

mount_smbfs - монтирование форточных шар.

Автор: Administrator

07.01.2010 22:02 - Обновлено 28.05.2010 13:22

mount_smbfs - монтирование форточных шар.

Автор: lissyara.

Оригинал: http://www.lissyara.su/articles/freebsd/file_system/mount_smbfs/

Жила себе сетка из виндовых машин - 2 сервака, три десятка узеров. Всё по уму - контроллер домена одна машина (Win2003), сервер БД и документов - другая (Win2000). Доки узеров и отделов на одном сервере, стриммер на другой. По ночам запускалось задание из планировщика, и на стриммер архивировались документы и БД. Через полгода илдиллия закончилась - пользовательские документы распухли до неприличных размеров, и за ночь всё записаться не успевало - в 9 утра народ на работе а залогиниться не может - контроллер домена слишком занят архивацией, чтобы отвлекаться на пользователей :)

Выбил денег на отдельный сервер - под архивацию. Купили - AMD64 3200+, гектар ОЗУ, 2 винта 40Gb и 300Gb. Ну, разумеется ставим FreeBSD :) Ввиду того, что моей любимой 4.11 в 64-х разрядном исполнении нету, а перекомпилировать вообще всё не было желания - решил поставить глюкавенькую FreeBSD6.0 - ибо машина будет внутри локалки, нестрашно.

Короче. :) Смысл всего - в примонтировании виндового раздела на FreeBSD. Для начала, как обычно, попробовал врезать поддержку SMB в ядро - попытался пересобрать со следующими опциями ядро: option NETSMB # SMB/CIFS requester option SMBFS # SMB/CIFS filesystem

Непрокатило. Сборка вылетала на ошибках. Пришлось загрузить модулем, добавив такую строку в /boot/loader.conf smbfs_load="YES"

Чтобы не перезагружаться можно подгрузить модуль `руками` :/usr/home/lissyara/>su
Password:

```
/usr/home/lissyara/>kldload smbfs  
/usr/home/lissyara/>
```

Всё. Команда mount_smbfs должна работать. Проверяем: /usr/home/lissyara/>df -h

```
Filesystem  Size  Used Avail Capacity Mounted on  
/dev/ad0s1a 496M 122M 334M 27% /  
devfs      1.0K 1.0K 0B 100% /dev  
/dev/ad0s1d 4.8G 42K 4.5G 0% /tmp  
/dev/ad0s1e 24G 3.4G 19G 15% /usr  
/dev/ad1s1d 271G 185G 64G 74% /usr/local/backup  
/dev/ad0s1f 4.6G 1.1G 3.1G 27% /var  
/usr/home/lissyara/>
```

```
/usr/home/lissyara/>mount_smbfs -E koi8-r:cp866 -W name_of_my_domain
```

mount_smbfs - монтирование форточных шар.

Автор: Administrator

07.01.2010 22:02 - Обновлено 28.05.2010 13:22

```
? //liss@server1/out /mnt
Password:
/usr/home/lissyara/>df -h
Filesystem      Size  Used Avail Capacity  Mounted on
/dev/ad0s1a      496M  122M  334M   27%   /
devfs            1.0K  1.0K   0B  100%  /dev
/dev/ad0s1d      4.8G   42K  4.5G    0%  /tmp
/dev/ad0s1e      24G   3.4G  19G   15%  /usr
/dev/ad1s1d      271G  185G   64G   74%  /usr/local/backup
/dev/ad0s1f      4.6G   1.1G  3.1G   27%  /var
//LISS@SERVER1/OUT  112G   89G   23G   80%  /mnt
/usr/home/lissyara/>cd /mnt
/mnt/>ls
1C          сканирование0003.jpg
24-01-2006.zip  tmp
алена1.tif    усилитель
пробный.jpg   фото.ppt
коннект.txt   футбол 16.09.05
/mnt/>
```

Пароль вводил доменный. Всё подмонтировалось. Параметр -E koï8-r:cp866 означает, что будет использоваться перекодировка, для имён файлов, из cp866 в koï8-r (ведь Вы уже русифицировали консоль, верно? :)). Надо заметить, что удалённая кодировка может быть и cp1251... Пару раз сталкивался - было вперемешку - часть файлов cp1251, а часть cp866 - встревон. Полный список кодировок поддерживаемых вашей машиной можно получить из вывода команды locale:/usr/home/lissyara/>locale -m

```
ARMSCII-8
Big5
Big5HKSCS
CP1131
CP1251
CP866
CP949
GB18030
GB2312
GBK
ISCII-DEV
ISO8859-1
ISO8859-13
ISO8859-15
ISO8859-2
ISO8859-4
ISO8859-5
ISO8859-7
ISO8859-9
```

mount_smbfs - монтирование форточных шар.

