

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П. КОРОЛЕВА»
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Сервера интернет приложений в Linux

*Утверждено Редакционно-издательским советом университета
в качестве методических указаний к лабораторной работе*

САМАРА
Издательство СГАУ
2010

УДК 004.451

Составитель А.М. С у х о в

Рецензент: к.т. н., доц. Попов С.Б.

Сервера интернет приложений в Linux: Методические указания к лабораторной работе/ Сост. А.М. Сухов. - Самара: Изд-во Самарского государственного аэрокосмического университета, 2010. 28 с.

В настоящих методических указаниях приведен материал, необходимый для выполнения лабораторных работ по дисциплинам «Операционная система Linux на высокопроизводительных кластерах», «Перспективные информационные технологии»

Целью лабораторных работ является изучение основных возможностей по установке серверов наиболее важных интернет приложений apache (веб сервер), PHP, MySQL, FTP, CMS Joomla и приобретение навыков работы в операционной системе Linux.

Предназначено для слушателей ФПК СГАУ и студентов специальностей 010500, 010501

**© Самарский государственный
аэрокосмический университет, 2010**

1. Цель лабораторной работы

- изучение основных возможностей по установке серверов наиболее важных интернет приложений apache (веб сервер), PHP, MySQL, FTP, CMS Joomla
- ознакомление с альтернативными путями инсталляции посредством командной строки и из графического интерфейса

2. Основные сведения

В мае 2010 года семейство операционных систем на базе ядра Linux — третье по популярности после Windows и Apple (1,13%) в мире на рынке настольных компьютеров. На рынке веб-серверов доля Linux порядка 65%. По данным TOP500, Linux используется на 91% самых мощных суперкомпьютеров планеты.

Можно выделить несколько основных областей, где нередко можно встретить Linux:

- Серверы, требующие высокого аптайма (время непрерывной работы).
- Компьютеры нестандартной архитектуры (например, суперкомпьютеры) — из-за возможности быстрой адаптации ядра операционной системы и большого количества ПО под нестандартную архитектуру.
- Системы военного назначения (например, МСВС РФ) — по соображениям безопасности.
- Компьютеры, встроенные в различные устройства (банкоматы, терминалы оплаты, мобильные телефоны, маршрутизаторы, стиральные машины и даже беспилотные военные аппараты) — из-за широких возможностей по конфигурированию Linux под задачу, выполняемую устройством, а также отсутствия платы за каждое устройство.
- Массовые специализированные рабочие места (например, тонкие клиенты, нетбуки) — также из-за отсутствия платы за каждое рабочее место и по причине их ограниченной вычислительной мощности, которой может не хватать для проприетарных ОС.
- Старые компьютеры с ограниченными ресурсами быстродействия и оперативной памяти, для них используются быстрые рабочие окружения или оконные менеджеры, не требовательные к ресурсам (например, LXDE, Openbox, Xfce, Fluxbox).

ПО для серверов Internet/Intranet

Стандартно в поставку Linux входят: *Apache* - самый популярный в Internet http-сервер; *Sendmail* - программа передачи электронной почты (Mail Transfer Agent), *ftp*, *pop3/imap*, *news* серверы, сервер доменных имен, ширс over tcp/ip, *squid* (кеширующий http/ftp прокси), средства динамической маршрутизации и др.

Файл-сервер

Linux может служить файл-сервером по протоколам NFS (как правило используем только на Unix машинах), SMB (Netbios over TCP/IP, используемый на различных

Windows платформах), FTP (File Transfer Protocol, один из наиболее распространённых протоколов передачи файлов по сети), AppleShare и IPX (Novell).

Middleware

Существует несколько свободных реализаций архитектуры OMG CORBA.

СУБД

Основные производители систем управления реляционными базами данных уже перенесли свои продукты на Linux: существуют Linux версии MySQL, PostgreSQL, IBM DB2, Informix, Oracle, Sybase, Firebird, Progress и др. Также под управлением Linux сегодня работает большинство постреляционных многомерных и объектных СУБД: jBase, IBM U2, TigerLogic D3, Intersystems Cache и др. Linux может использоваться в качестве платформы для запуска системы управления предприятием SAP R/3.

Офисные пакеты

Наиболее популярным является кроссплатформенный офисный пакет OpenOffice.org (ООо). Для оконных сред KDE и GNOME разрабатываются свои версии офисных пакетов: KOffice и GnomeOffice. Кроме них доступны коммерческие офисные пакеты от многих фирм.

Установка Apache:

Как правило, при установке операционной системы можно выбрать опцию и установить большинство серверов по умолчанию, если же этого не было сделано, то следуем установить соответствующее ПО самостоятельно. Разархивируем архив:

```
#tar -xjvf httpd-2.0.58.tar.bz2
```

Далее заходим в директорию, которую мы только что распаковали:

```
#cd httpd-2.0.58/
```

Начинаем установку:

```
./configure --prefix=/usr/local/httpd (--prefix=/user/local/httpd
```

- описывает путь установки)

```
#make
```

```
#make install
```

Далее мы должны перейти в директорию `/usr/local/httpd/conf` и отредактировать файл `httpd.conf`:

```
#cd /usr/local/httpd/conf/
```

```
#nano httpd.conf
```

Находим и изменяем строки в конфигурационном файле:

ServerAdmin eliziyim@example.com

(вместо eliziyim@example.com должен быть ваш почтовый ящик)

ServerName www.example.com:80

(www.example.com:80 - здесь вы должны указать адрес сервера и через двоеточие порт)

Находим строки:

```
<Directory />  
    Options FollowSymLinks  
    AllowOverride None  
</Directory>
```

И меняем их на:

```
<Directory />  
    Options None  
    AllowOverride None  
</Directory>
```

Находим:

```
<Directory "/usr/local/httpd/htdocs">  
    Options Indexes FollowSymLinks  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

Меняем на:

```
<Directory "/usr/local/httpd/htdocs">  
    Options None  
    AllowOverride None  
    Order allow,deny  
    Allow from all  
</Directory>
```

Далее скрываем информацию о версии Apache и немного защищаем его. Для этого редактируем в конфигурационном файле следующие записи:

```
ServerTokens Prod
ServerSignature Off
User nobody
Group nobody
Timeout 45
UserDir disable
```

Выставляем нужные права доступа:

```
#cd /usr/local/httpd/
#chown root . bin conf logs
#chgrp root . bin conf logs
#chmod 755 . bin conf logs
#chown root /usr/local/httpd/bin/httpd
#chgrp root /usr/local/httpd/bin/httpd
#chmod 511 /usr/local/httpd/bin/httpd
```

Установка MySQL:

Нужно добавить новую группу и пользователя в систему прежде чем начинать установку:

```
#groupadd mysql
#useradd -g mysql mysql
```

Разархивируем архив и переходим в полученную директорию:

```
#tar -zxvf mysql-5.0.22.tar.gz
#cd mysql-5.0.22/
```

Далее приступаем к конфигурированию и установке:

```
#!/configure --prefix=/usr/local/mysql
```

(--prefix=/usr/local/mysql - описывает путь установки)

```
#make
#make install
```

Копируем конфигурационный файл:

```
#cp support-files/my-medium.cnf /etc/my.cnf
```

Переходим в директорию /usr/local/mysql и выполняем там установку административных таблиц:

```
#cd /usr/local/mysql  
#bin/mysql_install_db -user=mysql
```

Выставляем нужные права доступа:

```
#chown -R root .  
#chown -R mysql var  
#chgrp -R mysql .
```

Затем нам нужно будет запустить сервер, подключиться к нему и изменить пароль:

```
#!/usr/local/mysql/bin/mysqld_safe &  
#!/usr/local/mysql/bin/mysql -u root
```

```
mysql> use mysql;  
mysql> set password for 'root'@'localhost' = password('passwd');  
mysql> flush privileges;  
mysql> \q
```

Вместо слова passwd напишите свой пароль.

На этом установка MySQL сервера окончена.

Установка PHP:

Разархивируем архив:

```
#tar -xjvf php-5.1.4.tar.bz2
```

Переходим в полученную директорию:

```
#cd php-5.1.4/
```

Приступаем к конфигурированию и установке:

```
#!/configure -with-apxs2=/usr/local/httpd/bin/apxs -with-  
mysql=/usr/local/mysql  
#make  
#make install
```

Скопируем конфигурационный файл:

```
#cp php.ini-recommended /usr/local/lib/php.ini
```

Теперь откроем отредактируем немного конфигурационный файл Apache:

```
#nano /usr/local/httpd/conf/httpd.conf
```

Добавляем в начало файла строку:

```
AddType application/x-httpd-php .php
```

```
AddType application/x-httpd-php .html
```

```
AddType application/x-httpd-php-source .phps
```

Ищем строку "DirectoryIndex", в её конец добавлем значение index.php.

Строка DirectoryIndex должна теперь выглядеть так:

```
DirectoryIndex index.html index.html.var index.php
```

На этом установка всего программного обеспечения завершена. Осталось запустить, если у вас не запущен Apache или перезагрузить его, для вступления в силу новых параметров которые мы внесли в конфигурационный файл.

```
#!/usr/local/httpd/bin/apachectl start - для запуска сервера
```

```
#!/usr/local/httpd/bin/apachectl restart - для перезагрузки сервера
```

На этом установка из командной строки закончена. Нужно проверить работоспособность всего установленного. Для этого мы напишем скрипт на PHP:

```
#cd /usr/local/httpd/htdocs/
```

```
#nano mysql.php
```

```
<?
```

```
$dblocation="127.0.0.1";
```

```
$dbuser="root";
```

```
$dbpasswd="passwd";
```

```
$dbcnx=mysql_connect($dblocation, $dbuser, $dbpasswd);
```

```
if(!$dbcnx)
```

```
{
```

```
    echo("Не удалось подключиться к базе данных");
```

```
    exit();
```

```
}
```



```
$dbq=mysql_query("select version();");
echo(mysql_result($dbq,0));

?>
```

В переменной "\$dbpasswd" замените параметр "passwd" на пароль от вашей базы данных(MySQL).

Теперь открываем любой браузер и переходим по ссылке <http://127.0.0.1/mysql.php> :

```
#lynx http://127.0.0.1/mysql.php
```

На этом установка, настройка и тестирование веб-сервера окончена.

Virtual Host (виртуальный хостинг)

Если вы хотите настроить виртуальные хосты, то вам нужно будет использовать директивы <NameVirtualHost> и <VirtualHost>.

Предположим, что в строке ServerName конфигурационного файла Apache мы указали www.example.com:80. Нам нужно настроить виртуальный хост forum.example.com и user.example.com. Для этого открываем конфигурационный файл и добавляем в его конец строки:

```
<NameVirtualHost 192.168.0.66>
<VirtualHost 192.168.0.66>
    ServerName forum.example.com
    ServerAdmin eliziyim@example.com
    DocumentRoot /usr/local/httpd/htdocs/forum
</VirtualHost>

<VirtualHost 192.168.0.66>
    ServerName user.example.com
    ServerAdmin eliziyim@example.com
    DocumentRoot /usr/local/httpd/htdocs/user
</VirtualHost>
```

Здесь вместо ip адреса 192.168.0.66 - вы должны указать свой ip адрес, если у вас их несколько, то просто поставьте вместо ip - *

В строке DocumentRoot вы должны указать директорию, в которой будут храниться файлы принадлежащие этому виртуальному хосту. Предположим, если у вас форум, то файлы форума вы должны поместить в директорию */usr/local/httpd/htdocs/forum*

После всех манипуляций мы должны перезагрузить сервер. Не забудьте добавить нужные для DNS сервера записи.

Настройка CMS Joomla

В настоящем разделе будет описана установка CMS Joomla в графическом режиме, ряд действий будут повторять описанную выше установку из командной строки.

Итак, для начала нам нужно будет скачать сам дистрибутив Joomla. Делать это придётся самим, поскольку в официальных репозиториях Ubuntu соответствующего пакета к сожалению нету. Joomla распространяется бесплатно, поэтому найти и скачать её довольно просто. Первая ссылка в google приводит нас на сайт русскоязычного сообщества: joomlaportal.ru. Для удобства я скачал русскоязычную версию, но если вы захотите использовать самую последнюю версию данной среды, скорее всего, придётся скачивать английский вариант.

На этом же сайте в разделе «Системные требования» можно найти список программ, которые нам понадобятся. Для полноценной работы с этой CMS (английское сокращение от «система управления контентом») нам нужны apache (веб-сервер), php (язык программирования, на котором написана Joomla) и mysql (сервер баз данных). Все эти программы легко устанавливаются из репозитория или из режима командной строки, как было описано выше.

Сочетание программ Apache-MySQL-PHP является довольно распространённым, поэтому для более быстрой и лёгкой его установки в Ubuntu предусмотрен так называемый «набор» (taskset), и все три программы устанавливаются одной командой. Запустите, пожалуйста, терминал (меню Приложения → Стандартные → Терминал) и выполните следующую команду (для её выполнения скорее всего потребуется ввести ваш пароль):

