

Установка ABills на Ubuntu

- [Скачивание ABills](#)
- [Настройка Apache](#)
- [Настройка MySQL](#)
- [Perl modules](#)
- [Установка RADIUS](#)
- [Постнастройка Abills](#)

Первые шаги

- [Проверка](#)
- [Дополнительно](#)



Версия системы под которую писалась инструкция [Ubuntu Server 16.04](#)

При установке выбраны только опции:

- standart system utilities
- OpenSSH Server

Работаем в консоли, переходим в режим **root**:

```
sudo su
```

Сначала обновим систему и пакеты

Перезагрузились и устанавливаем пакеты **MySQL**, **apache2** и библиотеки **perl**

```
apt-get install -yq mysql-server mysql-client libmysqlclient-dev apache2 apache2-utils libexpat1 ssl-cert
```

Скачивание ABILLS



На момент написания статьи последняя версия 0.77.77 ([Проверить](#))

```
cd /usr
wget https://sourceforge.net/projects/abills/files/abills/0.77/abills-0.77.77.tgz
tar zxvf abills-0.77.77.tgz
cp /usr/abills/libexec/config.pl.default /usr/abills/libexec/config.pl
```

Создаем недостающие каталоги и меняем права:

```
mkdir /var/log/httpd/
mkdir -p /usr/abills/var/log
touch /usr/abills/var/log/abills.log
chown -Rf www-data:www-data /usr/abills/cgi-bin
chown -Rf www-data:www-data /usr/abills/Abills/templates
chown -Rf www-data:www-data /usr/abills/backup
touch /usr/abills/var/log/sql_errors
chown nobody /usr/abills/var/log/sql_errors
chmod 666 /usr/abills/var/log/sql_errors
```

Настройка Apache

```
cp /usr/abills/misc/apache/abills_httpd.conf /etc/apache2/sites-enabled/
```

Создаем сертификат

```
/usr/abills/misc/certs_create.sh apache
```

Запускаем установленные модули для апача:

```
a2enmod ssl  
a2enmod rewrite  
a2enmod suexec  
a2enmod include  
a2enmod cgi
```

Перезапускаем apache:

```
/etc/init.d/apache2 restart
```

Настройка MySQL

```
mysql --default-character-set=utf8 -u root -p
```

Права на работу с базой

```
GRANT ALL ON abills.* TO `abills`@localhost IDENTIFIED BY "sqlpassword";  
CREATE DATABASE abills DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;  
quit;
```

Добавляем БД в MySQL

```
mysql --default-character-set=utf8 -u root -p -D abills < /usr/abills/db/abills.sql
```

Perl modules

Для работы системы нужны модули.

```
cd /usr/abills/misc/ && perl perldeps.pl apt-get -batch
```

Улучшаем себе жизнь в будущем:

```
apt-get install -y scanminus
```

Если какие-то модули не установились, можно запустить

```
scanm <>
```

Например у меня не установился Digest::SHA1, запускаю:

```
срnm Digest::SHA1
```

Установка RADIUS

устанавливаем нужные библиотеки для сборки freeradius

Установка библиотек

```
apt-get install -yq libmysqlclient-dev libgdbm3 libgdbm-dev make gcc libtalloc-dev libperl-dev libhiredis-dev
```

Проверка версии Perl

```
find /usr/lib/ | grep libperl.so
/usr/lib/x86_64-linux-gnu/libperl.so.5.22
/usr/lib/x86_64-linux-gnu/libperl.so.5.22.1
```

Делаем симлинк для того, чтобы библиотека подгрузилась во время компиляции

```
ln -s /usr/lib/x86_64-linux-gnu/libperl.so.5.22 /usr/lib/x86_64-linux-gnu/libperl.so
```

Скачиваем, компилируем и устанавливаем Freeradius версии 3.0.17

```
cd /tmp
wget ftp://ftp.freeradius.org/pub/freeradius/freeradius-server-3.0.17.tar.gz
tar zxvf freeradius-server-3.0.17.tar.gz
cd freeradius-server-3.0.17
./configure --prefix=/usr/local/freeradius --with-rlm-perl-lib-dir=/usr/lib/x86_64-linux-gnu/ \
--with-dhcp=yes --with-openssl=no > /dev/null
make && make install

ln -s /usr/local/freeradius/sbin/radiusd /usr/sbin/radiusd
```

Файл для автозапуска,- создаем и вставляем в него следующий текст

```
nano /etc/init.d/radiusd
```

