

Сервис точного времени

Подготовим Gate к роли сервера точного времени синхронизировав его с внешним сервером времени:

```
# ntpdate time.apple.com
```

Далее установить, настроить и перезапустить NTP сервер.

```
# apt-get install ntp
```

Смотрим конфиг файл, затем сотрем его содержимое:

```
# cat /etc/ntp.conf
```

```
# :> /etc/ntp.conf
```

И добавляем туда сервера для синхронизации:

```
# vi /etc/ntp.conf
```

```
server 0.ru.pool.ntp.org
```

```
server 1.ru.pool.ntp.org
```

```
server 2.ru.pool.ntp.org
```

```
server 3.ru.pool.ntp.org
```

Перезапустим NTP сервер:

```
# service ntp restart
```

Проверим его статус:

```
# service ntp status
```

Через 5-10 минут проверяем:

```
gate# ntpq -pn
```

Синхронизируем server с NTP сервером:

```
# ntpdate gate.corpX.un
```

Проврка результатов синхронизации:

```
# ntptrace
```

выведет расстояние:

Пример:

```
stratum 2, offset 0.006177, synch distance 0.004520
```

ntpd — это демон операционной системы, который устанавливает и поддерживает системное время, синхронизированное с серверами точного времени

```
# apt-get install ntp
```

Файловые сервисы

Импортируем для его установки и настройки server.corpX.un.

Сеть – виртуальный адаптер хоста.

Адрес: 192.168.X.10 (где X – номер стенда).

```
# vi /etc/network/interfaces
```

```
# apt-get update
```

Настройка SAMBA

На **Server**

```
# apt-get install samba
```

Смотрим конфиг файл:

```
# cat /etc/samba/smb.conf
```

Создадим **Публичный каталог доступный на чтение**

Определим где будем хранить файлы:

```
# mkdir /var/samba
```

Что укажем в smb.conf

```
# useradd smb_user -m -s /usr/sbin/nologin
```

-----Пример файла smb.conf-----

Секция [share] позволяет использовать обращение [\\Server\share](#)

[global]

unix charset = UTF-8 # Системная кодировка

dos charset = cp866

workgroup = CORPX

security = user

```
# hosts allow = 192.168.100+X.
```

map to guest = Bad User

[share]

path = /var/samba

guest ok = yes

read only = no

force user = smb_user

от имени которого будут выполняться все операции с файлами, после того как клиент пройдет процедуру аутентификации и укажет действительный пароль.

Делаем:

```
# :> /etc/samba/smb.conf
```

```
# vi /etc/samba/smb.conf
```

```
[global]
```

```
unix charset = UTF-8
```

```
dos charset = cp866
```

```
workgroup = CORPX
```

```
security = user
```

```
map to guest = Bad User
```

```
[share]
```

```
path = /var/samba
```

```
guest ok = yes
```

```
read only = no
```

```
force user = smb_user
```

```
# chown -R smb_user /var/samba
```

```
# chmod 755 /var/samba
```

проверяем доступ с клиента Win XP к [\\server.corpX.un\share](http://server.corpX.un/share)

Создадим там папку и файл

Проверим результат:

```
# ls /var/samba
```

Управление учетными записями

Создадим на **Server** пару пользователей:

```
# useradd user1 -m -s /bin/bash
```

```
# useradd user2 -m -s /bin/bash
```

Задаем пароли (password)

```
# passwd user1
```

```
# passwd user2
```

Просмотр пользователей:

```
# getent passwd
```

```
# cat /etc/nsswitch.conf настраивает откуда брать пользователей.
```

Предоставление доступа к файловым ресурсам на основании пользовательских учетных записей.

Заведем одноименных пользователей на клиенте XP:

```
Net User User1 123 (зададим новый пароль)
```

```
Net User User2 123 /Add
```

Задача: организовать на сервере samba персональные каталоги для пользователей.

БД SAMBA связана с системной БД Unix

```
server# smbpasswd -a user1
```

```
server# smbpasswd -a user2
```

...

