

[Q]: Как скрестить SMB с UNIX'ами?

[A]: Andrew Kolchoogin (2:5020/290.22)

Тут, как я погляжу, в последнее время появился живой интерес к тому, как скрестить SMB-сеть с UNIX'ами. 😊 Я третий год экспериментирую с этими вещами, и вот что я выяснил.

## 1. Терминология.

SMB - Server Message Block (альтернативно: Session Message Block) -

протокол работы файловых серверов. В качестве underlying-level использует NetBIOS. Был разработан фирмой Microsoft около десяти лет назад для своего программного продукта "Microsoft LAN Manager".

Как и любой другой протокол, со временем SMB развивался, и в данный момент

существует пять его модификаций:

a) CORE -- первоначальная модификация протокола. Отличается тем, что

\_полностью\_ завязан на MS DOS и поддерживает \_только\_ share-level access control и имена файлов в формате 8.3.

b) COREPLUS -- доработка протокола, вышедшая после появления Microsoft OS/2

V1.1 с HPFS. Поддерживает длинные имена (до 256 символов).

c) LANMAN1 -- первая попытка создания нормального серверного протокола.

Намёки на user-level access control, длинные имена. Поворот SMB-протокола лицом к пользователю связано с включением фирмы IBM в альянс разработчиков. 😊

d) LANMAN2 -- первая нормальная реализация SMB. Используется в полуосном

LAN-сервере.

e) NT1 -- из названия понятно, кем сделана эта доработка и где она

используется. 😊

Две последних модификации SMB-протокола поддерживают достаточно забавную

систему аутентикации - network logons. Microsoft и IBM считают, что network logon - рулез фарева, а вот юниксоиды считают, что network logon must die. И вот почему: UN\*X, в отличие от OS/2 и Windows 95/NT, является многопользовательской операционной системой, а network

logon с этим интерферирует. Да, в Windows NT есть "connect as", но если ты залогинился в сеть администратором, то при попытке поставить в это поле имя любого сетевого пользователя с меньшими правами система сразу посылает по факсу: "Credentials supplied are conflicting with existing set of credentials". Разумеется, верно и обратное. "вы, с этим приходится смириться.

SMB-протокол дал начало ещё одной разработке -- CIFS, Common Internet File

System. Это тоже сетевой протокол, но с изменениями, учитывающими тайм-ауты интернетовских соединений и т.д. Конкурирует с Sun'овским WebNFS. Причём, пока успешно.



## 2. Требования.

Как уже было сказано в предыдущем пункте, SMB требует для своей работы

NetBIOS. MS DOS, OS/2, Windows 95/98 и Windows NT могут рассылать NetBIOS-датаграммы непосредственно через EtherNet. Ни одна из известных мне UN\*X-подобных операционных систем этого не умеет. Но NetBIOS допускает инкапсуляцию себя в любые другие сетевые протоколы, в том числе, IPX и TCP/IP, что и используется в реальной жизни.

Поэтому, перед тем, как начать возиться с UN\*X'ами, на не-UN\*X-системах

следует установить поддержку NetBIOS over TCP/IP, без неё ничего работать \_точно\_ не будет. Тем не менее, есть один подводный камень: любой NetBIOS over TCP/IP-узел относится к одной из трёх категорий:

a) P-Node -- (Point-to-Point) узел использует для трансляции NetBIOS-имён в

IP-адреса используется специальный сервер. Виндозники его называют WINS (Windows Internet Name Service)-сервер, юниксоиды -- NetBIOS Name Server.

b) B-Node -- (Broadcast) узел использует для трансляции имён более простой

алгоритм -- NetBIOS Broadcast Receive.

c) H-Node -- (Hybrid) узел использует оба метода.

Для сведения: Windows всех версий -- это P-Node, в OS/2 это настраивается.