Файл автостарта /etc/init.d/radiusd

```
#!/bin/sh
#
# radiusd  Start the radius daemon.
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
```

```

#   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
#   GNU General Public License for more details.
#
#   You should have received a copy of the GNU General Public License
#   along with this program; if not, write to the Free Software
#   Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA
#
#   Copyright (C) 2001-2008 The FreeRADIUS Project http://www.freeradius.org
#   chkconfig: - 58 74
#   description: radiusd is service access provider Daemon.
### BEGIN INIT INFO
# Provides: radiusd
# Required-Start: $remote_fs $network $syslog
# Should-Start: mysql radiusd
# Required-Stop: $remote_fs $syslog
# Should-Stop: radiusd
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: start and stop radiusd
# Description: radiusd is access provider service Daemon.
### END INIT INFO

prefix=/usr/local/freeradius
exec_prefix=${prefix}
sbindir=${exec_prefix}/sbin
localstatedir=/var
logdir=${localstatedir}/log/radius
rundir=/usr/local/freeradius/var/run/radiusd/
sysconfdir=${prefix}/etc
#
# If you have issues with OpenSSL, uncomment these next lines.
#
# Something similar may work for MySQL, and you may also
# have to LD_PRELOAD libz.so
#
#LD_LIBRARY_PATH=
#LD_RUN_PATH=
#LD_PRELOAD=libcrypto.so
export LD_LIBRARY_PATH LD_RUN_PATH LD_PRELOAD

RADIUSD=${sbindir}/radiusd
RADDBDIR=${sysconfdir}/radddb
RADIUS_USER='freerad'
DESC="FreeRADIUS"

#
# See 'man radiusd' for details on command-line options.
#
ARGS=""

test -f $RADIUSD || exit 0
test -f $RADDBDIR/radiusd.conf || exit 0

if [ ! -d $rundir ] ; then
    mkdir $rundir
    chown ${RADIUS_USER}:${RADIUS_USER} $rundir
    chmod 775 $rundir
fi

if [ ! -d $logdir ] ; then
    mkdir $logdir
    chown ${RADIUS_USER}:${RADIUS_USER} $logdir
    chmod 770 $logdir
    chmod g+s $logdir
fi

if [ ! -f $logdir/radius.log ]; then
    touch $logdir/radius.log
fi

chown ${RADIUS_USER}:${RADIUS_USER} $logdir/radius.log

```

```

chown -R ${RADIUS_USER}:${RADIUS_USER} /usr/local/freeradius/etc/raddb
chown -R ${RADIUS_USER}:${RADIUS_USER} ${rundir}/..
chmod 660 $logdir/radius.log

case "$1" in
  start)
    echo -n "Starting $DESC:"
    $RADIUSD $ARGS
    echo "radiusd"
    ;;
  stop)
    [ -z "$2" ] && echo -n "Stopping $DESC: "
    [ -f $rundir/radiusd.pid ] && kill -TERM `cat $rundir/radiusd.pid`
    [ -z "$2" ] && echo "radiusd."
    ;;
  reload|force-reload)
    echo "Reloading $DESC configuration files."
    [ -f $rundir/radiusd.pid ] && kill -HUP `cat $rundir/radiusd.pid`
    ;;
  restart)
    sh $0 stop quiet
    sleep 3
    sh $0 start
    ;;
  check)
    $RADIUSD -CX $ARGS
    exit $?
    ;;
  *)
    echo "Usage: /etc/init.d/$RADIUS {start|stop|reload|restart|check}"
    exit 1
    stop
    ;;
  status)
    status `cat $prog`
    ;;
  restart|force-reload)
    stop
    start
    ;;
  try-restart|condrestart)
    if status `cat $prog` > /dev/null; then
      stop
      start
    fi
    ;;
  reload)
    exit 3
    ;;
  *)
    echo `cat $prog`"Usage: `cat $prog` {start|stop|status|restart|try-restart|force-reload}"
    exit 2
esac

```

Включаем в автозагрузку

```

chmod +x /etc/init.d/radiusd
update-rc.d radiusd defaults
update-rc.d radiusd enable

```

Устанавливаем конфигурационные файлы ABILLS

```
rm /usr/local/freeradius/etc/raddb/sites-enabled/*
cp /usr/abills/misc/freeradius/v3/mods-enabled/perl /usr/local/freeradius/etc/raddb/mods-enabled/perl
cp /usr/abills/misc/freeradius/v3/mods-enabled/sql /usr/local/freeradius/etc/raddb/mods-enabled/sql
cp /usr/abills/misc/freeradius/v3/sites-enabled/abills_default /usr/local/freeradius/etc/raddb/sites-enabled/abills_default
cp /usr/abills/misc/freeradius/v3/users /usr/local/freeradius/etc/raddb/users
```

Меняем в конфиге радиуса

```
nano /usr/local/freeradius/etc/raddb/radiusd.conf

prefix = /usr/local/freeradius

user = freerad
group = freerad
```

Благодаря модулю **sql**, можно хранить сервера доступа в базе данных (всё равно необходим перезапуск радиуса после добавления/изменения серверов, но не надо править **clients.conf**).

