
# СОЗДАНИЕ PDC (ОСНОВНОГО КОНТРОЛЛЕРА ДОМЕНА) ДЛЯ WINDOWS НА БАЗЕ SAMBA 2.2.5

АНДРЕЙ ГУСЕЛЕТОВ

**Не то чтобы настало время фанатиков от OpenSource, но всё-таки цена программного обеспечения имеет значение. Кроме того, если раньше ПО, построенное по технологии открытых исходников, было уделом узких специалистов, то сейчас оно стало гораздо удобнее в использовании, появляются дружелюбные инсталляционные скрипты, толковая документация и т. д. Люди вроде Столмена и Линуса, которые также борятся за OpenSource, зачастую вдохновляют окружающих своими идеями (впрочем, последний в меньшей степени). И эти самые окружающие (мы с вами) своим энтузиазмом двигаем всё это вперед. Я надеюсь. Итак, разговор пойдет о Samba.**

Samba – ПО, построенное по идеологии открытых исходников, которое позволяет на базе компьютера, работающего под управлением операционной системы UNIX, создать ресурсы, доступные для Windows-машин. Фактически, SAMBA – это UNIX-реализация протоколов SMB или CIFS фирмы Microsoft, предназначенных для доступа по сети к файлам и принтерам компьютеров, работающих под управлением MS Windows. Кроме того, Samba также может действовать как основной контроллер домена на уровне Microsoft Windows NT 4.0 Server. Когда я стал это настраивать, то столкнулся с массой мелких нюансов, нигде ранее не описанных; вот я и хочу помочь всем тем, кто будет делать то же самое – использовать Samba в качестве PDC.

## Начальные требования

Вам понадобится компьютер, работающий под управлением ASPLinux 7.2. Впрочем, для RedHat 7.2 всё будет абсолютно также, те же пути и те же методы настройки. Я выбрал ASPLinux, так как это достаточно дружелюбный и толковый русский дистрибутив, основанный на RedHat. Кроме того, он довольно-таки распространен и если вы захотите всё это точка в точку повторить, то найти ASPLinux будет просто. Там, где какие-то различия всё же имеются, я о них расскажу.

## Инсталляция

Скачиваем SAMBA-2.2.5.rpm для RedHat 7.2 с samba.org. После чего даем команду:

```
>    rpm -ivh samba-2.2.5.rpm
```

Если всё прошло нормально (а с чего бы ему пойти не так), продолжаем далее, иначе – смотрите, что у вас пошло не так, обычно диагностические сообщения менеджера пакетов помогают. Посмотрим, что у нас получилось. Итак,

- man-файлы пошли куда положено – в man;
- запускаемые файлы и библиотеки – в /usr/sbin;
- конфигурационные файлы – в /etc/samba;
- ну и еще есть SWAT (средство администрирования SAMBA через Web-интерфейс).

Впрочем, вы можете всё собрать и из исходников, я такой путь и предпочитаю, но для начала Samba проще поставить из пакета. Для желающих повторить:

- скачиваем samba-2.2.5.tar.gz;
- даем команду:

```
> tar xvzf samba-2.2.5.tar.gz
```

- переходим в каталог с конфигурационными скриптами:

```
> cd samba-2.2.5/source
```

- и запускаем конфигурирование (всё в одну строчку!):

```
> ./configure –prefix=/usr \
>      –bindir=/usr/bin \
>      –sbindir=/usr/sbin \
>      –libexecdir=/usr/libexec \
>      –datadir=/usr/share/samba \
>      –sysconfigdir=/etc/samba \
>      –with-msdfs \
>      –with-configdir=/etc/samba \
>      –with-winbind
```

Ключевые вещи здесь такие: мы хотим, чтобы бинарники были разложены по директориям /usr/bin и /usr/sbin и, кроме того, чтобы все конфигурационные файлы были в каталоге /etc/samba. Если же всего этого не сказать, то по умолчанию Samba станет в /usr/local/samba. И там же, в /lib, будет искать конфигурационные файлы. Мне это кажется несколько неудобным, я предпочитаю хранить все конфигурации либо в /etc, либо в /usr/local/etc (если под FreeBSD). И еще одно: обратите также внимание на параметр with-msdf, который определяет поддержку Microsoft Distributed File System, вполне возможно, что она вам пригодится. Далее делаем

```
> make
```

потом

```
> make install
```

Готово: всё на своих местах. Если что-то пошло не так, еще раз внимательно читайте вывод make! Впрочем, хотел бы обратить внимание вот на что: если говорить о версии 2.2.5 – то всё нормально, а вот версия 2.2.4 у меня в некоторых случаях не воспринимала параметры путей к конфигурационным файлам, и приходилось, после того как configure отработает, руками править make-файл. Так что имейте это в виду.