Посмотреть созданных пользователей можно:

```
server # pdbedit -w -L
```

Теперь избавляемся от гостевого доступа, редактируя наш конфиг:

```
server # :> /etc/samba/smb.conf
```

```
[global]
```

```
# map to guest = Bad User
```

```
[share]
```

```
# guest ok = yes
```

```
# vi /etc/samba/smb.conf
```

Конфиг примет такой вид:

```
[global]
```

```
    unix charset = UTF-8
```

```
    dos charset = cp866
```

```
    workgroup = CORPX
```

```
    security = user
```

```
[homes]
```

```
    read only = no
```

```
[share]
```

```
    path = /var/samba
```

```
valid users = user1 user2 smb_user
```

```
force user = smb_user
```

```
read only = No
```

Проверяем, заходим на WinXp как user1 и по пути [\\server.corpX.un\](http://server.corpX.un) видим папку user1 (она же home) и общую шару. Под user 2 увидим соотв. папки user2 (home)

сервер FTP:

```
# apt-get install proftpd
```

На клиенте Win XP в файловом проводнике набираем:

```
# ftp://user1@server.corpX.un
```

Синтаксис: ftp://login:password@host:port

Создадим файл на рабочем столе копируем файл на сервер FTP.

Проверим на сервере:

```
# ls ~user1
```

Модуль 7. Сервис HTTP

WWW хостинг

Ставим веб-сервер на Server

```
# apt-get install apache2
```

проверяем доступ с клиента Win:

```
http://server.corpX.un
```

Формат протокол://узел.директория.файл

Для ftp: протокол://логин:пароль@узел/

Настройка сайтов

```
# find /etc/apache2 | wc -l
```

Сайты в каталоге sites-enabled

000-default.conf – сайт по-умолчанию

Пример модификации:

```
# mv /var/www/html/index.html .
```

```
# cp /etc/samba/smb.conf /var/www/html/
```

```
# cp /etc/network/interfaces /var/www/html/
```

проверяем с клиента Win: <http://server.corpX.un>

публичный каталог через http:

Для публикации своего контента, пользователю достаточно создать в своем домашнем каталоге на сервере папку с названием **public_html**

С клиента Win пользователь подключается к своему домашнему каталогу и создает в нем папку **public_html**

Создаем на рабочем столе файл index.html с содержимым в

```
<h1> </h1>
```

Функционал пользовательских каталогов в apache включается скриптом a2enmod:

Делаем:

```
# a2enmod userdir                ##создаст линк
```

```
# service apache2 restart
```

С клиента Win XP: <http://server.corpX.un/~user1/>

Через ftp://user1@server.corpX.un/public_html выложим pict01.bmp

И для скачивания используем ссылку <http://server.corpX.un/~user1/pict01.bmp>

Задача: веб сервер должен быть доступен по имени www.corpX.un , а домашняя страница по имени домена (без www): corpX.un Пользовательские каталоги по **user1.corpX.un**

Создаем алиасы (cname) в dns и увеличиваем soa зоны:

```
$TTL 3h
```

```
@ IN SOA ns. root.ns. (
    199609209 ; serial, todays date + todays serial #
    8H ; refresh, seconds
    2H ; retry, seconds
    4W ; expire, seconds
    1D ) ; minimum, seconds
```

```
@ IN NS ns.
A 192.168.X.10
; MX 1 server
; MX 2 gate
ns IN A 192.168.X.10
gate IN A 192.168.X.1
server IN A 192.168.X.10
```

```
www CNAME server
```

```
user1 CNAME server
```

```
# named-checkconf -z
```

```
# rndc reload
```

```
# nslookup -q=A corpX.un
```

Использование виртуальных хостов

На одном адресе несколько сайтов

Сайты расположены здесь: `ls /etc/apache2/sites-available/`

и становятся доступными, как только скриптом `a2ensite` для них делается линк в `/etc/apache2/sites-enabled`

Создадим свой сайт:

[При обращении на адрес сервера по имени сайта, открывать связанную с ним страницу]

```
# vi /etc/apache2/sites-available/user1.conf
```

```
<VirtualHost *:80>
```

```
    ServerName user1.corpX.un
```

```
    DocumentRoot /home/user1/public_html
```

```
</VirtualHost>
```

```
root@server:~# a2ensite user1
```

```
# service apache2 reload
```

Вернем взад дефолтную станицу сайта апача:

```
# mv index.html /var/www/html/
```

```
# rm /var/www/html/smb.conf
```

```
# rm /var/www/html/interfaces
```

С клиента Win подключаемся по именам `corpX.un`, `www.corpX.un`, `user1.corpX.un`