

PowerDNS Recursor

Install PowerDNS + Recursor + Admin + MariaDB + DNSdist

Розгортаємо високопродуктивний безкоштовний DNS-сервер

Можливе кілька сценаріїв установки. У даному прикладі буде розглянуто варіант установки авторитарного сервера з доменами користувача, окремим рекурсивним сервером і балансувальником dnsdist



All we do from sudo user!!! Після будь-якого повідомлення, до команди “sudo su” and retype your password!

Install MariaDB



Починаючи з версії MariaDB 10.11 її кодова база суттєво відійшла і втратила сумісність з іншими форками MySQL.

Критичні відмінності в роботі з GTID унеможливають реплікацію з іншими форками, а одже і міграцію на них без простою БД.

На мою думку, краще використовувати [Percona MySQL Server 8](#)



Цей розділ більше мені не потрібен, тому з часом, його буде видалено

<https://blog.programs74.ru/how-to-install-mariadb-10-5-on-debian-10/>

```
apt-get install -y mariadb-server libmariadbclient-dev libmariadbclient-dev-compat
```

```
systemctl status mariadb
```

```
sudo mysql_secure_installation
```

```
-- It does not ask me for any password  
sudo mysql
```

```
-- Then in MariaDB/MySQL console:
```

```
UPDATE mysql.user SET plugin = 'mysql_native_password' WHERE USER='root';  
FLUSH PRIVILEGES;  
exit;
```



<https://galeracluster.com/library/documentation/galera-manager-monitoring-clusters.html>
<https://galeracluster.com/library/documentation/galera-manager.html>
<https://bogachev.biz/2017/03/28/neskolko-sposobov-vosstanovleniya-mariadb-galera-cluster/>
<https://galeracluster.com/library/documentation/arbitrator.html>

fix [Warning] [MY-013360] [Server] Plugin sha256_password reported

2022/03/05 01:01 · Method

Install apache2 php phpMyAdmin

Встановлення Apache

```
apt-get install -yq apache2
```

Включаємо потрібні нам модулі для апача:

```
sudo a2enmod ssl
```

```
sudo a2enmod rewrite
sudo a2enmod suexec
sudo a2enmod include
```

Застосовуємо зміни

```
systemctl restart apache2
systemctl status apache2
```

(1) Для включення https <https://www.8host.com/blog/sozдание-sertifikata-lets-encrypt-dlya-apache-v-debian-10/>

2022/03/05 00:05 · Method

<https://computingforgeeks.com/install-phpmyadmin-with-apache-on-debian-10-buster/>

Install phpMyAdmin php

```
sudo apt -y install wget php php-cgi php-pear php-mbstring php8.2-gettext libapache2-mod-php php-common php-
phpseclib php-mysql
```

```
cd /tmp
DATA="$(wget https://www.phpmyadmin.net/home_page/latest.txt -q -O-)"
URL="$(echo $DATA | cut -d ' ' -f 3)"
VERSION="$(echo $DATA | cut -d ' ' -f 1)"
wget https://files.phpmyadmin.net/phpMyAdmin/${VERSION}/phpMyAdmin-${VERSION}-all-languages.tar.gz
tar xvf phpMyAdmin-${VERSION}-all-languages.tar.gz
```

Delete compressed file and move the resulting folder to /usr/share/phpmyadmin folder.

```
rm *.tar.gz
sudo mv phpMyAdmin-*/ /var/www/phpmyadmin
```

Create directory for phpMyAdmin temp files.

```
sudo mkdir -p /var/lib/phpmyadmin/tmp
sudo chown -R www-data:www-data /var/lib/phpmyadmin
sudo mkdir /etc/phpmyadmin/
sudo cp /var/www/phpmyadmin/config.sample.inc.php /var/www/phpmyadmin/config.inc.php
nano /var/www/phpmyadmin/config.inc.php
```

#Edit the file and set secret passphrase:

```
$cfg['blowfish_secret'] = 'H20xcGXf1Sd8JwrwVlh6KW6s2rER63i';
```