```
$sudo taskset install lamp-server
```

В процессе выполнения этой команды вас также попросят ввести пароль для пользователя root сервера баз данных. Вводить его не обязательно, но очень желательно.

Кстати, LAMP расшифровывается как Linux+Apache+MySQL+Php.

После того, как установка завершена, мы закрываем менеджер пакетов и плавно переходим к настройке всей получившейся системы.

Настройка веб-сервера Apache

У Apache есть множество настроек и модулей. Давайте немного разберёмся, как это всё работает.

Конфигурация веб-сервера хранится в каталоге `/etc/apache2`. Перейдя туда, вы увидите несколько файлов и каталогов:

`apache2.conf` — это главный конфигурационный файл

ports.conf — здесь хранятся настройки портов, которые будет слушать сервер. Если вы не знаете, что такое порт, ничего страшного, — сейчас важно понять, что отредактировав этот файл можно ограничить доступ к нашему веб-серверу. Например, если заменить строчку «Listen 80» на «Listen 127.0.0.1:80» и перезапустить веб-сервер, то открывать страницы, которые находятся на вашем компьютере, сможете только вы.

В каталогах mods-available и sites-available находятся так называемые подключаемые модули и список доступных сайтов. Модули предназначены для того, чтобы можно было легко изменить конфигурацию сервера, не рискуя при этом запутаться в его настройках. Ниже я продемонстрирую, как это можно сделать; mods-enabled и sites-enabled — эти папки хранят списки подключенных модулей и сайтов.

Для того, чтобы убедиться, что ваш веб-сервер работает, откройте страницу <http://localhost> в браузере (адрес localhost указывает на ваш компьютер; вместо него можно с таким же успехом использовать набор цифр «127.0.0.1»). Если вы увидите страницу с надписью «It works!», значит всё идёт прекрасно и сервер работает.

Давайте попробуем немного изменить конфигурацию веб-сервера. Узнать, что делает тот или иной модуль можно по адресу <http://www.oglib.ru/apman/mod>. К сожалению, перевод машинный, так что не всегда будет понятно, что имел ввиду автор. Ну а если вы кое-как владеете английским языком, тогда вы сможете найти нужную информацию здесь: <http://httpd.apache.org/docs/2.0/mod>

Для начала нам нужно включить модуль *mod_userdir*, который позволяет создавать сайты прямо в домашнем каталоге пользователя без лишних усилий. Для этого мы открываем терминал и выполняем команды:

