

NoDeny Next

Встановлення NoDeny Next на Debian



Установка NoDeny+ на Debian мені потрібна, т.к. всі свої сервери я тримаю на віртуалках, а як відомо, у FreeBSD проблеми зі швидкістю запису на диск під KVM QEMU!

До того ж, я не використовую файрвол білінгу, який заточений під FreeBSD

NoDeny завантаження

на момент написання статті є 3 репозиторії:

- демOVERсія `svn://nodelny-plus.com.ua/nodelnyfree`
- стабільна `svn://nodelny-plus.com.ua/release`
- нова гілка `svn://nodelny-plus.com.ua/release/next`



я буду використовувати останній, оскільки у мене є підписка

і так, завантажуюмо білінг

```
cd /usr/local  
svn co svn://nodelny-plus.com.ua/release/next nodelny
```

Ставимо пакети для білінгу

```
apt-get install -y libcrypt-rijndael-perl libdbd-mysql-perl libjson-perl libnet-arp-perl libcpanel-json-xs-perl  
liblwp-protocol-https-perl libcgi-pm-perl
```

liqpay\pingserver\та інші

```
apt-get install -y libxml-simple-perl libdigest-sha-perl liblwp-useragent-determined-perl libcrypt-ssleay-perl
```

для telegram

```
cpanm -n WWW::Telegram::BotAPI
```

для ronnmon

```
apt-get install -y snmp libnet-telnet-cisco-perl libparallel-forkmanager-perl  
cpanm -n Net::SNMP::Util
```

Web Server

Встановлення Apache

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

Включаємо потрібні нам модулі для опача:

```
sudo a2enmod ssl rewrite suexec include cgid
```

Створимо конфіг замість дефолтного

```
nano /etc/apache2/sites-available/nodeny.conf
```

з таким вмістом

```
<VirtualHost *:80>
    ServerAdmin admin@example.com
    ServerName example.com
    ServerAlias www.example.com
    DocumentRoot /var/www/nodeny
    <FilesMatch "\.(pl|cgi|shtml|phtml|php)$">
        AddHandler cgi-script .pl
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
        CGIPassAuth On
    </FilesMatch>
    <IfModule alias_module>
        ScriptAlias /cgi-bin/ "/usr/lib/cgi-bin/"
        #ErrorDocument 404 "/cgi-bin/stat.pl"
        DirectoryIndex "/cgi-bin/stat.pl"
    </IfModule>
    <IfModule !alias_module>
        DirectoryIndex index.html
    </IfModule>
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Створимо симлінки для білінгу

```
rm -fR /usr/lib/cgi-bin
ln -s /usr/local/nodeny/htdocs/ /var/www/nodeny
ln -s /usr/local/nodeny/cgi-bin/ /usr/lib/cgi-bin
```

Виставимо права

```
perl /usr/local/nodeny/install.pl -w=www-data
```

Застосовуємо свій конфіг

```
sudo a2ensite nodeny.conf
sudo a2dissite 000-default.conf
```

перевіримо помилки конфігурації

```
sudo apache2ctl configtest
```

Ви повинні побачити наступний результат:

```
Syntax OK
```

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/>

База даних



Я не використовуватиму MySQL, оскільки з білінгом чудово працює і Percona server for mysql 8! [Джерело](#)
А на Percona можна буде підняти Galera Cluster!

Percona MySQL Server 8

Встановлення Percona MySQL Server 8 на Debian 12

Отримуємо репозиторій і встановлюємо його

```
apt install curl
curl -O https://repo.percona.com/apt/percona-release_latest.generic_all.deb
apt install gnupg2 lsb-release -y
apt install ./percona-release_latest.generic_all.deb
```

Встановлюємо сервер

Для версії percona 8.0

```
percona-release enable-only ps-80 release
apt update
apt install percona-server-server percona-server-client
```

В процесі встановлення буде запропоновано встановити пароль для користувача root, я на цьому етапі встановлюю пароль також root, після чого його зміню сам.

* Percona Server is distributed with several useful UDF (User Defined Function) from Percona Toolkit. * Run the following commands to create these functions:

```
mysql -e "CREATE FUNCTION fnv1a_64 RETURNS INTEGER SONAME 'libfnv1a_udf.so'"
mysql -e "CREATE FUNCTION fnv_64 RETURNS INTEGER SONAME 'libfnv_udf.so'"
mysql -e "CREATE FUNCTION murmur_hash RETURNS INTEGER SONAME 'libmurmur_udf.so'"
```

* See http://www.percona.com/doc/percona-server/8.0/management/udf_percona_toolkit.html for more details

При необхідності встановлюємо утиліти **percona-toolkit**

