

AIO install

Швидка установка "Все в одному"



Весь процес установки та налаштування за даною інструкцією від початку і до кінця займає близько півгодини! Мій рекорд 22 хвилини :)

Процедура оновлення всіх пакетів відбувається швидко за командою

```
pkg upgrade
```

Встановлення системи

Підготовляємо дистрибутив FreeBSD x.x **Release** Записуємо завантажений дистрибутив на диск і сміливо приступаємо:



Все перевірено на FreeBSD v13.0, v12.1, v11.0-2, v10.1-3 та v9.3!

1. Під час установки, після вибору локалі, обов'язково вибираємо **SRC** (у майбутньому може знадобиться для перескладання ядра), решта

- за бажанням (я б не ставив тільки порти та доки, і так їх оновлювати потрібно)
2. Далі вибрати метод розбики диска за промовчанням - Auto (UFS)
 3. Тому що розбивати: весь диск або розділ (припустимо весь диск)
 4. установка запропонує найкращий варіант, де

під завантажувальний сектор = 512 кб
розділ підкачування свап = 1 гб (2*рам але не більше 1 гб)
кореневий розділ / = все інше

5. тиснемо Finish ⇒ Commit = пішла установка FreeBSD
6. потім вводимо пароль для рут-користувача
7. потім налаштуємо доступ до інтернету
8. після чого вибираємо часовий пояс
9. далі можна вибрати які послуги вам потрібні, **SSHD** обов'язково
10. потім потрібно додати локального користувача помістивши його до групи **wheel**
11. Установка системи завершена Exit ⇒ No ⇒ Reboot



важливо витягнути інсталяційний диск під час перезавантаження, щоб не розпочати інсталяцію заново



Після перезавантаження сервера, підключаємося до нього через SSH, щоб легше копі-пастити

Налаштування завантажувача

Для прискорення завантаження сервера відключимо таймер завантажувального меню, а також саме меню

```
echo autoboot_delay="-1" >> /boot/loader.conf
echo beastie_disable="YES" >> /boot/loader.conf
```

Початкова конфігурація

```
sysrc fsck_y_enable=YES
sysrc background_fsck=NO
sysrc sshd_enable=YES
sysrc firewall_enable=YES
sysrc gateway_enable=YES

ee /etc/rc.conf
```

приводимо до вигляду

```
hostname="nodeny.com.ua"
defaultrouter="1.2.3.1"
ifconfig_fxp0="inet 10.0.0.1 netmask 255.255.255.0"
ifconfig_fxp1="inet 1.2.3.4 netmask 255.255.255.0"
ifconfig_lo0_alias0="inet 1.1.1.1 netmask 255.255.255.255"
fsck_y_enable="YES"
background_fsck="NO"
sshd_enable="YES"
firewall_enable="YES"
gateway_enable="YES"
```

де



- background_fsck="NO" відключає перевірку дисків у бекграунді та робить при старті
- fsck_y_enable="YES" каже "yes" на всі запитання під час перевірки.
- 1.1.1.1 - віртуальний ір, який автор налаштовує усім своїх серверах на інтерфейсі lo0. Цей ір клієнти вказують у своїх



- настройках як dns сервер, сервер авторизації і т.д.
- Інтерфейс fxp1 дивиться на шлюз вищого провайдера з ip 1.2.3.1.
- Інтерфейс fxp0 дивиться у мережу клієнтів.

Відключаємо підтримку X11 та GUI

```
echo 'WITHOUT_X11="yes"' >> /etc/make.conf
echo 'WITHOUT_GUI="yes"' >> /etc/make.conf
```

Ставимо PKGNG



З вересня 2014 розробники FreeBSD замінили застарілий пакетний менеджер на новий - PKGNG (package new generation). З базової системи були видалені утиліти pkg_* (такі, як: pkg_install, pkg_add, pkg_create, pkg_delete, pkg_info, pkg_updating та pkg_version).