```
$sudo a2enmod userdir
```

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

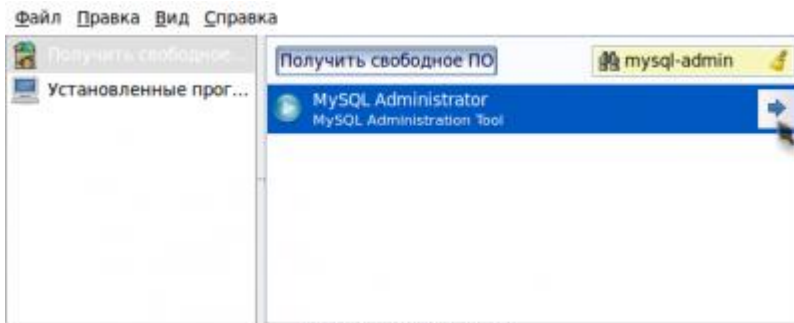
Первая команда подключает нужный нам модуль, а вторая перезапускает веб-сервер для того, чтобы изменения вступили в силу. Вот и всё, модуль подключен.

Теперь для того, чтобы сделать сайт, нужно в нашем домашнем каталоге создать папку с названием *public_html* и заполнить её содержимым. Чтобы создать папку, идём в меню «Переход → Домашняя папка», в открывшемся окне кликаем правой кнопкой мыши и выбираем соответствующий пункт меню. Распакуйте в эту папку скачанный архив Joomla. Надеюсь это не вызовет у вас больших затруднений.

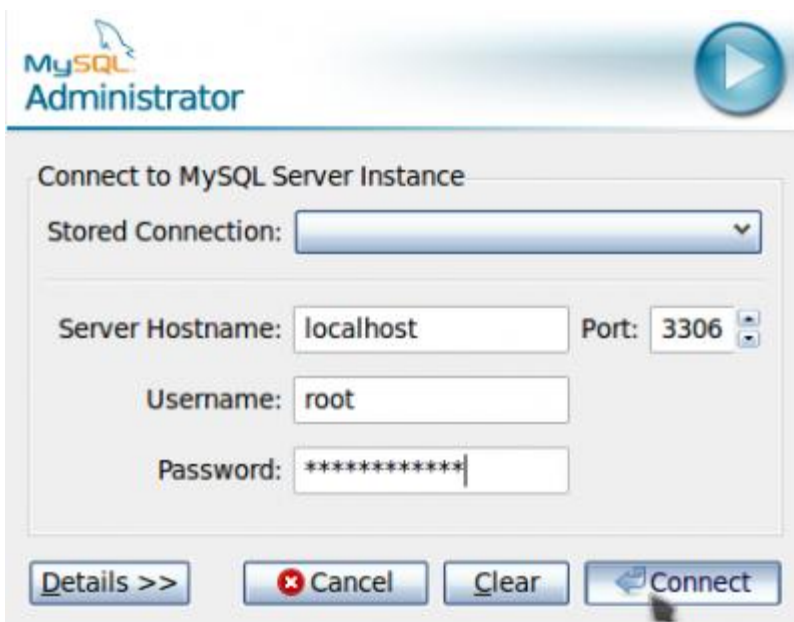
Если вы всё сделали правильно, то открыв страницу <http://localhost/~username> (замените username на имя вашего домашнего каталога; обратите внимание на символ ~), вы получите приглашение к началу установки Joomla. Но делать это еще рано, потому что у нас пока не настроена база данных.

Настройка базы данных

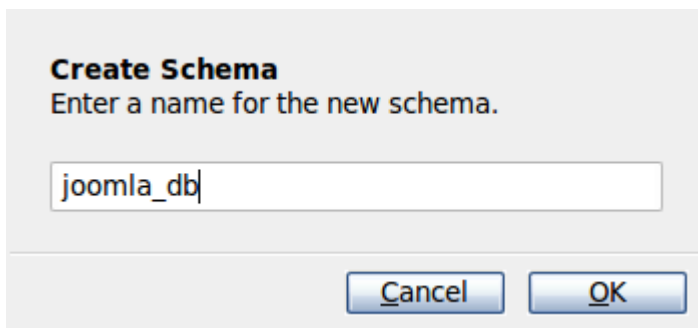
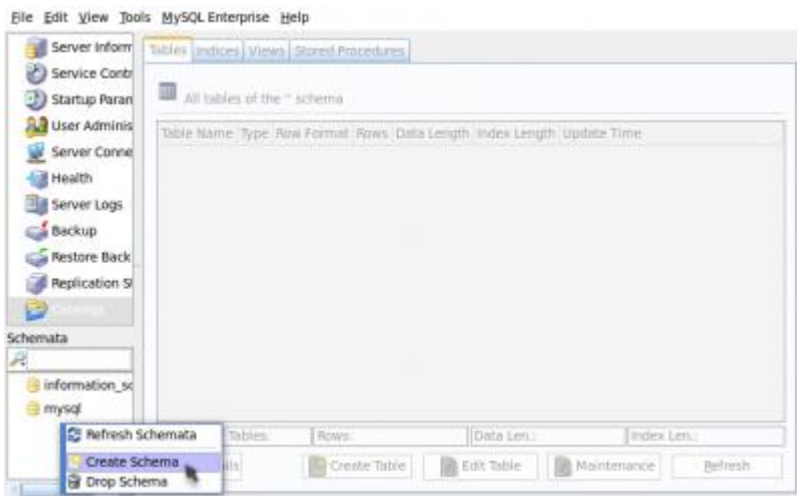
Для того, чтобы было удобнее работать с сервером баз данных MySQL, мы установим небольшую программку под названием mysql-admin. Зайдите, пожалуйста, в «Центр приложений Ubuntu», введите в строке поиска «mysql-admin» (без кавычек) и установите программу:



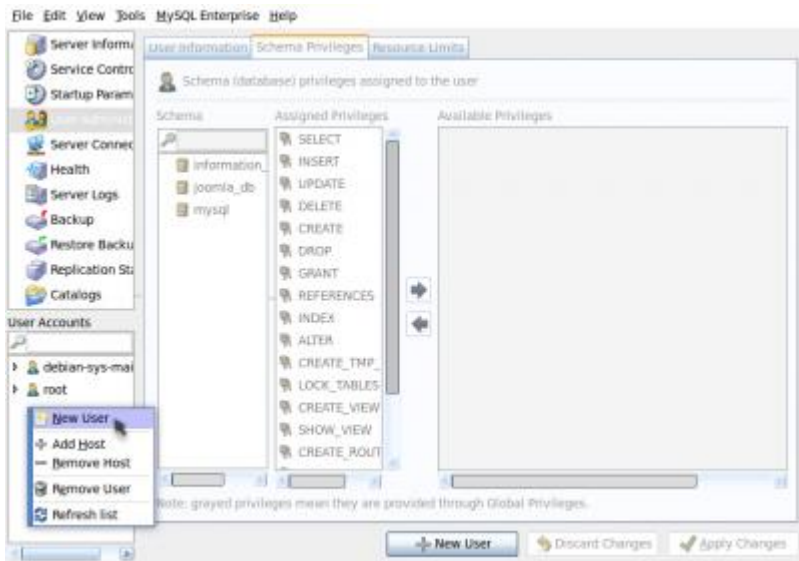
Для того, чтобы запустить установленную только что программу, перейдем в меню Приложения → Программирование → MySQL Administrator. В появившемся окне вводим свои данные:



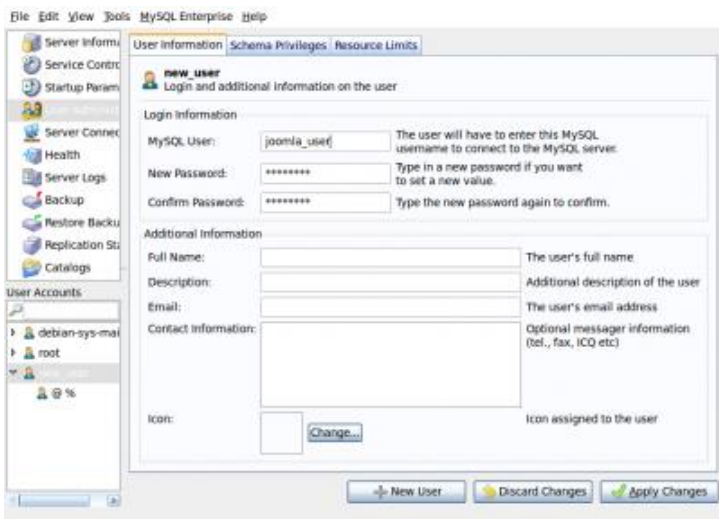
Для наших целей мы создадим базу данных под названием joomla_bd:



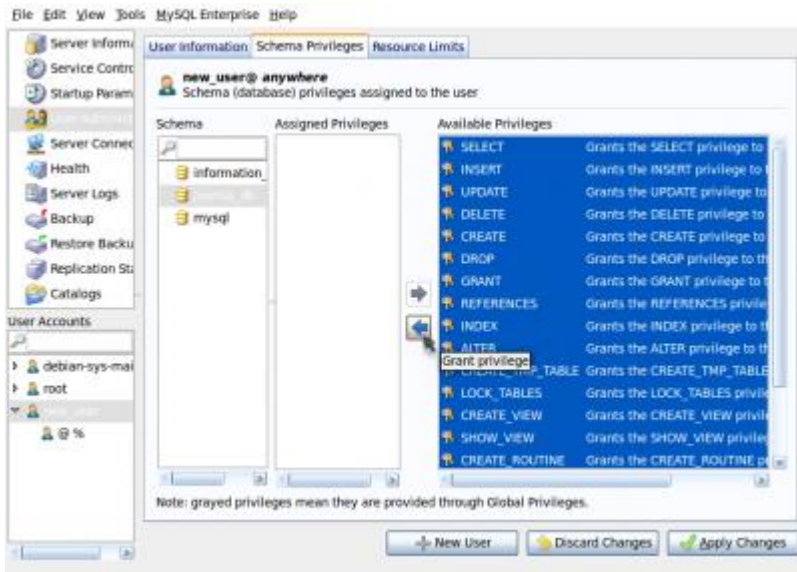
Теперь давайте добавим специального пользователя для работы с нашей новой базой данных. Можно конечно этого не делать и при установке Joomla указать пользователя root, но это будет очень не правильно.



Укажем в поле имени пользователя `joomla_user`. Также нужно будет задать пароль для этого пользователя, который нам понадобится во время установки Joomla. Кстати, хранить этот пароль после того, как вы его ввели в Joomla, не обязательно, так как он вам больше скорее всего не понадобится.



Теперь зададим права на доступ к базе данных. Для этого перейдём на вкладку «Schema Privileges», выберем нашу базу данных, кликнем левой кнопкой мыши на какое-нибудь поле в списке «Available Privileges», затем нажмём на клавиатуре Ctrl-A (для этого сначала нужно нажать клавишу Ctrl, а потом не отпуская её латинскую «A»), этим мы добъёмся того, что не нужно будет выделять каждую привилегию отдельно. Теперь осталось переместить все привилегии в столбик «Assigned Privileges», для этого нужно кликнуть левой кнопкой мыши по стрелочке.



Подтвердим изменения нажав «Apply Changes» (справа снизу). Всё, дело сделано, теперь можно закрыть программу MySQL Administrator.

Установка Joomla в Ubuntu Linux

Возвращаемся в браузер и снова открываем страницу <http://localhost/~username> (не забывайте, что username нужно заменять). Теперь всё готово для нашей установки.


http://localhost/~spinal/installation/index.php

Joomla! Web Installer

Joomla! Установка 1.5.14

Шаги

- 1: Выбор языка
- 2: Начальная проверка
- 3: Лицензия
- 4: База данных
- 5: Конфигурация FTP
- 6: Конфигурация
- 7: Завершение



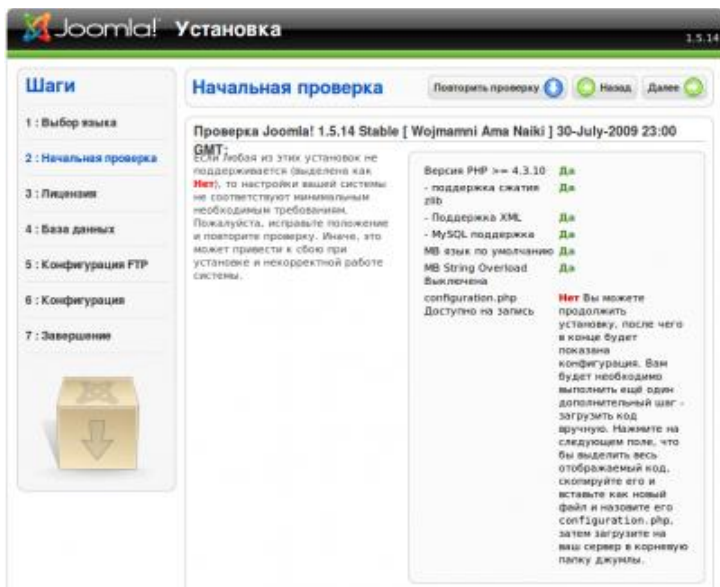
Выберите язык