Автор: Administrator

07.01.2010 22:02 - Обновлено 28.05.2010 13:22

```
KOI8-R
KOI8-U
PT154
SJIS
US-ASCII
UTF-8
eucCN
eucJP
eucKR
/usr/home/lissyara/>
```

Также можно коннектится гостем:
/usr/home/lissyara/>mount_smbfs -E koi8-r:cp866 -W тут_что_угодно

```
? //guest@server1/distrib /mnt
```

Password:

```
/usr/home/lissyara/>df -h
```

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad0s1a	126M	44M	72M	38%	/
/dev/ad0s1e	504M	7.4M	456M	2%	/tmp
/dev/ad0s1f	24G	15G	7.4G	66%	/usr
/dev/ad0s1g	12G	1.2G	9.5G	11%	/var
procfs	4.0K	4.0K	0B	100%	/proc
/dev/ad2s1e	110G	35G	67G	34%	/var/db/mysql
//GUEST@SERVER1/DISTRIB	112G	89G	23G	80%	/mnt

```
/usr/home/lissyara/>
```

На запросе пароля я просто нажал "Enter". Кстати, операции эти надо делать от рута - от простого пользователя получите ошибку:
/usr/home/lissyara/>mount_smbfs -E

```
koi8-r:cp866 -W name_of_my_domain
```

```
? //liss@server1/distrib /mnt
```

Warning: no cfg file(s) found.

```
mount_smbfs: can not setup kernel iconv table (koi8-r:tolower):
```

```
syserr = Operation not permitted
```

```
/usr/home/lissyara/>
```

Всё это конечно хорошо - но монтированием, копированием и архивацией должен был заниматься скрипт. А он пароль ввести не мог :) Надо было чё-то думать. Курение манов дало ключевое слово - nsmb.conf. Вот тут-то и была главная засада. Я полдня убил на этот файл - не хотел коннектится с его помощью и всё тут. пароль неверный... Сидел смотрел tcpdump`ом - думал может не понимает какой домен... Всё оказалось гораздо проще - я вводил названия серверов, домена, и имя пользователя маленькими буквами... А надо было большими... Итого у меня получилось:[default]
workgroup=NAME_OF_MY_DOMAIN

mount_smbfs - монтирование форточных шар.

Автор: Administrator

07.01.2010 22:02 - Обновлено 28.05.2010 13:22

```
[SERVER2]
```

```
addr=192.168.20.2
```

```
[SERVER2:BACKUP]
```

```
# charset - но с ним косяки - под фрэй всё пучком, всё видно,
```

```
# всё читается, но в архивах, если их открывать из-под форточек,
```

```
# русские имена слетают... Как вариант, если надо открывать под
```

```
# окошками - поставить cp866:cp1251 (именно так! :))
```

```
charset=koi8-r:cp866
```

```
password=password_usera_backup
```

После чего всё прекрасно монтируется такой командой: /usr/home/lissyara/>mount_smbfs

```
-N //BACKUP@SERVER2/Bases /mnt
```

```
/usr/home/lissyara/>df -h
```

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/ad0s1a	496M	122M	334M	27%	/
devfs	1.0K	1.0K	0B	100%	/dev
/dev/ad0s1d	4.8G	42K	4.5G	0%	/tmp
/dev/ad0s1e	24G	3.4G	19G	15%	/usr
/dev/ad1s1d	271G	210G	39G	84%	/usr/local/backup
/dev/ad0s1f	4.6G	1.1G	3.1G	27%	/var
//BACKUP@SERVER2/BASES	92G	78G	14G	85%	/mnt

```
/usr/home/lissyara/>
```

Всё пожалуй. Если кому-то нужны будут подробности - man :)