[Посібник FreeBSD:PKGNG](#)

установка PKGNG

```
pkg bootstrap -f
```

Перекладаємо базу даних портів працювати з PKGNG, відключаємо бекап старої бази, створюємо конфіг PKGNG

```
echo 'WITH_PKGNG="yes"' >> /etc/make.conf
echo 'daily_backup_pkgdb_enable="NO"' >> /etc/periodic.conf
cp /usr/local/etc/pkg.conf.sample /usr/local/etc/pkg.conf
```

За замовчуванням підключений квартальний репозиторій пакетів, але я вмикаю останній

```
cp /etc/pkg/FreeBSD.conf /etc/pkg/FreeBSD.conf.old
```

```
sed -i -p "s/quarterly/latest/" /etc/pkg/FreeBSD.conf  
ee /etc/pkg/FreeBSD.conf  
url: "pkg+http://pkg.FreeBSD.org/${ABI}/latest",
```

оновимо базу доступних пакетів

```
pkg update -f
```

Ставимо Пакети

```
pkg install perl5 p5-App-cpanminus p5-Crypt-Rijndael p5-JSON p5-JSON-XS p5-LWP-Protocol-https p5-CGI \  
bash subversion apache24 mc-nox11 mysql80-server
```

```
ln -s /usr/local/bin/perl /usr/bin/perl > /dev/null
```

```
cpanm DBD::mysql
```

Завантажуємо NoDeny Plus

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

Налаштування NAT (pf)

```
echo pf_load=YES >> /boot/loader.conf  
echo pf_enable=\"YES\" >> /etc/rc.conf  
cp /usr/local/nodeny/etc/pf.conf /etc/
```

```
ee /etc/pf.conf
```



- вписати в ext_if = "fxp1" свій інтерфейс, який у вас дивиться в інет
- залишити правило тієї мережі, яка дивиться в локалку



Важливо, щоб адреси зовнішньої та внутрішньої мереж не були однаковими!

Налаштування збір статистики трафіку (ipcad)

```
echo ipfw_load=\"YES\" >> /boot/loader.conf
echo ipdivert_load=\"YES\" >> /boot/loader.conf
echo dumynet_load=\"YES\" >> /boot/loader.conf
echo ipcad_enable=\"YES\" >> /etc/rc.conf
cp /usr/local/nodeny/etc/ipcad.conf /usr/local/etc/
ipcad -d
```

Налаштування firewall

Фаєрвол із блокуванням неавторизованих клієнтів

```
cp /usr/local/nodeny/etc/rc.firewall /etc/
ee /etc/rc.firewall
```





Редагуємо rc.firewall, міняємо зовнішній інтерфейс
ifOut='fxp1' на той, що у вас дивиться в інет

Налаштування mysql

```
sysrc mysql_enable=yes
sysrc mysql_optfile=/usr/local/etc/mysql/my.cnf

mkdir /var/db/mysql_tmpdir/
chown mysql /var/db/mysql_tmpdir/

service mysql-server start
```

Підключаємось до mysql і встановимо пароль руту

```
mysql -u root
# замість hardpass ваш пароль
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'hardpass', `root`@`localhost` PASSWORD EXPIRE NEVER;
exit;
```

Підключаємось до mysql з паролем

```
# замість hardpass ваш пароль
mysql -u root --password='hardpass'
ALTER USER `root`@`localhost` IDENTIFIED BY `hardpass`, `root`@`localhost` PASSWORD EXPIRE NEVER;
```

Створимо базу даних nodeny

```
create database nodeny;
```

```
use nodeny;
source /usr/local/nodeny/bill.sql;
exit;
```

Tuning

```
pkg install mysqltuner
mysqltuner --user root --pass hardpass --color
```

Налаштування apache24

```
sysrc apache24_enable=yes
echo /sbin/kldload accf_http >> /etc/rc.local
/sbin/kldload accf_http
```

```
echo "ServerName 0.0.0.0:80" >> /usr/local/etc/apache24/Includes/httpd.conf
echo "LoadModule cgi_module libexec/apache24/mod_cgi.so" >> /usr/local/etc/apache24/Includes/httpd.conf
```