```
percona-release enable tools release  
apt update  
apt install percona-toolkit percona-xtrabackup-80 sysbench
```

Стартуємо сервер MySQL

```
root@db ~ # service mysql start
```

Datadir

Якщо є потреба в зміні розміщення каталогу даних на примотований диск, даний абзац допоможе в цьому.

Припустимо, що ми під'єднали додатковий диск та змонтували його в `/mysql_data`

```
mkdir -p /mysql_data/data/  
mkdir -p /mysql_data/logs/  
  
[mysqld]  
datadir                = /mysql_data/data/  
# logs  
log_bin                = /mysql_data/logs/binlog  
relay_log              = /mysql_data/logs/relaylog  
relay_log_index        = /mysql_data/logs/relaylog.index  
binlog_expire_logs_seconds = 604800 # 7 days  
max_binlog_size        = 100M  
relay_log_recovery     = on  
master_info_repository = 'TABLE'  
relay_log_info_repository = 'TABLE'
```

```
systemctl stop mysql
```

```
rsync -auxHAXSv --chown=mysql:mysql /var/lib/mysql/* /mysql_data/data/  
systemctl start mysql
```

14/03/2024 12:42 · Method

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

Створимо базу даних nodeny

```
create database nodeny;  
use nodeny;  
source /usr/local/nodeny/bill.sql;
```

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 включения <https://www.8host.com/blog/sozдание-sertifikata-lets-encrypt-dlya-apache-v-debian-10/>

05/03/2022 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'] = 'H20xcGXflSd8JwrwVlh6KW6s2rER63i';
```

```
//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-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.

05/03/2022 01:19 · Method

Webmin

For simply management OS

```
wget https://raw.githubusercontent.com/webmin/webmin/master/setup-repos.sh  
sh setup-repos.sh -f  
apt-get install --install-recommends webmin
```

Edit webmin config

```
nano /etc/webmin/miniserv.conf
```

Setting port & disable ssl via change next

```
ssl=0
```

```
port=8000
```

Apply changes

```
systemctl restart webmin
```

Open firewall

NFTables

```
nft add rule inet filter input ct state new tcp dport 8000 counter accept comment "WEBMIN"
```

Open [http://\[Host_ip\]:8000/](http://[Host_ip]:8000/) in your browser & log in with your system credentials – username & password.

05/03/2022 01:27 · Method

FreeRadius



Даний варіант установки не з офіційної документації і розроблений для нашої мережі, але може бути портований і в інші мережі! Він не має MySQL модуля, але є потужний модуль PERL, який дозволяє робити запити до бази набагато зручніше в плані маніпуляцій. На офіційному сайті FreeRadius сказано, що rlm_perl продуктивніше rlm_sql. Використання цього варіанта ви робите на свій страх і ризик :).

```
apt-get install freeradius freeradius-mysql freeradius-utils
```

Конфігурація

Видалимо зайве, попередньо зробивши бекап



```
cp -r /etc/freeradius/3.0/ /etc/freeradius/3.0.orig/  
rm /etc/freeradius/3.0/sites-enabled/*  
rm /etc/freeradius/3.0/mods-enabled/eap  
rm /etc/freeradius/3.0/mods-enabled/ntlm_auth  
rm /etc/freeradius/3.0/mods-enabled/mschap
```

```
nano /etc/freeradius/3.0/mods-config/attr_filter/access_reject
```

поправимо тут

```
DEFAULT  
# EAP-Message = * ANY,  
State =* ANY,  
Message-Authenticator =* ANY,  
Error-Cause =* ANY,  
Reply-Message =* ANY,  
MS-CHAP-Error = * ANY,  
Proxy-State =* ANY
```

Створимо основний конфіг

```
nano /etc/freeradius/3.0/sites-enabled/nodeny
```

з таким вмістом:

</etc/freeradius/3.0/sites-enabled/nodeny>

```
server nodeny {  
    listen {  
        type = auth  
        ipaddr = *  
        port = 1812  
    }  
}
```