Выберите язык установки

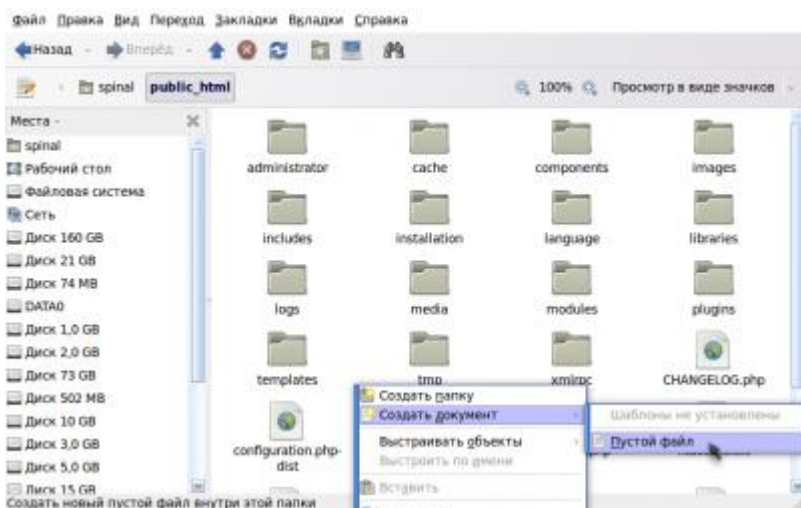
10-LA - Lao (ລາວ)
lt-LT - Lithuanian
lv-LV - Latvian
mk-MK - Македонски (Macedonia)
mn-MN - Mongolian
nb-NO - Norsk bokmål (Norway)
nl-NL - Nederlands (nl-NL)
pl-PL - Polish (Poland)
pt-BR - Português (Brasil)
pt-PT - Português (pt-PT)
ro-RO - Română (Romania)
ru-RU - Русский (RU)
sd-PK - سنڌي (Sindh)
si-LK - Sinhala (Sri Lanka)
sk-SK - Slovenčina (Slovenská Republika)
sq-AL - Albanian-AL
sr-ME - Montenegrin
sr-RS - Serbian (RS)
sv-SE - Svenska (Sverige)
sy-NO - Syriaci (Iraq)
ta-LK - Tamil (Sri Lanka)
th-TH - Thai (ประเทศไทย)
tr-TR - Türkçe (Türkiye)
uk-UA - Ukrainian (Ukraine)
ur-PK - Urdu Pakistan (الہند)
zh-CN - Simplified Chinese
zh-TW - Traditional Chinese (Taiwan)

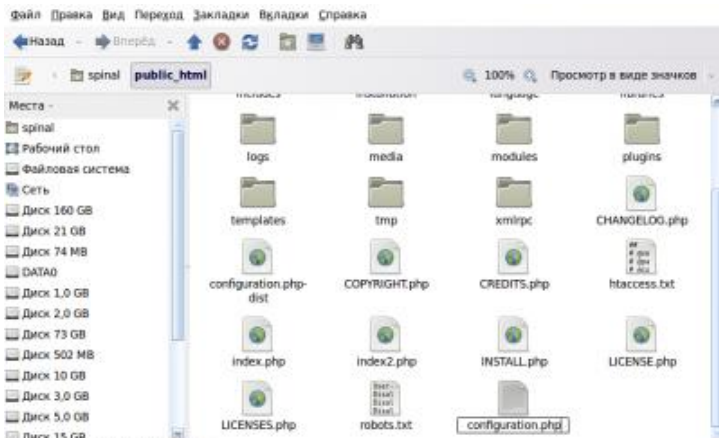
Далее

Joomla! Распространяется по лицензии GNU/GPL

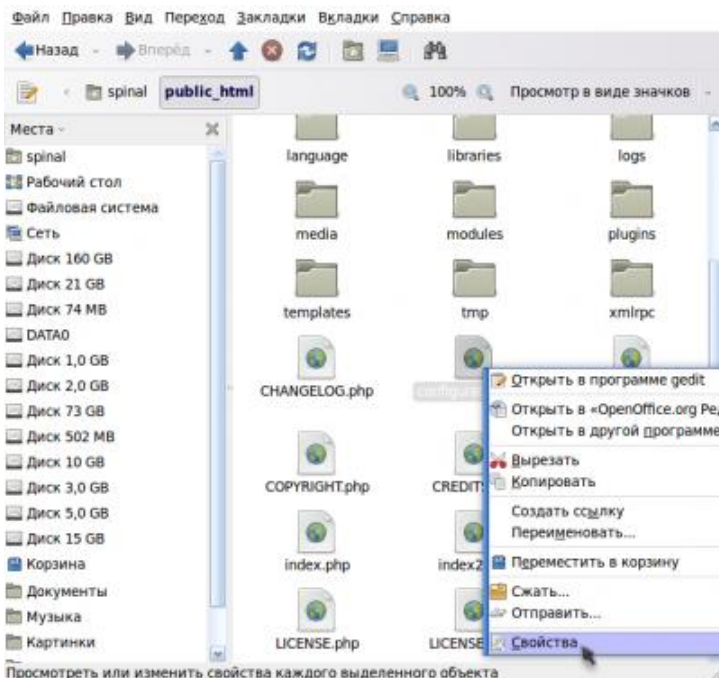


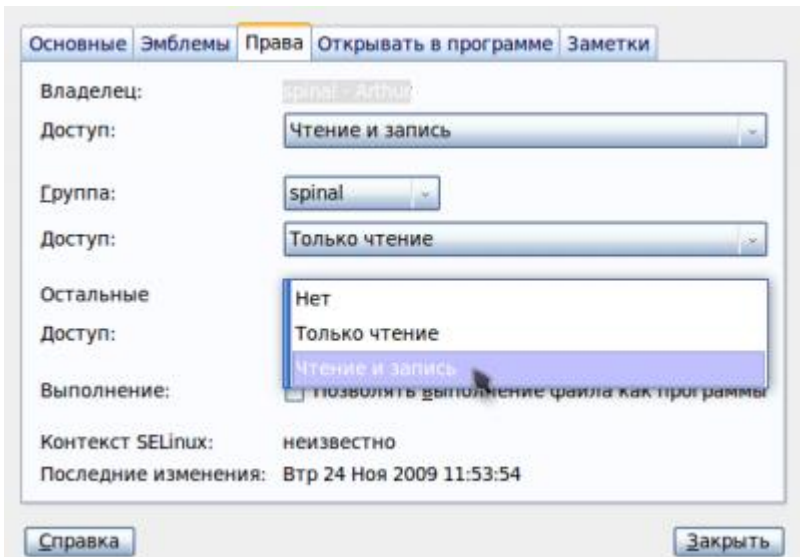
И здесь мы видим, что одно из условий (configuration.php доступно на запись) не удовлетворено. Давайте это исправим. Перейдём в папку public_html нашего домашнего каталога и создадим там файл configuration.php:





Заходим в свойства созданного файла и предоставляем нашему веб-серверу права на запись в него (это нужно для того, чтобы Joomla! смогла сохранить свою конфигурацию)

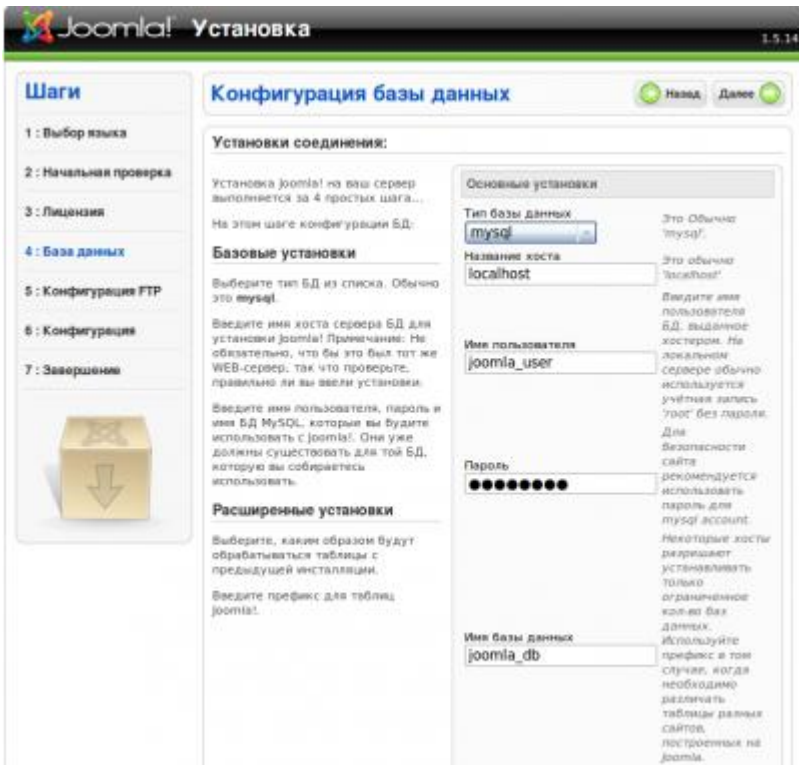




Закроем и проверим (нужно нажать «Повторить проверку»):



Идём дальше и плавно переходим на страницу конфигурации базы данных:



Если вы нигде ничего не напутали, то следующая страница должна предложить нам ввести установки ftp сервера.

Настройка FTP сервера

Устанавливаем ftp сервер vsftp. Сделать это можно введя в терминале:

```
$sudo aptitude install vsftpd
```

Вы наверняка уже заметили, что в Ubuntu есть несколько способов установить ту или иную программу. Теперь немного изменим конфигурацию нашего ftp сервера, чтобы он сработался с Joomla!

```
$sudo gedit /etc/vsftpd.conf
```

1) анонимный доступ к ftp нам не понадобится, поэтому строчку «anonymous_enable=YES» меняем на «anonymous_enable=NO»

2) чтобы предоставить доступ Joomla, раскомментируем следующие 3 строчки (это значит, что нужно удалить значок # вначале):

```
#local_enable=YES
#write_enable=YES
#local_umask=022
```

3)в конце файла можно добавить запись «listen_address=127.0.0.1» – для того, чтобы нас не пытались взломать

Перезапускаем ftp сервер командой:

```
$sudo /etc/init.d/vsftpd restart
```

Добавим отдельного пользователя joomla_user для доступа по ftp:

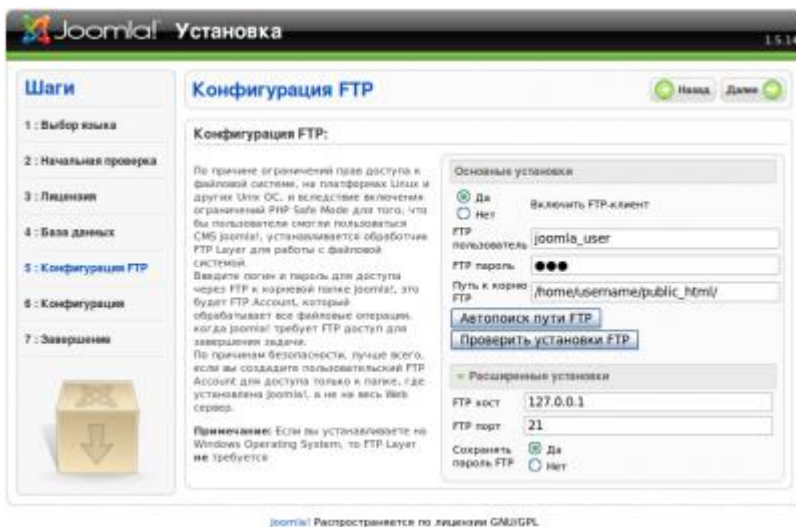
```
$sudo useradd -d $HOME/public_html -ou $(id -u) -g $(id -g) joomla_user
```

И сделаем для него пароль (нужно ввести 2 раза):

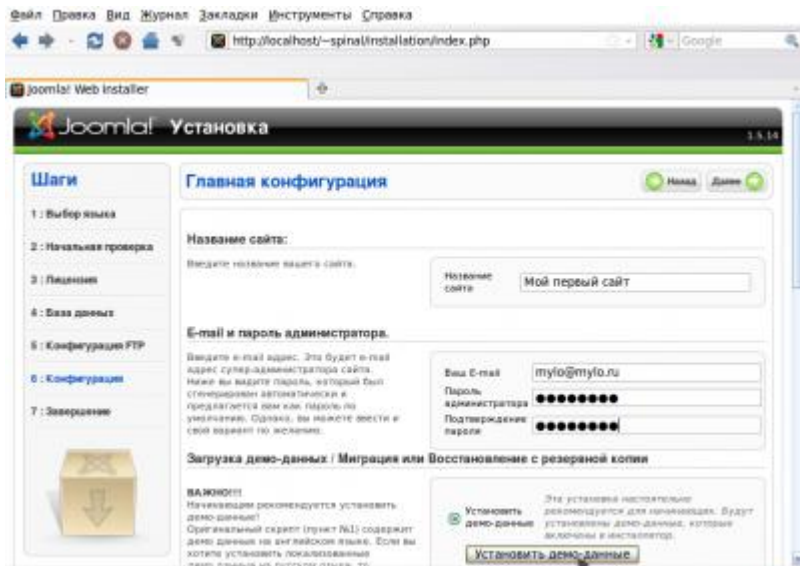
```
$sudo passwd joomla_user
```

Заканчиваем установку Joomla

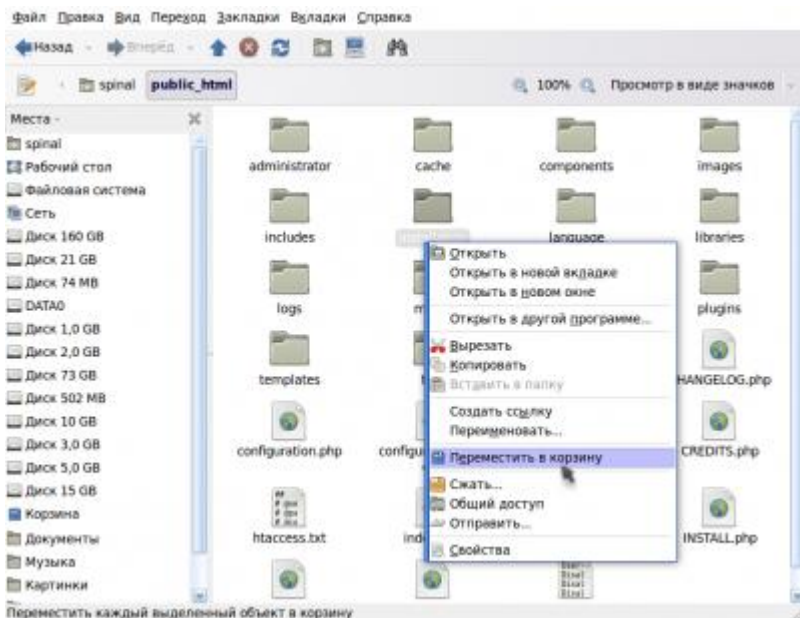
Мы остановились на том, что Joomla требовала у нас данные для доступа по ftp. Чтож, теперь мы можем ей на это ответить (не забудьте поменять username, или нажмите «Автопоиск пути FTP» для автоматического определения):

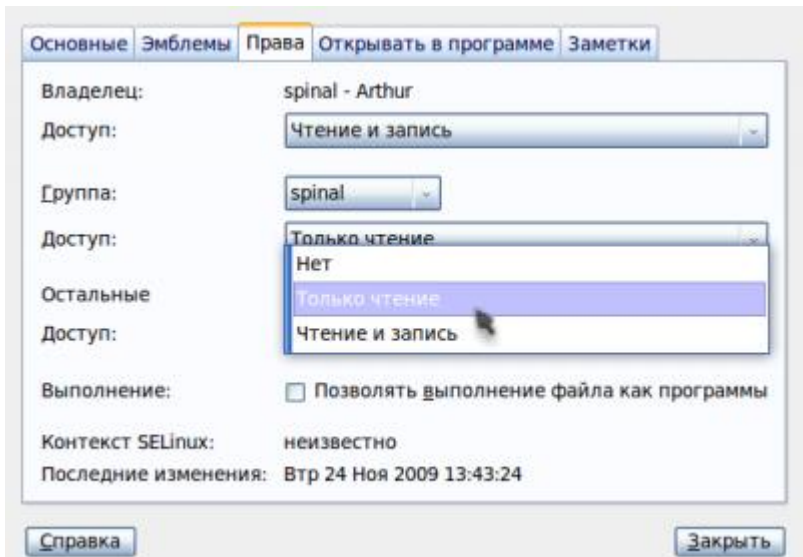


На следующей странице нужно ввести желаемую конфигурацию. Пароль администратора желательно запомнить, он нам ещё пригодится. Если вы такой же новичок в Joomla, как и я, тогда нажмите кнопку «Установить демо-данные», как того предлагает установка.



Теперь остался последний штрих: нужно удалить папку installation и снять с файла configuration.php права на запись для всех кого не попадая:





Теперь можно перейти к администрированию:

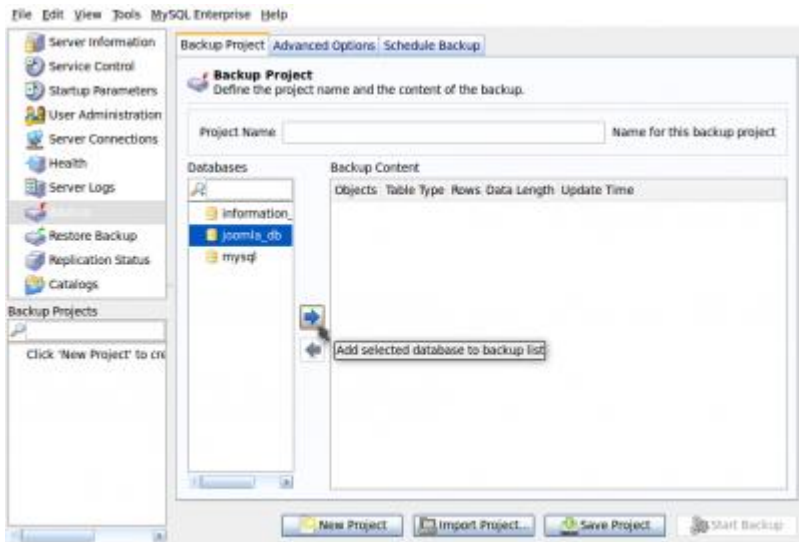
<http://localhost/~username/administrator/>

Взглянуть на то, что у нас получилось можно здесь:

<http://localhost/~username/>

Экспорт базы данных.

Для того, чтобы экспортировать базу данных, воспользуемся программой MySQL Administrator, которую мы недавно установили.



Кликаем по стрелочке, потом нажимаем кнопку «Start Backup», выбираем имя файла, куда сохранять базу данных и... всё!

Список контрольных вопросов

1. В каких сегментах информационных технологий ОС Linux занимает лидирующее положение
2. Перечислите названия основных программных продуктов, реализующих интернет приложения под Linux
3. Что такое виртуальный хостинг и как его настроить
4. Перечислите файлы конфигурации apache, MySQL, PHP
5. Какие компоненты необходимо установить для полноценной работы CMS Joomla?
6. В чем заключается настройка базы данных

Задания по лабораторной работе

1. Установить apache в консольном режиме, отредактировать домашнюю страничку и посмотреть файлы конфигурации
2. Установить в консольном режиме базу данных и PHP
3. Проверить работоспособность всех компонентов
4. Установить Joomla в графическом режиме
5. Проверить работоспособность системы, установив шаблон сайта.

Список литературы

1. *Уэли. М. и-др.*, Руководство по установке и использованию системы **Linux**. М.: ИЛКиРЛ, 1999
