

# Лабораторная работа №4 (дополнение)

## Настройка файлового и веб сервера.

**Цель работы:** Получить базовые знания управления процессом установки, программных пакетов.

### Ход работы:

В предыдущей работе Вы установили пакеты файлового сервера и настроили конфигурационные файлы. Не забудьте создать соответствующие сетевые ресурсы, описанные в конфигурационном файле (*/disks/public*). Для упрощенного создания используйте команду `mkdir` с опцией `-p`.

Для проверки работы файлового сервера нам необходимо настроить взаимодействие клиентов с нашим сервером для этого необходимо выполнить следующие действия:

1. Создать новую «группу» («team»). Вспомним задание одной из предыдущих работ.
2. Добавить операционную систему сервера (RHEL 4.0).
3. Добавить операционную систему клиента (Windows 2000, XP или др.).
4. Настроить сетевое взаимодействие, включив все виртуальные машины в одну виртуальную сеть.
5. Зададим адреса и маски из одной подсети.

Для сервера используя команду `ifconfig eth0 192.168.0.1 netmask 255.255.255.0`.

А для клиента посредством графического интерфейса – `192.168.0.2 (255.255.255.0)`.

6. Проверьте правильность настройки, выполнив команду `ping [IP-адрес удаленного компьютера]` предварительно отключив firewall Win XP.

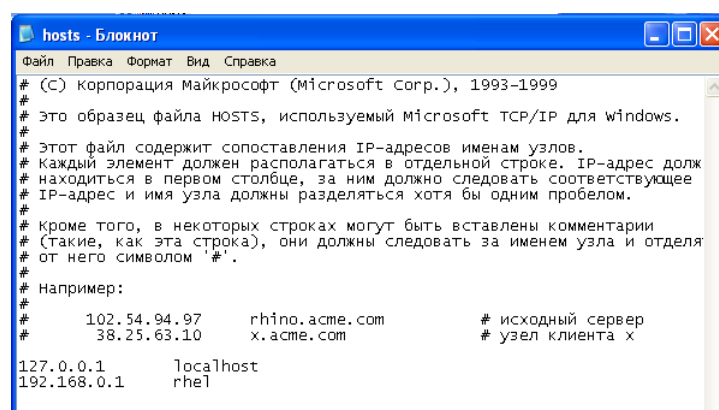
После перезагрузки сервера, если не выполнить дополнительных действий, необходимо снова запустить процесс файлового сервера `service smb start`.

Далее необходимо создать пользователя и группу (в нашем случае `staff`). Добавить пользователя в данную группу.

Добавить данного пользователя в пакет samba и ввести для него пароль (пароли UNIX и samba могут отличаться), используя команду `smbpasswd -a [имя пользователя]`.

На последнем этапе зададим права на папку `/disks/public`. Для членов группы `staff` все права.

А теперь проверка наших трудов. Для этого необходимо зайти на клиентскую систему и в адресной строке ввести `\\192.168.0.1\public`, после чего ввести имя пользователя и пароль. Т.к. службу DNS мы не устанавливали обращение к серверу возможно только с использованием IP-адреса. Для устранения этого неудобства внесем изменения в файл `C:\Windows\System32\drivers\etc\hosts` добавив одной строкой адрес и имя сервера. Узнать и изменить его можно при помощи команды `hostname`.



```
hosts - Блокнот
Файл Правка Формат Вид Справка
# (C) корпорация Майкрософт (Microsoft Corp.), 1993-1999
# Это образец файла HOSTS, используемый Microsoft TCP/IP для Windows.
# Этот файл содержит сопоставления IP-адресов именам узлов.
# Каждый элемент должен располагаться в отдельной строке. IP-адрес долж
# находится в первом столбце, за ним должно следовать соответствующее
# IP-адрес и имя узла должны разделяться хотя бы одним пробелом.
# Кроме того, в некоторых строках могут быть вставлены комментарии
# (такие, как эта строка), они должны следовать за именем узла и отделя
# от него символом '#'.
# Например:
#
# 102.54.94.97 rhino.acme.com # исходный сервер
# 38.25.63.10 x.acme.com # узел клиента x
127.0.0.1 localhost
192.168.0.1 rhel
```

Теперь у нас появилась возможность обратиться к нашему ресурсу таким образом `\\rhel\public`.