//Configure Temp directory by add

```
$cfg['TempDir'] = '/var/lib/phpmyadmin/tmp';
```

```
nano /etc/apache2/conf-available/phpmyadmin.conf
```

phpMyAdmin default Apache configuration

```
Alias /pma /var/www/phpmyadmin
```

```
<Directory /var/www/phpmyadmin>
    Options SymLinksIfOwnerMatch
    DirectoryIndex index.php
```

```
    <IfModule mod_php5.c>
        <IfModule mod_mime.c>
            AddType application/x-httpd-php .php
        </IfModule>
        <FilesMatch ".+\.php$">
            SetHandler application/x-httpd-php
        </FilesMatch>
```

```
    php_value include_path .
    php_admin_value upload_tmp_dir /var/lib/phpmyadmin/tmp
```

```
php_admin_value open_basedir /var/www/phpmyadmin/:/etc/phpmyadmin/:/var/lib/phpmyadmin/:/var/www/php/php-  
gettext/:/var/www/php/php-  
gettext/:/var/www/javascript/:/var/www/php/tcpdf/:/var/www/doc/phpmyadmin/:/var/www/php/phpseclib/  
php_admin_value mbstring.func_overload 0  
</IfModule>  
<IfModule mod_php.c>  
  <IfModule mod_mime.c>  
    AddType application/x-httpd-php .php  
  </IfModule>  
  <FilesMatch ".+\.php$">  
    SetHandler application/x-httpd-php  
  </FilesMatch>  
  
  php_value include_path .  
  php_admin_value upload_tmp_dir /var/lib/phpmyadmin/tmp  
  php_admin_value open_basedir  
/var/www/phpmyadmin/:/etc/phpmyadmin/:/var/lib/phpmyadmin/:/usr/share/php/php-gettext/:/usr/share/php/php-  
gettext/:/usr/share/javascript/:/usr/share/php/tcpdf/:/var/www/doc/phpmyadmin/:/usr/share/php/phpseclib/  
  php_admin_value mbstring.func_overload 0  
</IfModule>  
  
</Directory>  
  
# Authorize for setup  
<Directory /var/www/phpmyadmin/setup>  
  <IfModule mod_authz_core.c>  
    <IfModule mod_authn_file.c>  
      AuthType Basic  
      AuthName "phpMyAdmin Setup"  
      AuthUserFile /etc/phpmyadmin/htpasswd.setup  
    </IfModule>  
    Require valid-user  
  </IfModule>  
</Directory>
```

```
# Disallow web access to directories that don't need it
<Directory /var/www/phpmyadmin/templates>
    Require all denied
</Directory>
<Directory /var/www/phpmyadmin/libraries>
    Require all denied
</Directory>
<Directory /var/www/phpmyadmin/setup/lib>
    Require all denied
</Directory>
```

You can restrict access from specific IP by adding line like below

```
Require ip 127.0.0.1 192.168.18.0/24
```

```
a2enconf phpmyadmin.conf
systemctl reload apache2
systemctl restart apache2
```

Visit phpMyAdmin Web interface Open the URL [http://\[ServerIP/Hostname\]/pma](http://[ServerIP/Hostname]/pma) to login into phpMyAdmin dashboard with your Database credentials – username & password.