## Конфигурирование

Ну вот, Samba поставлена, теперь займемся её конфигурированием. Если вы ставили её из rpm, то у вас уже имеются все необходимые конфигурационные файлы в каталоге /etc/samba. А вот если вы её собирали из исходников, то всё придется делать ручками. Впрочем, не расстраивайтесь. Мы тоже всё будем делать с самого начала. Основной конфигурационный файл Samba – smb.conf. Сохраняем его на всякий случай: cd /etc/samba cp smb.conf smb.conf.bak.

После чего очищаем smb.conf и вбиваем туда все, что нам нужно. Структура этого файла очень проста. Он состоит из двух секций: [global], в которой описано всё, что касается настройки программы в целом, и «shares», где перечислены все создаваемые вами общедоступные ресурсы. Описание каждого ресурса начинается с его названия в квадратных скобках, например, [homes], а далее следуют различные опции: путь, права доступа и пр. Некоторые опции дублируются и в глобальной секции, и в каждом ресурсе; причем если в каком-то ресурсе значение опции явно не указано, то берутся соответствующие данные из секции [global]. Если и там их нет, то – по умолчанию. А вот если вам

надо, то вы можете перекрыть для некоторых ресурсов значения, выставленные в секции [global], в явном виде прописав нужные опции в описании этих ресурсов.

Я надеюсь, что вы уже определились с именем, которое дадите своему домену, если нет – то самое время это сделать. Начинаем:

```
# /etc/samba/smb.conf
# SAMBA configuration file
# Created by GUS 04.08.2002
# Last updated : 05.08.2002 by GUS

[global]

; Basic setting for our server
; NetBIOS name for our server
netbios name = DREAM
; workgroup name, here - DOMAIN NAME
workgroup = DREAMHOUSE
; server description string
server string = DREAMHOUSE Primary Domain Controller running
Samba %v
```

Теперь по порядку:

- netbios name = DREAM – наш компьютер будет виден в Windows-сети под названием «DREAM». Кстати, вы можете ничего тут и не писать: если в целом сетевые настройки вашего компьютера правильны, то SAMBA подставит его DNS-имя и будет чудненько работать. Однако лучше подстраховаться, потому что для более ранних версий SAMBA это может и не работать;
- workgroup = DREAMHOUSE – название нашего домена, причем названия компьютера и домена не должны совпадать, их, конечно, можно сделать одинаковыми, но работать нормально оно тогда не будет!
- server string = DREAMHOUSE Primary Domain Controller running Samba %v – это описательная строка, которая будет видна при просмотре сети с Windows-компьютеров. Можете писать тут что угодно, в моем случае написано PDC какого это домена, и что он работает под Samba версии %v, где %v возвращает версию SAMBA.

Обратите внимание на то, что комментариев в конце строк нет, SAMBA плохо к этому относится, и лучше всего их вынести либо на строку выше, либо на строку ниже – это смотря что вы предпочитаете.

Далее добавляем в smb.conf следующее:

```
; PDC settings
os level = 64
domain master = yes
preffered master = yes
local master = yes
```

Вкратце можно сказать вот что: в один момент времени в сети существует какой-то компьютер, который содержит список всех активных в данный момент компьютеров, вот эта машина и называется в английском языке local master browser, естественно, просто так local master browser'ом не становятся: по мере включения и выключения компьютеров проводится процедура выбора, в которой одна из машин побеждает и становится master browser'ом. Процедура выбора в частности основывается на 4 приведенных параметрах (чем выше OS Level, тем выше шансы) и времени. Кроме того, выборы проводятся и по алфавитному порядку NetBIOS-имени компьютера, но это в последнюю очередь. Параметр domain master, в частности, говорит SAMBA изображать основной контроллер домена.