Для закрепления материала добавьте еще сетевой ресурс `/disks/share`. И настройте доступ к нему на запись и чтение пользователю `teacher`, а только на чтение всем членам группы `students`. Создайте диск `/disks/text` и разрешите всем пользователям чтение инструкции находящейся на нем (файл `info.txt`).

Проверим полученные результаты, помимо вызова с клиентской машины, на сервере командой `smbclient -L localhost -N`.

Лабораторная работа №4. Установка и обновление программных пакетов. Настройка веб-сервера и файлового сервера.

**Проектирование и администрирование компьютерных сетей. Олейник М.А.**

### Краткие теоретические сведения:

HTTP («протокол передачи гипертекста») — сетевой протокол прикладного уровня передачи данных в виде текстовых сообщений. Основой протокола HTTP является технология «клиент-сервер», то есть предполагается существование потребителей (клиентов), которые инициируют соединение и посылают запрос, и поставщиков (серверов), которые ожидают соединения для получения запроса, производят необходимые действия и возвращают обратно сообщение с результатом.

Протокол HTTP в настоящее время повсеместно используется во Всемирной паутине для получения информации с веб-сайтов.

Основным объектом манипуляции в HTTP являются ресурс, на который указывает URL (ссылка) в запросе клиента. Особенностью протокола HTTP является возможность указать в запросе и ответе способ представления одного и того же ресурса по различным параметрам: формат, кодировка, язык и т.д.

### Ход работы:

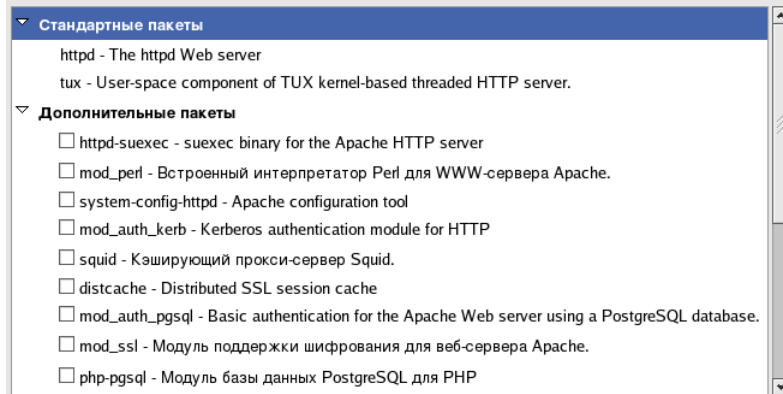
Для разнообразия установим [httpd-сервер](#) при помощи графических средств Красная шапка->Системные параметры->Установка и удаление приложений

**Веб-сервер**



Эти средства позволяют использовать систему как Веб-сервер

Снимем все галочки с дополнительных пакетов (кроме [system-config-httpd](#)), они нам не нужны.



Зайдите в конфигурационный файл веб-сервера [/etc/httpd/conf/httpd.conf](#) и найдите значение параметра [DocumentRoot](#). В данной директории создайте файл `index.html` с содержимым:

```
Hello, <b>WORLD!</b>
```

Запустите процесс веб-сервера `service httpd start`.

Запустите [system-config-httpd](#) и ознакомьтесь с графической возможностью редактировать конфигурационные файлы.

В браузере введите [//localhost](#) и [//rhel](#) (имя сервера) на серверной и клиентской машинах.

### Уровни загрузки (run level) операционной системы RHEL.

#### Краткие теоретические сведения:

Каждый уровень исполнения имеет свой номер – часть этих номеров стандартизована. В любой момент времени система может находиться в одном из них – изменение режима работы производится с помощью перезапуска `init` с указанным номером.