2. Александр Боковой, Александр Колотов, Александр Прокудин, Алексей Новодворский, Алексей Смирнов, Анатолий Якушин, Антон Бояршинов, Антон Ионов, Вадим Виниченко, Виталий Липатов, Георгий Курячий, Даниил Смирнов, Дмитрий Аленичев, Дмитрий Левин, Илья Трунин, Кирилл Маслинский, Максим Отставнов, Мэтт Уэлш, Олег Власенко, Сергей Турчин, Станислав Иевлев, Юрий Коновалов и другие; ALT Linux снаружи. ALT Linux внутри, ISBN 5-9706-0029-6, Издатель: ДМК пресс, 2006 г. Москва
3. Марк Г. Собелл, Практическое руководство по Red Hat Linux: Fedora Core и Red Hat Enterprise Linux, 2-е издание (Practical Guide to Red Hat Linux: Fedora Core and Red Hat Enterprise Linux), 1072 стр., с ил.; ISBN 5-8459-0841-8, 0-13-147024-8; формат 70x100/16; твердый переплет DVD-ROM; 2005, 2 кв.; Вильямс.
4. Разработка приложений в среде Linux. Программирование для linux, 2-е издание, Майкл К. Джонсон, Эрик В. Троан
5. Руководство администратора Linux. Установка и настройка. 2-е издание, Эви Немет, Гарт Снайдер, Трент Хейн
6. Linux. Библия пользователя, Кристофер Негус
7. Linux для чайников, 6-е издание, Ди-Анн Лебланк
8. Разработка ядра Linux, 2-е издание, Роберт Лав
9. Библиотека Qt 4. Программирование прикладных приложений в среде Linux., Чеботарев Арсений Викторович
10. Red Hat Linux Fedora 4. Полное руководство, Пол Хадсон, Эндрю Хадсон, Билл Болл, Хойт Дафф
11. Искусство программирования для Unix, Эрик С. Реймонд
12. Linux для "чайников", 5-е издание, Ди-Анн Лебланк
13. Red Hat Linux. Секреты профессионала, Наба Баркакати
14. Использование Linux, Apache, MySQL и PHP для разработки Web-приложений, Джеймс Ли, Brent Уэр
15. Секреты хакеров. Безопасность сетей - готовые решения, 4-е издание, Стюарт Мак-Клар, Джоэл Скембрей, Джордж Курц
16. FreeBSD: полный справочник., Родерик Смит

17. Секреты хакеров. Безопасность Linux — готовые решения, 2-е издание, Брайан Хатч, Джеймс Ли, Джордж Курц
18. Red Hat Linux 8. Библия пользователя, Кристофер Негус
19. Серверы Linux. Самоучитель, Птицын Константин Александрович
20. Безопасность Linux, 2-е издание, Скотт Манн, Эллен Л. Митчелл, Митчелл Крелл
21. Сетевые средства Linux, Родерик Смит
22. Руководство администратора Linux, Эви Немет, Гарт Снайдер, Трент Хейн
23. Сети TCP/IP, том 3. Разработка приложений типа клиент/сервер для Linux/POSIX, Дуглас Камер, Дэвид Л. Стивенс
24. Секреты хакеров. Безопасность Linux — готовые решения, Брайан Хатч, Джеймс Ли, Джордж Курц
25. Программирование для Linux. Профессиональный подход, Марк Митчелл, Джеффри Оулдем, Алекс Самьюэл
26. Использование Linux, 6-е издание. Специальное издание, Дэвид Бендел, Роберт Нейпир
27. Создание сетевых приложений в среде Linux, Шон Уолтон
28. Освой самостоятельно Linux за 24 часа, 3-е издание,
29. Система электронной почты на основе Linux. Руководство администратора, Ричард Блам
30. Системное администрирование Linux, М. Карлинг, Стефан Деглер, Джеймс Деннис

Учебное издание

Сервера интернет приложений в Linux

Методические указания

Составитель: Сухов Андрей Михайлович

**Изд-во Самарского государственного
аэрокосмического университета.
443086 Самара, Московское шоссе, 34.**