Следующим этапом будет добавление некоторых настроек касательно безопасности:

```
; Security settings
security = user
encrypt passwords = yes
domain logons = yes
hosts allow = 127.0.0.1 10.150.150.
```

где

- security = user – обязательно должно быть user для SAMBA, работающей в качестве основного контроллера домена, на самом деле, как говорится в оригинальном SAMBA-PDC-FAQ, здесь может быть и SERVER и DOMAIN, но не может быть SHARE, однако подробное объяснение этих параметров будет здесь явно лишним. Кому интересно – можете либо поискать это в документации по SAMBA, либо связывайтесь со мной;
- encrypt passwords = yes – включить использование шифрованных паролей, т.е. не посылать их по сети открытым текстом, вообще-то специальные методы работы с паролями применяют только NT-семейство Windows (начиная с NT4.0+SP3), Windows Me. Опять же, хочется подробнее – ищите в документациях Microsoft или Samba;
- domain logons = yes – включает поддержку авторизации в домене;
- hosts allow = 127.0.0.1 10.150.150. – список адресов отдельных компьютеров и подсетей (обычно, конечно, подсетей), которые могут использовать ваш SAMBA-сервер. Вообще-то, 127.0.0.1 (localhost) писать совсем не обязательно, но вот FAQ по SAMBA говорит обратное, да и мой опыт показал то же самое – лучше написать.

Настроим различные параметры авторизации:

```
; Various logon settings

; path - where to store user profiles
logon path = \\%N\profiles\%u
; home directory - where it is, and where it should be
mounted
logon drive = Z:
logon home = \\homes\%u
; default domain logon script - generic script for all users
; NOTE : this is  relative !!! DOS !!! path to the [netlogon]
share
logon script = start.cmd
```

Теперь по порядку:

- logon path = \\%N\profiles\%u – путь к месту хранения профиля подключившегося пользователя;
- logon drive = Z: – буква диска, под которым будет автоматически подключена директория, находящаяся по пути logon home;
- logon home = \\homes\%u – путь к «домашней», т.е. личной директории пользователя;

- logon script = start.cmd — это командный файл, автоматически запускаемый при входе пользователя в домен.

Обратите внимание на две важные вещи: во-первых, это путь относительно ресурса общего доступа NETLOGON предоставляемого КАЖДЫМ контроллером домена. Во-вторых, этот файл выполняется на стороне клиента, а соответственно должен иметь такое расширение, чтобы клиентская ОС узнавала его как выполняемый, т.е. в общем случае это *.BAT-файлы, а для NT – *.CMD-файлы.

Всё, с секцией [global] мы покончили, освежевали и переварили, на очереди – «shares» или, говоря самым могучим языком мира, – общедоступные ресурсы.

Секция описания общедоступных ресурсов:

```
; Netlogon share
[netlogon]
  comment = Network logon service
  path = /home/netlogon
  browseable = No
```

Секция [netlogon] создаёт служебный ресурс для удаленного администрирования. В основном этот ресурс используется администраторами для различных изменений реестра. Его нужно и удобно использовать совместно со стартовым скриптом.

Здесь опять же:

- comment = Network logon service – это комментарий к ресурсу, который вы можете увидеть рядом с именем ресурса в проводнике на Windows-машине;
- path = /home/netlogon – путь к ресурсу;
- browseable = No – ресурс не является просматриваемым по сети, фактически, его могут увидеть только пользователи с правами администратора домена.

Стоит отметить, что некоторые администраторы предпочитают в описание этого ресурса добавить еще и строчки:

```
read only = yes
write list = admin
```

Таким образом, строкой read only = yes ресурс объявляется всем пользователям только для чтения, а строкой write list = admin, некоему гипотетическому администратору дается право записи.

Следущий ресурс – «homes»:

```
; User's home directories
[homes]
        comment = Home Directories
        valid users = %S
        read only = No
        create mask = 0664
        directory mask = 0775
        browseable = No
```

Это тот ресурс, в котором создаются пользовательские директории, здесь пользователь, естественно, имеет смысл – «пользователь домена».