```
listen {
    type = acct
    ipaddr = *
    port = 1813
}
authorize {
    detail
    # preprocess
    files
    perl
}

authenticate {
    Auth-Type PAP {
        pap
    }
    # Auth-Type CHAP {
    #     chap
    # }
    # Auth-Type MSCHAP {
    #     mschap
    # }
    Auth-Type Perl {
        perl
    }
}

preacct {
    acct_unique
    preprocess
}

accounting {
    attr_filter.accounting_response
```

```
    perl
}

session {
    radutmp
}

post-auth {
    perl
}
}
```

Також, за бажанням, можна завантажити конфіг сервера статистики

```
ln -s /etc/freeradius/3.0/sites-available/status /etc/freeradius/3.0/sites-enabled/status
```

Опишемо локальний сервер доступу для тестів, сюди ж вписуємо всі свої БРАСи за аналогією

```
echo '' > /etc/freeradius/3.0/clients.conf
nano /etc/freeradius/3.0/clients.conf
```

[/etc/freeradius/3.0/clients.conf](#)

```
client 127.0.0.1 {
    ipaddr = 127.0.0.1
    secret = hardpass5
    shortname=NoDenyDB
    nastype = cisco
}
```

```
echo '' > /etc/freeradius/3.0/users
nano /etc/freeradius/3.0/users
```

[/etc/freeradius/3.0/users](#)

```
DEFAULT Auth-Type = Perl
Fall-Through = yes
```

```
echo '' > /etc/freeradius/3.0/mods-enabled/perl
nano /etc/freeradius/3.0/mods-enabled/perl
```

[/etc/freeradius/3.0/mods-enabled/perl](#)

```
perl {
    filename = /usr/local/nodeny/nod/_radius.pl
}
```

Permission denied

Оскільки freeradius використовує модулі з директорії білінгу, при запуску можемо отримувати помилку, схожу на наступну

```
'Can't open perl script "/usr/local/nodeny/nod/_radius.pl": Permission denied'
'rlm_perl: perl_parse failed: /usr/local/nodeny/nod/_radius.pl not found or has syntax errors'
'/etc/freeradius/3.0/mods-enabled/perl[1]: Instantiation failed for module "perl"'
```

Щоб це виправити, необхідно змінити користувача запуску freeradius, щоб freeradius міг використовувати файли за межами каталогу своїх конфігурацій

```
nano /etc/freeradius/3.0/radiusd.conf
```


У секції 'security' коментуємо користувача та групу

```
#user = freerad  
#group = freerad
```

Також перевизначимо системний юніт

```
systemctl edit --full freeradius
```

В ньому потрібно закоментувати рядки де вказано користувача і групу, від імені яких відбувається запуск.

```
#User=freerad  
#Group=freerad
```

```
systemctl daemon-reload  
service freeradius restart
```

Перевіряємо

```
journalctl -xau freeradius.service
```

Маємо побачити щось схоже на наступне

```
freeradius[240578]: Copyright (C) 1999-2022 The FreeRADIUS server project and contributors  
freeradius[240578]: There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A  
freeradius[240578]: PARTICULAR PURPOSE  
freeradius[240578]: You may redistribute copies of FreeRADIUS under the terms of the  
freeradius[240578]: GNU General Public License  
freeradius[240578]: For more information about these matters, see the file named COPYRIGHT  
freeradius[240578]: Starting - reading configuration files ...  
freeradius[240578]: Debugger not attached  
freeradius[240578]: Perl version: 5.36.0  
freeradius[240578]: Creating attribute Unix-Group
```

```
freeradius[240578]: rlm_detail (auth_log): 'User-Password' suppressed, will not appear in detail output
freeradius[240578]: Compiling Autz-Type Status-Server for attr Autz-Type
freeradius[240578]: Compiling Auth-Type PAP for attr Auth-Type
freeradius[240578]: Compiling Auth-Type Perl for attr Auth-Type
freeradius[240578]: radiusd: ##### Skipping IP addresses and Ports #####
freeradius[240578]: Configuration appears to be OK
```

Завершение установки

```
systemctl enable freeradius
systemctl restart freeradius
systemctl status freeradius
```

Не забуваємо налаштувати файрвол

Перевіряємо використання портів

```
ss -alun4 | grep -E '1812|1813'
```

Бачимо

```
UNCONN 0      0      127.0.0.1:18121      0.0.0.0:*
UNCONN 0      0      0.0.0.0:1812        0.0.0.0:*
UNCONN 0      0      0.0.0.0:1813        0.0.0.0:*
```

https://wiki.mikbill.pro/billing/bez_deneg

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

Permanent link:
<https://ndp.pp.ua/doku.php/debian/nodeny?rev=1733819599>

Last update: **10/12/2024 08:33**