2022/03/05 01:19 · Method

recursor

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

```
apt-get update
apt-get install -y pdns-recursor
nano /etc/powerdns/recursor.conf
```

```
setgid=pdns
setuid=pdns
config-dir=/etc/powerdns
include-dir=/etc/powerdns/recursor.d
quiet=yes
security-poll-suffix=
```

```
nano /etc/powerdns/recursor.d/recursor.local.conf
```

```
local-address=0.0.0.0
local-port=53
```

```
nano /etc/powerdns/recursor.d/allow-from.conf
```

```
# !!! Першою лінією мастика є запис "allow-from=[ip/mask]", будь-яка інша лінія як "allow-from+=[ip/mask]" (WITH "+" before "=") !!!
allow-from=0.0.0.0/0 # Allow from any
allow-from+=127.0.0.1/8 # Allow from loopback
allow-from+=193.0.0.1/8 # Allow from arpa (PTR)
allow-from+=192.168.0.1/16 # Allow from arpa (PTR)
```

```
cd /etc/powerdns/
wget ftp://ftp.rs.internic.net/domain/root.zone.gz && gunzip root.zone.gz
systemctl disable systemd-resolved
systemctl stop systemd-resolved
ls -lh /etc/resolv.conf
mv /etc/resolv.conf /etc/resolv.old.conf
echo "nameserver 127.0.0.1" > /etc/resolv.conf
systemctl restart pdns-recursor
```

```
echo "forward-zones-recurse=.1.1.1.1;1.0.0.1;8.8.8.8;8.8.4.4;" > /etc/powerdns/recursor.d/forward-zones-recurse.conf
```

prepeare mysql db

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

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

```
CREATE TABLE domains (  
  id INT AUTO_INCREMENT,  
  name VARCHAR(255) NOT NULL,  
  master VARCHAR(128) DEFAULT NULL,  
  last_check INT DEFAULT NULL,  
  type VARCHAR(8) NOT NULL,  
  notified_serial INT UNSIGNED DEFAULT NULL,  
  account VARCHAR(40) CHARACTER SET 'utf8' DEFAULT NULL,  
  options VARCHAR(64000) DEFAULT NULL,  
  catalog VARCHAR(255) DEFAULT NULL,  
  PRIMARY KEY (id)  
) Engine=InnoDB CHARACTER SET 'latin1';
```

```
CREATE UNIQUE INDEX name_index ON domains(name);  
CREATE INDEX catalog_idx ON domains(catalog);
```

```
CREATE TABLE records (  
  id BIGINT AUTO_INCREMENT,
```

```
domain_id          INT DEFAULT NULL,
name               VARCHAR(255) DEFAULT NULL,
type              VARCHAR(10) DEFAULT NULL,
content            VARCHAR(64000) DEFAULT NULL,
ttl               INT DEFAULT NULL,
prio              INT DEFAULT NULL,
disabled           TINYINT(1) DEFAULT 0,
ordername         VARCHAR(255) BINARY DEFAULT NULL,
auth              TINYINT(1) DEFAULT 1,
PRIMARY KEY (id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX nametype_index ON records(name,type);
CREATE INDEX domain_id ON records(domain_id);
CREATE INDEX ordername ON records (ordername);

CREATE TABLE supermasters (
  ip                VARCHAR(64) NOT NULL,
  nameserver        VARCHAR(255) NOT NULL,
  account           VARCHAR(40) CHARACTER SET 'utf8' NOT NULL,
  PRIMARY KEY (ip, nameserver)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE TABLE comments (
  id                INT AUTO_INCREMENT,
  domain_id         INT NOT NULL,
  name              VARCHAR(255) NOT NULL,
  type              VARCHAR(10) NOT NULL,
  modified_at       INT NOT NULL,
  account           VARCHAR(40) CHARACTER SET 'utf8' DEFAULT NULL,
  comment           TEXT CHARACTER SET 'utf8' NOT NULL,
  PRIMARY KEY (id)
```

```
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX comments_name_type_idx ON comments (name, type);
CREATE INDEX comments_order_idx ON comments (domain_id, modified_at);

CREATE TABLE domainmetadata (
  id                INT AUTO_INCREMENT,
  domain_id         INT NOT NULL,
  kind              VARCHAR(32),
  content           TEXT,
  PRIMARY KEY (id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX domainmetadata_idx ON domainmetadata (domain_id, kind);

CREATE TABLE cryptokeys (
  id                INT AUTO_INCREMENT,
  domain_id         INT NOT NULL,
  flags             INT NOT NULL,
  active            BOOL,
  published         BOOL DEFAULT 1,
  content           TEXT,
  PRIMARY KEY(id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE INDEX domainidindex ON cryptokeys(domain_id);

CREATE TABLE tsigkeys (
  id                INT AUTO_INCREMENT,
  name              VARCHAR(255),
  algorithm         VARCHAR(50),
```

```
secret          VARCHAR(255),
PRIMARY KEY (id)
) Engine=InnoDB CHARACTER SET 'latin1';

CREATE UNIQUE INDEX namealgoindex ON tsigkeys(name, algorithm);

ALTER TABLE records ADD CONSTRAINT `records_domain_id_ibfk` FOREIGN KEY (`domain_id`) REFERENCES `domains` (`id`)
ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE comments ADD CONSTRAINT `comments_domain_id_ibfk` FOREIGN KEY (`domain_id`) REFERENCES `domains`
(`id`) ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE domainmetadata ADD CONSTRAINT `domainmetadata_domain_id_ibfk` FOREIGN KEY (`domain_id`) REFERENCES
`domains` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
ALTER TABLE cryptokeys ADD CONSTRAINT `cryptokeys_domain_id_ibfk` FOREIGN KEY (`domain_id`) REFERENCES `domains`
(`id`) ON DELETE CASCADE ON UPDATE CASCADE;
quit;
```