По порядку:

- comment = Home Directories – это я уже говорил – комментарий к ресурсу, который вы можете увидеть рядом с именем ресурса в проводнике на Windows-машине;
- valid users = %S – допускаются только пользователи домена;
- read only = No – ресурс предназначен для записи, а не только для чтения;
- create mask = 0664 и directory mask = 0775 – их следует рассматривать только вместе – подобным образом указанные маски не дают пользователю выбраться выше уровня своей домашней директории. Более детальное рассмотрение данного пункта здесь делать не будем, интересно – пишите;
- browseable = No – и наконец, этот ресурс не просматривается по сети (естественно, незачем обычным пользователям видеть всю папку с пользовательскими директориями).

```
; User's profiles
[profiles]
        path = /home/samba/profiles
        create mask = 0600
        directory mask = 0700
        browseable = No
```

Вот очень важный ресурс, особенно если вы собираетесь реализовать перемещаемые профили пользователей. В этом ресурсе описываются пути и маски создания пользовательских профилей.

Давайте разберемся с каждой строчкой:

- path = /home/samba/profiles – путь к профилям;
- create mask = 0600 и directory mask = 0700 – опять же, рассматриваются только вместе, вот тут давайте поподробнее:
  - маска создания – 0600 – это есть rwx-xxx-xxx, т.е. только пользователь может читать и писать файлы сюда, маска директорий;
  - 0700 – rwx-xxx-xxx – директории должны быть запускаемыми, если их нужно просматривать.
- browseable = No – а вот сам ресурс пользователям видеть ну совершенно не обязательно. Что этой строчкой мы и делаем.

Примечание: будьте очень аккуратны с этим ресурсом, так как Windows NT и Windows 9x по-разному реализуют профили, и, соответственно, могут наблюдаться конфликты и различные необъяснимые файлы и ошибки. Фактически, Windows 9x помещают профиль не в эту директорию, а в домашнюю директорию пользователя.

Последний ресурс:

```
; Printers
[printers]
  comment = All Printers
  path = /var/spool/samba
  printable = Yes
  browseable = No
```

Здесь описываются общие настройки принтеров. К сожалению, детальное описание настройки принтеров для печати из-под Windows на SAMBA-машину я не смогу

сделать. Халтурить мне не позволяет совесть, а сделать так как надо – это было бы очень объемно. Но если будут пожелания – можно будет сделать отдельную статью, посвященную целиком этому вопросу.

Кратко:

- comment = All Printers – комментарий ресурса;
- path = /var/spool/samba – путь к каталогу спулера печати;
- printable = Yes – сюда можно печатать;
- browseable = No – этот ресурс непросматриваемый.

Вроде бы всё. Но к редактированию этого файла мы еще вернемся. Сейчас сделаем необходимые директории и пользователей и – пробный запуск.

## Создание необходимых административных директорий

Предпочтительно сделать две группы пользователей (UNIX-группы!), одну – для адмнистраторов, другую – для компьютеров (да, именно так, компьютер как член домена Windows NT, а в нашем случае – SAMBA, для UNIX выглядит просто как пользователь, с особенным образом заданным именем и паролем). GID следует выбрать так, чтобы его значение не конфликтовало с другими идентификаторами групп в вашей системе, поскольку мы предполагаем установку на «свежепроинсталлированный» ASPLinux, то безопасным будет взять 200 и 201.

Группу администраторов назовем admins:

```
> groupadd -g 200 admins
```

а группу с именами машин назовем machines:

```
> groupadd -g 201 machines
```

Обратите внимание – для RedHat Linux (а не для ASPLinux) эти команды выглядели бы так:

```
> group -g 200 admins
> group -g 201 machines
```

Следующим шагом будет создание необходимых директорий и – самое главное – верных прав на них:

```
> mkdir -m 0775 /home/netlogon
> chown root.admins /home/netlogon
```

Надеюсь, здесь всё прозрачно, владелец – root из группы admins, чётко это запомните, потому как позднее, уже при присоединении машины к домену это вам понадобится.

```
> mkdir /home/samba /home/samba/profiles
> chown 1757 /home/samba/profiles
```

На директорию /home/samba/profiles, как я уже говорил выше про пользовательские директории, сделаны такие права, чтобы пользователь не выходил за пределы отведенной ему иерархии и случайно или преднамеренно не повредил информацию других пользователей.

## Создание пользовательских и машинных «бюджетов»

Слово «бюджет» я избрал для замены англоязычного термина account, мне кажется, что этот вариант наиболее подходящий по смыслу.