Очищаем список серверов доступа из **/usr/local/freeradius/etc/raddb/clients.conf** (чтоб не было дубликатов из-за записей в базе)

```
echo '' > /usr/local/freeradius/etc/raddb/clients.conf
cp /usr/abills/misc/freeradius/v3/mods-enabled/sql /usr/local/freeradius/etc/raddb/mods-enabled/
```

Заполняем нужные нам параметры соединения с БД

```
nano /usr/local/freeradius/etc/raddb/mods-enabled/sql
```

```
sql {
    database = "mysql"
    driver = "rlm_sql_${database}"
    server = "localhost"
    port = 3306
    login = "abills"
    password = "sqlpassword"
    radius_db = "abills"

    '%secretkey%' 'test12345678901234567890'
```

Создаём user и group (если freeradius ранее не был установлен, то без этого не стартует)

```
groupadd freerad
useradd -g freerad -s /bash/bash freerad
```

```
chown -R freerad:freerad /usr/local/freeradius/etc/raddb
mkdir /var/run/radiusd/
chown -R freerad:freerad /var/run/radiusd/
```

стартуем в режиме отладки

```
radiusd -X
```

Если запустилось без ошибок, запускаем

```
service radiusd start
```

Постнастройка Abills

Вносим периодические процессы

/etc/crontab

```
* /5 * * * * root /usr/abills/libexec/billd -all
1 0 * * * root /usr/abills/libexec/periodic daily
1 1 * * * root /usr/abills/libexec/periodic monthly
```

и даем доступ на ведение лога SQL ошибок

```
touch /usr/abills/var/log/sql_errors && chmod 666 /usr/abills/var/log/sql_errors
```

Добавляем линки на **gzip** и **mysqldump** для создания бэкапов базы:

```
ln -s /bin/gzip /usr/bin/gzip
ln -s /usr/bin/mysqldump /usr/local/bin/mysqldump
```

Веб интерфейс администратора: (IP адрес можно узнать с помощью `ip a`)

<https://your.host:9443/admin/>

Логин администратора по умолчанию **abills** пароль **abills**

Веб интерфейс для пользователей:

<https://your.host:9443/>

Первые шаги

В интерфейсе администратора прежде всего надо сконфигурировать сервера доступа NAS (Network Access Server).

Переходим в меню

Настройка > Сервера доступа

Параметры

IP	IP адрес NAS сервера
Название	Название
Radius NAS-Identifier	Идентификатор сервера (можно не вписывать)
Описание	Описание сервера
Тип	Тип сервера. В зависимости от типа по разному обрабатываются запросы на авторизацию
Alive	Период отправки Alive пакетов
Disable	Отключить
:Manage:	Секция менеджмента NAS сервера
IP:PORT	IP адрес и порт для контроля соединения. Например, для отключения пользователя из веб-интерфейса

User	Пользователь для контроля
Password	Пароль
RADIUS Parameters	Дополнительные параметры которые передаются NAS серверу после успешной авторизации

После заведения сервера доступа добавьте ему пул адресов IP POOLS.

Первый IP	Первый адрес в пуле
Количество	Количество адресов

Одному серверу доступа может принадлежать несколько пулов адресов.

Создание тарифного плана

Меню

Настройка→Интернет→Тарифные планы

Регистрация пользователя

Клиенты→Логины→Добавить пользователя

Заведение сервиса Internet на пользователя.

Клиенты→Логины→Сервисы→Интернет

Проверка

Для проверки правильно ли настроен сервис нужно запустить утилиту radtest указав логин и пароль существующего пользователя.

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

Логин: test Пароль: 123456

```
radtest test 123456 127.0.0.1:1812 0 secretpass 0 127.0.0.1
```

Если всё правильно настроено, в журнале ошибок **Отчёт>Интернет+>Последнее подключение** должна появиться строка

```
2014-12-23 12:55:55 LOG_INFO: AUTH [test] NAS: 1 (xxx.xxx.xxx.xxx) GT: 0.03799
```

Если Вы увидите другие ошибки смотрите в [список ошибок](#). Если журнал ошибок пуст, значит неправильно настроено взаимодействие с RADIUS сервером (еще раз пройдите по секции FreeRadius).

Дополнительно

- [Настройка NAS accel-ppp](#)
- [Настройка NAS Mikrotik](#)
- [Настройка PPPoE Ubuntu](#)