Install PowerDNS server

```
apt-get update
apt-get install -y pdns-server pdns-backend-mysql
nano /etc/powerdns/pdns.d/pdns.local.gmysql.conf
```

```
# MySQL Configuration
# Launch gmysql backend
launch+=gmysql
# gmysql parameters
gmysql-host=localhost
gmysql-port=3306
gmysql-dbname=powerdns
gmysql-user=powerdns
gmysql-password=strongpassword
gmysql-dnssec=yes
```

```
# gmysql-socket=
```

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

```
setgid=pdns  
setuid=pdns  
local-address=127.0.0.1  
local-port=5300  
include-dir=/etc/powerdns/pdns.d  
launch=
```

```
systemctl start pdns-recursor  
systemctl start pdns  
netstat -tap | grep pdns
```

```
dig mydomain.local @127.0.0.1 -p 5300  
dig @127.0.0.1
```

Заголовок

```
apt-get install -y python3-dev libmysqlclient-dev libsasl2-dev libldap2-dev libssl-dev libxml2-dev libxslt1-dev \\\nlibxmlsec1-dev libffi-dev pkg-config apt-transport-https virtualenv build-essential  
apt-get install nodejs
```

Running 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>

```
curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | apt-key add -  
echo "deb https://dl.yarnpkg.com/debian/ stable main" | tee /etc/apt/sources.list.d/yarn.list  
apt-get update -y  
apt-get install -y yarn
```

```
git clone https://github.com/ngoduykhanh/PowerDNS-Admin.git /opt/web/powerdns-admin  
cd /opt/web/powerdns-admin  
cp config_template.py config.py  
mysql -u root -p
```

```
CREATE DATABASE pda;  
GRANT ALL PRIVILEGES ON pda.* TO 'pda'@'127.0.0.1' IDENTIFIED BY 'changeme';  
FLUSH PRIVILEGES;  
quit
```

```
nano /opt/web/powerdns-admin/config.py
```

```
BIND_ADDRESS = '0.0.0.0'
```

```
virtualenv -p python3 flask  
source ./flask/bin/activate  
pip install -r requirements.txt  
pip install python-dotenv  
export FLASK_APP=app/__init__.py  
flask db migrate -m "Init DB"  
flask db upgrade
```

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

```
/etc/init.d/pdns restart  
/etc/init.d/pdns status
```

```
nano /etc/systemd/system/powerdns-admin.service
```

```
[Unit]
Description=PowerDNS-Admin
After=network.target

[Service]
User=root
Group=root
WorkingDirectory=/opt/web/powerdns-admin
#ExecStart=/opt/web/powerdns-admin/flask/bin/gunicorn --workers 2 --bind unix:/opt/web/powerdns-admin/powerdns-admin.sock app:app
ExecStart=/opt/web/powerdns-admin/flask/bin/gunicorn --workers 2 --bind 0.0.0.0:9191 app:app

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