Начнем с более простого – создания бюджета пользователя. Создание бюджета пользователя проходит в два этапа. Вам нужно:

- создать бюджет пользователя в UNIX;
- повторить те же действия для самой SAMBA.

И не забудьте – это всё еще и нужно будет поддерживать в синхронизированном виде!

Приступим: сделаем пользователя «gus» сначала в системе:

```
> useradd gus
> passwd gus
> New password:
> Retype new password:
> passwd: all authentication tokens updated successfuly
```

теперь в SAMBA:

```
> smbpasswd -a gus
> New SMB password:
> Retype new SMB password:
> Added user gus
```

Один нюанс: иногда, и это зависит от версии SAMBA, пользователь добавляется в домен, но в состоянии «выключен». И его бюджет нужно принудительно включить следующей командой:

```
> smbpasswd -e gus
```

Теперь, в случае если пользователь захочет сменить свой пароль с Windows-машины, то произойдет следующее: SAMBA-то у себя его поменяет, а вот в UNIX'е он останется старым. Что делать? К счастью, решение есть.

## Синхронизация пользовательских данных между SAMBA и UNIX

Для того чтобы это работало, следует внести в секцию [global] следующие строчки:

```
  ; UNIX password syncing
  unix password sync = Yes
  passwd program = /usr/bin/passwd %u
  passwd    chat    =    *New*UNIX*password*    %n\n
*Retype*new*UNIX*password* %n\n *Enter*new*UNIX*password* %n\n
  *Retype*new*UNIX*password*     %n\n     *passwd:
*all*authentication*tokens*updated*successfully*
```

Обратите внимание на строку passwd chat: все, что после символа «равно», должно быть набрано в одну строку!

## Создания бюджетов компьютеров

Для создания бюджетов компьютеров может применять-

ся два способа: ручной – перед присоединением компьютера к домену; и автоматический – во время присоединения компьютера к домену. Сразу нюанс: автоматический способ в том виде, в котором его реализацию предлагают создатели SAMBA, в ASPLinux не делается.

### Вариант 1
### Ручное создание бюджетов машин

Вот тут начнем наоборот – для RedHat 7.2 всё просто – даем команду:

```
>/usr/sbin/useradd -g machines -d /dev/null -c «machine
nickname» -s /bin/false machine_name$
> passwd -l machine_name$
> Changing password for user machine_name$
> Locking password for user machine_name$
```

где machine_name – это имя машины, а после него – знак доллара, это обязательно, и вот в этом-то и загвоздка. ASPLinux знак доллара не воспринимает. Поэтому делаем так:

```
> /usr/sbin/useradd -g machines -d /dev/null -c «machine
nickname» -s /bin/false machine_name
> passwd -l machine_name
> Changing password for user machine_name
> Locking password for user machine_name
```

затем

```
> vipw
```

и руками редактируем файл паролей, (наш компьютер будет последний), добавляем там к имени компьютера $. Вот тут-то вам и пригодится так рекомендуемое всеми для UNIX знание редактора VI. Далее даем команду:

```
>smbpasswd -a -m machine_name
```

где machine_name – NetBIOS-имя машины; обратите внимание, что здесь имя машины без знака доллара, так как SAMBA опознает, что это компьютер по ключу -m.

Внимание! После того как вы сделали бюджет компьютера, настоятельно рекомендуется сразу же присоединить этот компьютер к домену! Почему? Потому что когда клиент подсоединяется к домену, он на самом деле поменяет пароль и так называемый «секрет» на SAMBA, т.е. некий уникальный ключ. Если вы этого сразу же не сделаете, то имейте в виду: что тогда в этот промежуток времени в домен может подключиться любой компьютер с таким же NetBIOS-именем. А это – огромная дырка в безопасности.

### Вариант 2
### Автоматическое создание бюджетов машин

Для RedHat 7.2 необходимо в секцию [global] файла smb.conf внести следующую строчку:

```
add user script = /usr/sbin/useradd -d /dev/null -g machines
-s /bin/false -M %u
```

а вот для ASPLinux 7.2 – увы, если всё-таки хочется, то вам придется самостоятельно написать скрипт или программу, который (-ая) делает следующее:

- добавляет необходимого пользователя;
- делает для него passwd -l;
- блокирует базу паролей;
- вносит необходимые изменения (добавляет знак $);
- обновляет базу паролей и разблокирует её.