**Уровень 0.** Остановка системы (`halt`) – работа системы должна быть прекращена;

**Уровень 1.** Однопользовательский режим работы – система инициализирует минимум служб и даёт единственному пользователю (как правило, суперпользователю) без проведения аутентификации командную строку. Как правило, этот режим используется для восстановления системы;

**Уровень 2.** Многопользовательский режим – пользователи могут работать на разных терминалах, вход в систему с процессом аутентификации;

**Уровень 3.** Многопользовательский сетевой режим – в отличие от предыдущего уровня, осуществляется настройка сети и запускаются различные сетевые службы;

**Уровень 4.** Не имеет стандартного толкования и практически не используется;

**Уровень 5.** Запуск графической подсистемы – по сравнению с уровнем 3 производится также старт графической подсистемы X11 и вход в систему осуществляется уже в графическом режиме;

**Уровень 6.** Перезагрузка системы – при включении этого режима останавливаются все запущенные программы и производится перезагрузка.

Таким образом, каждый уровень системы подразумевает запуск определённого набора программ, который может быть задан администратором системы. Стартовые скрипты, соответствующие уровням выполнения располагаются в директории [/etc/rc.d](#).

На практике, в серверных системах обычно при старте системы используется 3-й уровень выполнения, в домашних – 5-й.

Конфигурация процесса `init` находится в файле [/etc/inittab](#). Ниже приведён пример такого файла.

```
# Default runlevel. The runlevels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

#### Ход работы:

Измените уровень загрузки по умолчанию в файле [/etc/inittab](#) на 3й и перезагрузите виртуальный сервер. После загрузки верните 5й уровень и снова перезагрузитесь.

Выставьте уровень загрузки для файлового и веб сервисов, используя команду [chkconfig](#):

[chkconfig --list](#) - список всех процессов на всех уровнях

[chkconfig --list | grep smb](#) – интересующие нас процессы

```
[root@localhost disks]# chkconfig --list | grep smb
smb                0:выкл  1:выкл  2:выкл  3:выкл  4:выкл  5:выкл  6:выкл
```

Выполнив команду [chkconfig --level 3 smb on](#) мы включим автоматическую загрузку сервиса `smb` на 3 уровне загрузки ОС.

```
[root@localhost disks]# chkconfig --list | grep smb
smb                0:выкл  1:выкл  2:выкл  3:вкл   4:выкл  5:вкл   6:выкл
```

#### Если злоумышленник сменил пароль root или Вы его просто забыли

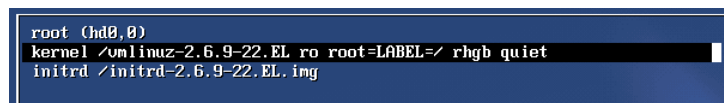
Для выполнения этого задания попросите Вашего соседа исполнить роль злоумышленника, а именно изменить пароль `root` при помощи команды [passwd](#) и перезагрузить виртуальную машину при помощи команды [shutdown -r now](#) или [init 6](#).

Существует механизм позволяющий установить новый пароль пользователя `root`:

1. Когда увидите экран загрузчика GRUB нажмите «пробел».



2. Потом нажмите «e» (edit).

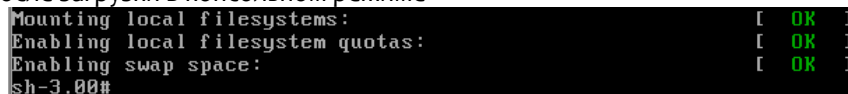


3. Выберите строку с kernel и потом снова нажмите «e» (edit).



4. Допишите слово «single» (однопользовательский режим) в конце строки. Нажмите «enter»
5. Нажмите «b» (boot).

6. После загрузки в консольном режиме



7. Меняем пароль при помощи passwd.
8. И перезагружаемся `shutdown -r now` или `init 6`. Входим с новым паролем в систему.

#### Контрольные вопросы:

1. Сколько уровней загрузки (run level) имеет операционная система RHEL?
2. Почему после перезагрузки samba не работала?
3. Почему обращение к серверу возможно было только с использованием IP-адреса?