```
systemctl daemon-reload
systemctl start powerdns-admin
systemctl enable powerdns-admin
```

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

```
api=yes
api-key=changeme
```

```
/etc/init.d/pdns restart
```

Bonus

Install DNSdist

Встановлення та налаштування балансувальника dnsdist

Dnsdist - це високопродуктивний DNS-, DoS- та abuse балансувальник. Основне його завдання полягає у маршрутизації трафіку на найкращий сервер, що забезпечує максимальну продуктивність для дозволених користувачів, у той час як відбувається шунтування або блокування шкідливого трафіку.



Має величезну кількість фічів:

- Фільтрувати трафік (з ядра)
- Перевіряти прямий трафік з консолі
- Затримувати та обмежувати швидкість поганих запитів
- Інтелектуальне балансування навантаження
- Обмеження QPS та ін.

Встановимо пакет dnsdist.

```
apt-get update -y  
apt-get install -y dnsdist
```

Відкриваємо конфігураційний файл

```
nano /etc/dnsdist/dnsdist.conf
```

і наводимо до такого вигляду:

```
setLocal('127.0.0.1')  
addLocal('ANOTHER_IP')  
addLocal('ANOTHER_IPV6_IP')
```

```
setACL({'0.0.0.0/0'}) -- Allow all IPs access
newServer({address='127.0.0.1:5300', pool='auth'})
newServer({address='127.0.0.1:5301', pool='recursor'})
recursive_ips = newNMG()
recursive_ips:addMask('127.0.0.1/32')
recursive_ips:addMask('192.168.2.0/23')
addAction(NetmaskGroupRule(recursive_ips), PoolAction('recursor'))
addAction(AllRule(), PoolAction('auth'))
```

Якщо хочемо відкрити рекурсію для всіх, то прибираємо всі правила і додаємо recursive_ips:addMask('0.0.0.0/0').



УВАГА! У такому режимі є можливість DDoS-атаки!

Підправимо конфіг рекурсора:

```
nano /etc/powerdns/recursor.d/recursor.local.conf
```

```
local-address=127.0.0.1
local-port=5301
```

Підправимо конфіг auth:

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

```
local-address=127.0.0.1
local-port=5300
```

```
/etc/init.d/pdns-recursor restart
/etc/init.d/pdns restart
```

<https://stat.ripe.net/widget/dns-check>

Додаємо сервіс dnsmasq в автозавантаження та перезапускаємо:

```
systemctl enable dnsmasq
systemctl start dnsmasq
```

Нотатки

History

```
/sbin/usermod -aG sudo method
apt-get update -yq && apt-get upgrade -yq
sync
apt-get install libperl-dev gcc make gnupg net-tools sudo wget curl git mc qemu-guest-agent
apt-get install ufw
nano /etc/default/ufw
sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo ufw allow ssh
sudo ufw enable
sudo ufw reload
sudo ufw status verbose
sh -c 'echo "deb http://download.webmin.com/download/repository sarge contrib" >
/etc/apt/sources.list.d/webmin.list'
wget http://www.webmin.com/jcameron-key.asc
apt-key add jcameron-key.asc
apt update
apt install -y webmin
nano /etc/webmin/miniserv.conf
systemctl restart webmin
sudo ufw allow 8000/tcp
sudo ufw reload
```

