почерзі, починаючи з версії з резервної копії, наприклад з 4.1.0

```
mysql powerdns < /usr/share/pdns-backend-mysql/schema/4.1.0_to_4.2.0_schema.mysql.sql  
mysql powerdns < /usr/share/pdns-backend-mysql/schema/4.2.0_to_4.3.0_schema.mysql.sql
```

Підключимо нашу базу даних

```
nano /etc/powerdns/pdns.d/pdns.local.gmysql.conf
```

```
# MySQL Configuration  
# Launch gmysql backend  
launch+=gmysql  
# gmysql parameters  
gmysql-host=127.0.0.1  
gmysql-port=3306
```

```
gmysql-dbname=powerdns
gmysql-user=powerdns
gmysql-password=strongpassword
gmysql-dnssec=yes
# gmysql-socket=
```

```
systemctl restart pdns
netstat -tap | grep pdns
```

```
dig mydomain.local @127.0.0.1
dig @127.0.0.1
```

Панель адміністрування PowerDNS-Admin

<https://computingforgeeks.com/install-powerdns-and-powerdns-admin-on-debian/>

<https://github.com/PowerDNS-Admin/PowerDNS-Admin/wiki/Running-PowerDNS-Admin-on-Ubuntu-or-Debian>

```
apt install python3-dev python3-venv git libmysqlclient-dev libsasl2-dev libldap2-dev libssl-dev libxml2-dev
libxslt1-dev libxmlsec1-dev libffi-dev pkg-config apt-transport-https virtualenv build-essential curl jq
```

```
curl -sL https://deb.nodesource.com/setup_18.x | bash -
apt-get update && apt install -y nodejs
```

```
curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | gpg --dearmor | sudo tee /usr/share/keyrings/yarnkey.gpg >
/dev/null
echo "deb [signed-by=/usr/share/keyrings/yarnkey.gpg] https://dl.yarnpkg.com/debian stable main" | sudo tee
/etc/apt/sources.list.d/yarn.list
apt-get update && apt-get install yarn
```

```
mysql
```

```
CREATE DATABASE poweradmin CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
GRANT ALL PRIVILEGES ON poweradmin.* TO 'pdnsadmin'@'%' IDENTIFIED BY 'p4ssw0rd';
FLUSH PRIVILEGES;
quit
```

```
git clone https://github.com/PowerDNS-Admin/PowerDNS-Admin.git /opt/web/powerdns-admin
```

```
cd /opt/web/powerdns-admin
python3 -m venv .venv --prompt VirtualEnv
source .venv/bin/activate
python -V
```

```
pip install --upgrade pip
pip install -r requirements.txt
```

```
cp /opt/web/powerdns-admin/configs/development.py /opt/web/powerdns-admin/configs/production.py
nano /opt/web/powerdns-admin/configs/production.py
```

```
import os
import urllib.parse
basedir = os.path.abspath(os.path.dirname(__file__))

### BASIC APP CONFIG
SALT = '$2b$12$yLUMTIfl21FKJQpTkRQXCu'
SECRET_KEY = 'e951e5a1f4b94151b360f47edf596dd2'
BIND_ADDRESS = '0.0.0.0'
PORT = 9191
OFFLINE_MODE = False

### DATABASE CONFIG
SQLA_DB_USER = 'pdnsadmin'
SQLA_DB_PASSWORD = 'p4ssw0rd'
SQLA_DB_HOST = '127.0.0.1'
SQLA_DB_NAME = 'poweradmin'
```

```
SQLALCHEMY_TRACK_MODIFICATIONS = True
```

```
### DATABASE - MySQL
```

```
SQLALCHEMY_DATABASE_URI = 'mysql://{}:{}@{}/{}'.format(  
    urllib.parse.quote_plus(SQLA_DB_USER),  
    urllib.parse.quote_plus(SQLA_DB_PASSWORD),  
    SQLA_DB_HOST,  
    SQLA_DB_NAME  
)
```

```
### DATABASE - SQLite
```

```
#SQLALCHEMY_DATABASE_URI = 'sqlite:/// ' + os.path.join(basedir, 'pdns.db')
```

```
nft add rule ip filter input ct state new tcp dport 9191 counter accept comment "PDNS_ADMIN"
```

```
export FLASK_CONF=../configs/production.py  
export FLASK_APP=powerdnsadmin/__init__.py  
flask db upgrade
```

```
yarn install --pure-lockfile  
flask assets build
```

```
./run.py
```

```
tee /etc/systemd/system/powerdns-admin.service<<EOF  
[Unit]  
Description=PowerDNS-Admin  
Wants=network.target  
After=network.target mysqld.service mariadb.service  
  
[Service]  
PIDFile=/run/powerdns-admin/pid  
User=www-data  
Group=root
```

```
WorkingDirectory=/opt/web/powerdns-admin
ExecStart=/opt/web/powerdns-admin/venv/bin/gunicorn --pid /run/powerdns-admin/pid --workers 2 --bind
'0.0.0.0:9191' 'powerdnsadmin:create_app()'
ExecReload=/bin/kill -s HUP $MAINPID
ExecStop=/bin/kill -s TERM $MAINPID
PrivateTmp=true
Restart=on-failure
RestartSec=10
StartLimitInterval=0

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

```
mkdir -p /etc/systemd/system/powerdns-admin.service.d/
tee /etc/systemd/system/powerdns-admin.service.d/override.conf<<EOF
[Service]
Environment="FLASK_CONF=../configs/production.py"
EOF
```

```
echo 'd /run/powerdns-admin 0755 www-data root -' > /etc/tmpfiles.d/powerdns-admin.conf
mkdir -p /run/powerdns-admin
chown -R www-data:root /run/powerdns-admin
chown -R www-data:root /opt/web/powerdns-admin
```

```
systemctl daemon-reload
systemctl enable powerdns-admin
systemctl restart powerdns-admin
systemctl status powerdns-admin
```

```
nano /etc/powerdns/pdns.d/api.conf
```

```
webserver-port=8081
api=yes
```

```
api-key=changemechangeme
```

```
chown -R pdns:root /etc/powerdns/  
systemctl restart pdns
```

Visit PowerDNS-Admin web interface. <http://your-host-ip:9191/>

Click “Create an account” button and Register a user. The first user will be in the Administrator role.

Provide PowerDNS API connection information before PowerDNS-Admin can query your PowerDNS data. This is done under Settings > PDNS

Нотатки

[Установка DNSCrypt-сервера](#)

<https://dnslookup.online/ptr.html>

<https://github.com/ngoduykhanh/PowerDNS-Admin/wiki>

<https://repo.powerdns.com/>

<https://fossies.org/linux/pdns-dnsdist/pdns/dnsdistdist/docs/advanced/snmp.rst>

```
pdnsutil create-zone example.com ns1.example.com
```

```
/etc/init.d/pdns-recursor restart  
rec_control wipe-cache  
/etc/init.d/pdns-recursor status
```


From:
<https://ndp.pp.ua/> - my NoDeny Wiki

Permanent link:
https://ndp.pp.ua/doku.php/debian/pdns_server?rev=1745019352

Last update: **2025/04/19 02:35**