## 3. UN\*X-сервер для SMB-сетей.

В солнечной Австралии, где много кенгуру и диких кроликов, разрабатывается

бесплатный сервер для SMB-клиентов. Называется он Samba. Последняя версия на момент написания этого письма -- 1.9.18 patchlevel 10. Состоит он из нескольких частей.

a) Сервер. Сервером служит `smbd` -- SMB Демон. Его можно настроить

практически на все случаи жизни. Самое тёмное место в его настройке, разумеется,

аутентикация пользователей. Она может быть трёх видов: share-level, user-level и server-level. С share-level и user-level всё понятно: в одном случае прописывается пароль к ресурсу, в другом – создаётся файл, похожий на /etc/master.passwd, где самба хранит пароли. Server-level – это маленький извратик. Самба смотрит на имя пользователя в /etc/passwd, а пароль его

спрашивает у другого сервера. OS/2 и Windows NT катит, проверено. 😊 Правда, если пользователя в /etc/passwd нет, то ему уже ничего не поможет. 😊

b) NetBIOS Name Server/Datagram Distributor. Это nmbd. Предназначен для тупых виндов (см. предыдущее письмо).

c) SMB Networks Client. Это ftp-like программа, позволяющая обмениваться файлами между локальной машиной и SMB-серверами. Называется smbclient.

#### 4. Проблемы криптографии.

Основные геморрой в скрещивании SMB-серверов и клиентов начинаются тогда, когда приходит понимание того, что каждая ОС шифрует пароли по-своему.

- a) Windows NT (до SP3) -- пароли\_не\_шифруются.
- b) Windows NT SP3 или выше -- MD5.
- c) OS/2 -- DES
- d) Windows 95 -- MD5

Ну, про DES и про то, как его вывезти из Америки, все знают. :-) Именно

по-этому так сложно скрестить OS/2 и smbд. Его для этого нужно пересобирать с -ldes. И никак иначе, увы. 😞 Но после перекомпиляции у меня всё работало – и OS/2, и Windows NT, и Samba.

#### 5. SMB-клиент.

Есть на свете замечательный человек -- Линус Торвальдс. Он написал весёлую

операционную систему – Пингвиникс... 😊 И есть у Пингвиникса smbfs. Она, естественно, kernel-mode, и завязана на Линуксовое ядро. Но умные люди её оттуда выдрали и приписали к ней NFS-интерфейс. После чего появилась противоположность самбе – румба. 😊

Разумеется, румба обладает всеми достоинствами NFS-серверов -- в частности, способностью завешивать машину, если NFS-сервер куда-то делся. Кроме всего прочего, запись файлов через 3 уровня VFS и тормознутый NetBIOS over TCP/IP добавляет удовольствия

от работы со столь быстросействующей сетевой файловой системой.



Шли годы, развивалась и румба. И появились два её суперседера -- Sharity и

Sharity-Light. Sharity – shareware-продукт с бесплатной лицензией для институтов. Поставляется он в исходниках, так что при желании всю защиту оттуда можно выдрать. Но это нехорошо.



Sharity-Light -- бесплатный, но он базируется на SMBFS, поэтому гораздо

более тормознутый.

Сам Sharity (не Light) -- довольно умный программный продукт. Во-первых, он позволяет использовать \_несколько\_ сетевых логонов от разных пользователей. Как – представления не имею. Но может. Естественно, auto-logon, reconnect и т.д., всё это имеется.



6. Как всё-таки поступить?

Если есть возможность -- переходите на UN\*X целиком. Если есть желание

повозиться – попробуйте CFS (Coda File System), у неё есть клиент под Windows 95 и UN\*X. Ну, а коли нужно именно SMB – попробуйте перетащить под UN\*X сервера. Samba работает гораздо более устойчиво, чем UN\*X'овые клиенты, да и в последних версиях самбы появился Domain Logon.

From:  
<http://www.osfree.org/doku/> - **osFree wiki**

Permanent link:  
<http://www.osfree.org/doku/doku.php?id=ru:os2faq:os2comm:os2comm.072>

Last update: **2014/06/20 05:08**