Створимо симлінк на веб-скрипти NoDeny

```
rm -fR /usr/local/www/apache24/cgi-bin
rm -fR /usr/local/www/apache24/data
ln -s /usr/local/nodeny/cgi-bin/ /usr/local/www/apache24/cgi-bin
ln -s /usr/local/nodeny/htdocs/ /usr/local/www/apache24/data
```

```
service apache24 start
```

```
mkdir /usr/local/etc/newsyslog.conf.d/
ee /usr/local/etc/newsyslog.conf.d/apache24.conf
```

```
# Apache  
/var/log/httpd-access.log www:wheel 600 10 200 * JC /var/run/httpd.pid 30  
/var/log/httpd-error.log www:wheel 600 10 200 * JC /var/run/httpd.pid 30
```

freebsd:install:certbot

Налаштування NoDeny Plus

Підключимо модулі (-x), встановимо власника та права на файли (-w), створимо стартовий конфіг (-m), створимо суперадміну (-p):

```
cd /usr/local/nodeny  
perl install.pl -m  
perl install.pl -x  
perl install.pl -w=www  
perl install.pl -p admin=12345
```

Відновлення з Бекапа

Обов'язково скопіювати в папку /usr/local/nodeny/ файли:



sat.cfg - файл з налаштуваннями відновлюваної NoDeny
backup.sql - файл бази даних, що відновлюється NoDeny

Створимо структуру бази nodeny

```
mysql --default-character-set=utf8 -u root -p  
# вводимо пароль рута  
DROP DATABASE nodeny;
```

```
CREATE DATABASE nodeny;
exit;
#source /usr/LOCAL/nodeny/backup.sql;
```

і наповнимо даними з бекапа:

```
mysql --default-character-set=utf8 -u root -p nodeny < /usr/local/nodeny/backup.sql
Enter password: # вводимо пароль рута
```

Підключимо модулі (-x), встановимо власника та права на файли (-w):

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

Запуск ядра NoDeny+

Ядро NoDeny+ працює на тлі та виконує такі функції: обробка авторизацій, збір статистики трафіку, управління послугами тощо. Конфігі знаходяться у файлах /usr/local/nodeny/kernel/*.cfg. Обов'язковий параметр run - вказує, чи буде plagin запущений при старті ядра чи ні. Якщо run встановлено в 0, то plagin можна запустити лише вказавши його у параметрах командного рядка під час запуску ядра.

Отримаємо список усіх plaginів

```
perl nokernel.pl -L
```

Навпроти назви модуля 1 означає, що модуль запускатиметься автоматично при старті ядра.

Запускаємо ядро зі стандартним набором модулів

```
perl nokernel.pl
```

Якщо на консоль не вивело жодних помилок – ядро успішно запустило всі модулі.

Запускаємо скрипт управління фаєрволом

```
perl noserver.pl -v
```

Якщо на консоль не вивело жодних помилок – ставимо в автозавантаження:

```
echo '/usr/bin/perl /usr/local/nodeny/nokernel.pl -d &' >> /etc/rc.local
echo '/usr/bin/perl /usr/local/nodeny/noserver.pl -d &' >> /etc/rc.local
```

Запускаємо скрипти на тлі

```
perl /usr/local/nodeny/nokernel.pl -d &
perl /usr/local/nodeny/noserver.pl -d &
```

Додаткові проги для зручності

Установка Webmin

```
pkg install sysutils/webmin
/usr/local/lib/webmin/setup.sh
sysrc webmin_enable=YES
service webmin start
```

не забудьте відкрити у файрволі порти

```
ee /etc/rc.firewall
замінити
${f} add 50 allow tcp from any to me 22
```

```
 ${f} add 51 allow tcp from me 22 to any
на
${f} add 50 allow tcp from any to me 22,10000
${f} add 51 allow tcp from me 22,10000 to any
```



де 10000 = порт webmin

Встановлення phpMyAdmin

```
pkg install phpMyAdmin5-php80 mod_php80 php80-iconv
```