```
apt-get install -y mariadb-server libmariadbclient-dev libmariadbclient-dev-compat
systemctl status mariadb
sudo mysql_secure_installation
sudo mysql
apt-get install -yq apache2
sudo a2enmod ssl
sudo a2enmod rewrite
sudo a2enmod suexec
sudo a2enmod include
systemctl restart apache2
systemctl status apache2
sudo ufw allow http
sudo ufw allow https
sudo ufw reload
sudo apt -y install wget php php-cgi php-mysqli php-pear php-mbstring php-gettext libapache2-mod-php php-common
php-phpseclib php-mysql
cd /tmp
DATA="$(wget https://www.phpmyadmin.net/home_page/latest.txt -q -O-)"
URL="$(echo $DATA | cut -d ' ' -f 3)"
VERSION="$(echo $DATA | cut -d ' ' -f 1)"
wget https://files.phpmyadmin.net/phpMyAdmin/${VERSION}/phpMyAdmin-${VERSION}-all-languages.tar.gz
tar xvf phpMyAdmin-${VERSION}-all-languages.tar.gz
rm *.tar.gz
sudo mv phpMyAdmin-* /usr/share/phpmyadmin
sudo mkdir -p /var/lib/phpmyadmin/tmp
sudo chown -R www-data:www-data /var/lib/phpmyadmin
sudo mkdir /etc/phpmyadmin/
sudo cp /usr/share/phpmyadmin/config.sample.inc.php /usr/share/phpmyadmin/config.inc.php
nano /usr/share/phpmyadmin/config.inc.php
nano /etc/apache2/conf-enabled/phpmyadmin.conf
systemctl reload apache2
systemctl restart apache2
sudo apt-get install pdns-recursor dnsmdist pdns-server pdns-backend-mysql dnsutils -yq
sudo apt install python3-dev
```

```
sudo apt install -y libsasl2-dev libldap2-dev libssl-dev libxml2-dev libxslt1-dev libxmlsec1-dev libffi-dev pkg-  
config apt-transport-https virtualenv build-essential  
curl -sL https://deb.nodesource.com/setup_10.x | bash -  
apt install -y nodejs  
sudo apt-get install gcc g++ make  
curl -sL https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -  
echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee /etc/apt/sources.list.d/yarn.list  
sudo apt-get update && sudo apt-get install yarn  
git clone https://github.com/ngoduykhanh/PowerDNS-Admin.git /opt/web/powerdns-admin  
cd /opt/web/powerdns-admin  
virtualenv -p python3 flask  
source ./flask/bin/activate  
pip install -r requirements.txt  
cp ./powerdnsadmin/default_config.py ./powerdnsadmin/config.py  
nano ./powerdnsadmin/config.py  
cp ./powerdnsadmin/config.py ./configs/config.py  
__import__ db dump and users  
export FLASK_CONF=./powerdnsadmin/config.py  
export FLASK_CONF=./configs/config.py  
export FLASK_APP=powerdnsadmin/__init__.py  
flask db upgrade  
yarn install --pure-lockfile  
flask assets build  
sudo ufw allow 9191/tcp  
sudo ufw reload  
./run.py
```

WSGI

```
apt install apache2-dev
pip3 install mod-wsgi
mod_wsgi-express install-module > /etc/apache2/mods-available/wsgi.load
a2enmod wsgi
sudo a2enmod wsgi
systemctl restart apache2
nano /etc/apache2/sites-available/powerdns-admin.conf
mc
nano /etc/apache2/sites-available/powerdns-admin.conf
nano /opt/web/powerdns-admin/powerdnsadmin.wsgi
sudo a2dissite 000-default.conf
sudo a2ensite powerdns-admin.conf
sudo apache2ctl configtest
nano /etc/apache2/sites-available/powerdns-admin.conf
sudo apache2ctl configtest
nano /etc/apache2/sites-available/powerdns-admin.conf
sudo apache2ctl configtest
sudo apache2ctl configtest
systemctl restart apache2
systemctl status apache2
nano /etc/apache2/sites-available/powerdns-admin.conf
nano /etc/apache2/sites-available/powerdns-admin.conf
nano /etc/apache2/sites-available/powerdns-admin.conf
nano /etc/apache2/sites-available/powerdns-admin.conf
sudo apache2ctl configtest
systemctl restart apache2
mc
nano /opt/web/powerdns-admin/powerdnsadmin.wsgi
systemctl restart apache2
mc
sudo python /opt/web/powerdns-admin/powerdnsadmin.wsgi
/opt/web/powerdns-admin/powerdnsadmin.wsgi
sudo apt-get install uwsgi uwsgi-plugin-python
```