Этого мы здесь делать не будем, есть желание – пишите, поделюсь. Хотя я настоятельно рекомендую вместо этого использовать RedHat 7.2 либо 7.3, так как в ASPLinux даже 7.3 таже самая проблема.

Один нюанс: в Samba 2.2.1-2.2.5 только учетная запись root может быть использована для создания машинных учетных записей. Следовательно, необходимо в SAMBA создать запись для root. Более ранние версии SAMBA я не проверял. Пароли root для UNIX и SAMBA должны различаться по соображениям безопасности. И всё это – вопреки FAQ по SAMBA, где такого не сказано, но работает оно, по крайней мере, на текущую версию именно так.

Всё! Настройка закончена. Теперь запускаем SAMBA.

Фактически SAMBA состоит из двух «демонов» – smbd и nmbd. Первый из которых отвечает, собственно говоря, за ресурсы; а второй – за имена NetBIOS, поэтому добавляем в стартовые файлы следующие строчки:

```
> nmbd -D
> smbd -D
```

и перегружаем компьютер. Почему я не говорю как это сделать? Потому что методов может быть много – я вообще предпочитаю сделать отдельную директорию. Откуда мой скрипт, помещенный в /etc/rc.d/rc.local, вылавливает всё и по очереди запускает – a'la FreeBSD. Возможно, вы делаете по-другому, вот и не буду навязываться. Вот теперь точно всё – работайте на здоровье.

## Заключение

На самом деле статья получилась не настолько полная, насколько хотелось бы. Не хватает настройки принтеров, тонкой настройки прав доступа (через ACL), скрипта автоматического создания машинных бюджетов, описания скриптов – до и после подключения клиентов, не хватает описания winbind и собственно подключения клиентских машин. Также можно было бы рассмотреть распределенную файловую систему Microsoft – MS DFS. Но тогда статья получилась гораздо бы более тяжелой и объемной. Если будут пожелания – постараюсь их выполнить. Либо на страницах этого журнала, либо по электронной почте, в последнем случае я отвечаю всегда, но иногда могу отложить решение вашей проблемы на 3-5 дней, о чем всегда вам сообщу.

**P. S.** 20 ноября 2002 года команда разработчиков SAMBA выпустила версию 2.2.7, которую и желательно использовать вместо описанной в статье: в ней ликвидирована одна потенциально опасная дыра при проверке длины запроса на смену зашифрованного пароля от клиента. Подчеркну, что готового кода, использующего данную уязвимость, не существует, – ошибка была найдена самими разработчиками. Подробнее смотрите на сайте разработчиков.

# АБСОЛЮТНО ВСЕ О X.25

СЕРГЕЙ РОПЧАН

*Сети, построенные на основе технологии X.25, на сегодняшний день являются самыми распространенными сетями с коммутацией пакетов, используемыми в качестве корпоративных. Основная причина такой ситуации состоит в том, что долгое время сети X.25 были единственными доступными сетями с коммутацией пакетов коммерческого типа, в которых давались гарантии коэффициента готовности сети. Кроме того, они достаточно надежно работают даже на нестабильных линиях благодаря протоколам с установлением соединения и коррекцией ошибок на двух уровнях – канальном и сетевом.*

Стандарт X.25 трактуется как “интерфейс между оконечным оборудованием данных и аппаратурой передачи данных для терминалов, работающих в пакетном режиме в сетях передачи данных общего пользования”. Он был разработан комитетом CCIT в 1974 году и пересматривался несколько раз. Стандарт наилучшим образом подходит для передачи трафика низкой интенсивности, характерного для терминалов, и в меньшей степени соответствует более высоким требованиям трафика локальных сетей. Как видно из названия, стандарт не описывает внутреннее устройство сети X.25, а только определяет пользовательский интерфейс с сетью. Взаимодействие двух сетей X.25 определяет стандарт X.75.

Технология сетей X.25 имеет несколько существенных признаков, отличающих ее от других сетевых технологий:

- наличие в структуре сети специального устройства PAD (Packet Assembler Disassembler), предназначенного для выполнения операции сборки нескольких низкоскоростных потоков байт от алфавит-