Create directory for phpMyAdmin temp files.

```
mkdir -p /usr/local/www/phpMyAdmin/tmp
chown -R www:www /usr/local/www/phpMyAdmin/tmp
nano /usr/local/www/phpMyAdmin/config.inc.php
```

```
#Edit the file and set secret passphrase:
$cfg['blowfish_secret'] = 'H20xcGXxflSd8JwrwVlh6KW6s2rER63i';

//Configure Temp directory by add
$cfg['TempDir'] = '/usr/local/www/phpmyadmin/tmp';
```

У файлі /usr/local/etc/apache24/Includes/httpd.conf допищемо:

```
ee /usr/local/etc/apache24/Includes/httpd.conf
```

```
<IfModule dir_module>
    DirectoryIndex index.html index.htm index.php index.shtml
```

```
</IfModule>
```

У папці /usr/local/etc/apache24/Includes створюємо файл php-application.conf

```
ee /usr/local/etc/apache24/Includes/php-application.conf
```

і внесемо до нього ці рядки:

/usr/local/etc/apache24/Includes/php-application.conf

```
<FilesMatch "\.php$">
    SetHandler application/x-httpd-php
</FilesMatch>
<FilesMatch "\.phps$">
    SetHandler application/x-httpd-php-source
</FilesMatch>
```

Після цього створимо файл /usr/local/etc/php.ini з файлу php.ini-production та надамо йому необхідні права

```
cp /usr/local/etc/php.ini-production /usr/local/etc/php.ini
chmod u+w /usr/local/etc/php.ini
```

Після установки є два варіанти налаштування: як аліас або як піддомен

Налаштування phpMyAdmin як аліас

створимо файл phpmyadmin.conf

```
ee /usr/local/etc/apache24/Includes/phpmyadmin.conf
```

з таким змістом:

/usr/local/etc/apache24/Includes/phpmyadmin.conf

```
Alias /pma "/usr/local/www/phpMyAdmin/"
<Directory "/usr/local/www/phpMyAdmin/">
    Options none
    AllowOverride Limit
    Order Deny,Allow
    Require all granted
</Directory>
```

Перезапустимо apache

```
service apache24 restart
```

Налаштування phpMyAdmin як піддомен

```
ee /usr/local/etc/apache24/Includes/vhost_db.example.org.conf
```

вставляємо наступне

```
<VirtualHost *:80>
    DocumentRoot "/usr/local/www/phpMyAdmin"
    ServerName db.example.org
    ServerAlias db.local
    ErrorLog "/var/log/apache/db.example.org-error.log"
    CustomLog "/var/log/apache/db.example.org-access.log" common
<Directory "/usr/local/www/phpMyAdmin">
    AllowOverride None
    Options None
    Require all granted
</Directory>
```

```
</VirtualHost>
```



де
example.org - ваш домен
db.example.org - ваш піддомен для phpMyAdmin

також не забудьте створити файли логів, інакше apache буде дуже сваритися

```
mkdir -p /var/log/apache/
touch /var/log/apache/db.example.org-error.log
touch /var/log/apache/db.example.org-access.log
```

Перезапускаємо apache

```
service apache24 restart
```

Встановлення qemu-guest-agent

Агент віртуалізації Proxmox VE

Починаючи FreeBSD 13.0 (у 12.2 не перевіряв)

```
pkg install qemu-guest-agent

sysrc qemu_guest_agent_enable="YES"
sysrc qemu_guest_agent_flags="-d -v -l /var/log/qemu-ga.log"

service qemu-guest-agent start
```

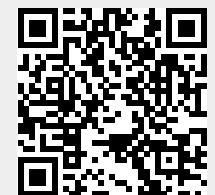
Встановлення BRAS

BRAS - Broadband Remote Access Server - у перекладі, Маршрутизатор широкосмугового віддаленого доступу. Він покликаний здійснювати авторизацію абонентів у мережі. Зазвичай складається з бази даних, Radius'a та точок доступу.

Але кожна мережа має свій вибір...

From:

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



Permanent link:

<https://ndp.pp.ua/doku.php/nodeny/fastinstall?rev=1675167222>

Last update: **2023/01/31 14:13**