```
uwsgi
uwsgi /opt/web/powerdns-admin/powerdnsadmin.wsgi
wsgi /opt/web/powerdns-admin/powerdnsadmin.wsgi
uwsgi powerdnsadmin.wsgi
sudo vim /etc/systemd/system/powerdns-admin.service
nano /etc/systemd/system/powerdns-admin.service
nano /etc/systemd/system/powerdns-admin.service
nano /etc/systemd/system/powerdns-admin.service
nano /etc/systemd/system/powerdns-admin.service
nano /etc/systemd/system/powerdns-admin.service
systemctl daemon-reload
systemctl start powerdns-admin
systemctl enable powerdns-admin
sudo ufw allow 9191/tcp
sudo ufw reload
systemctl status powerdns-admin
nano /etc/systemd/system/powerdns-admin.service
/usr/bin/gunicorn-3.6 --workers 4 --log-level info --pid /run/powerdns-admin/pid --bind unix:/run/powerdns-
admin/socket "powerdnsadmin:create_app(config='config.py')"
/opt/web/powerdns-admin/flask/bin/gunicorn --workers 2 --bind 0.0.0.0:9191
"powerdnsadmin:create_app(config='config.py')"
[Unit]
Description=PowerDNS-Admin
Requires=powerdns-admin.socket
Wants=network.target
After=network.target mysqld.service slapd.service mariadb.service
[Service]
PIDFile=/run/powerdns-admin/pid
User=www-data
Group=www-data
WorkingDirectory=/opt/web/powerdns-admin
#ExecStart=/opt/web/powerdns-admin/flask/bin/gunicorn --workers 4 --log-level info --pid /run/powerdns-admin/pid
--bind unix:/run/powerdns-admin/socket "powerdnsadmin:create_app(config='config.py')"
ExecStart=/opt/web/powerdns-admin/flask/bin/gunicorn --workers 4 --log-level info --pid /run/powerdns-admin/pid -
```

```
-bind 0.0.0.0:9191 "powerdnsadmin:create_app(config='config.py')"  
ExecReload=/bin/kill -s HUP $MAINPID  
ExecStop=/bin/kill -s TERM $MAINPID  
PrivateTmp=true  
Restart=on-failure  
RestartSec=10  
StartLimitInterval=0  
[Install]  
nano /etc/systemd/system/powerdns-admin.service  
systemctl daemon-reload  
systemctl disable powerdns-admin  
systemctl enable powerdns-admin  
systemctl restart powerdns-admin  
nano /etc/systemd/system/powerdns-admin.socket  
systemctl restart powerdns-admin  
d /run/powerdns-admin 0755 www-data www-data -  
chmod /run/powerdns-admin 0755 www-data www-data -  
chmod /run/powerdns-admin 0755 www-data www-data -  
chown www-data:www-data /run/powerdns-admin  
nano /etc/tmpfiles.d/powerdns-admin.conf  
sudo systemctl daemon-reload; sudo systemctl start powerdns-admin.socket; sudo systemctl enable powerdns-  
admin.socket  
systemctl restart powerdns-admin  
systemctl status powerdns-admin  
nano /etc/systemd/system/powerdns-admin.socket  
nano /etc/systemd/system/powerdns-admin.service  
systemctl disable powerdns-admin  
systemctl enable powerdns-admin  
systemctl restart powerdns-admin  
systemctl status powerdns-admin  
systemctl status powerdns-admin  
nano /etc/systemd/system/powerdns-admin.service  
systemctl disable powerdns-admin  
systemctl enable powerdns-admin
```

```
systemctl restart powerdns-admin
systemctl status powerdns-admin
nano /etc/systemd/system/powerdns-admin.service
systemctl restart powerdns-admin
systemctl status powerdns-admin
/opt/web/powerdns-admin/flask/bin/gunicorn --workers 4 --log-level info --pid /run/powerdns-admin/pid --bind
0.0.0.0:9191 "powerdnsadmin:create_app(config='config.py')"
nano /etc/systemd/system/powerdns-admin.service
systemctl disable powerdns-admin
sudo systemctl daemon-reload;
systemctl restart powerdns-admin
systemctl enable powerdns-admin
systemctl restart powerdns-admin
systemctl stop powerdns-admin
nano /etc/systemd/system/powerdns-admin.service
systemctl disable powerdns-admin
sudo systemctl daemon-reload;
systemctl enable powerdns-admin
systemctl restart powerdns-admin
systemctl disable powerdns-admin
nano /etc/systemd/system/powerdns-admin.service
nano /etc/systemd/system/powerdns-admin.service
systemctl daemon-reload
systemctl start powerdns-admin
systemctl enable powerdns-admin
systemctl start powerdns-admin
systemctl status powerdns-admin
systemctl stop powerdns-admin
systemctl stop powerdns-admin.socket
systemctl stop powerdns-admin.socket
systemctl stop powerdns-admin
/opt/web/powerdns-admin/flask/bin/gunicorn --pid /run/powerdns-admin/pid --bind 0.0.0.0:9191
powerdnsadmin:create_app()
/opt/web/powerdns-admin/flask/bin/gunicorn --bind 0.0.0.0:9191 powerdnsadmin:create_app()
```

```
/opt/web/powerdns-admin/flask/bin/gunicorn --bind 0.0.0.0:9191 powerdnsadmin:create_app()  
/opt/web/powerdns-admin/flask/bin/gunicorn --bind 0.0.0.0:9191 "powerdnsadmin:create_app()"   
/opt/web/powerdns-admin/flask/bin/gunicorn --pid /run/powerdns-admin/pid --bind 0.0.0.0:9191  
"powerdnsadmin:create_app()"  
nano /etc/systemd/system/powerdns-admin.service  
systemctl daemon-reload  
systemctl stop powerdns-admin.socket  
systemctl stop powerdns-admin  
systemctl enable powerdns-admin  
systemctl start powerdns-admin  
systemctl status powerdns-admin  
nano /etc/systemd/system/powerdns-admin.service  
systemctl daemon-reload  
systemctl enable powerdns-admin  
systemctl restart powerdns-admin  
systemctl status powerdns-admin  
nano /etc/apache2/sites-available/powerdns-admin.conf  
sudo apache2ctl configtest  
mod_proxy  
sudo a2enmod proxy  
systemctl restart apache2  
sudo apache2ctl configtest  
nano /etc/apache2/sites-available/powerdns-admin.conf  
sudo apache2ctl configtest  
systemctl restart apache2  
nano /etc/apache2/sites-available/powerdns-admin.conf  
sudo apache2ctl configtest  
systemctl restart apache2  
history
```

[Установка DNSCrypt-сервера](#)

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

new version <https://kifarunix.com/easily-install-and-setup-powerdns-admin-on-ubuntu-20-04/>

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

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

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

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

```
nano /etc/apt/preferences.d/pdns
```

```
Package: pdns-*
Pin: origin repo.powerdns.com
Pin-Priority: 600
```

```
curl https://repo.powerdns.com/CBC8B383-pub.asc | sudo apt-key add - &&
sudo apt-get update &&
sudo apt-get install pdns-recursor
```

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

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

SNMP

<https://jira.observium.org/browse/OBS-3914?attachmentSortBy=dateTime> .1.3.6.1.4.1.43315

```
apt-get install snmpd
```

```
#apt-add-repository non-free  
#apt-get update  
#apt install snmp-mibs-downloader
```

```
nano /etc/snmp/snmpd.conf
```

```
master agentx  
agentxperms 0700 0700 _dnsmdist _dnsmdist  
rocommunity dnsmdist42
```

```
systemctl restart snmpd
```

DNSDIST BLACK LIST

<https://github.com/enilfodne/dnsmdist-adblock/blob/master/dagg/dagg.lua>

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

Permanent link:
https://ndp.pp.ua/doku.php/debian/pdns_recursor?rev=1665184753

Last update: **2022/10/07 23:19**

