

Samba: оптимизация

[global]

1. Параметр `wide links` определяет, как Samba будет следовать по символическим ссылкам. Если `wide links=no`, то Samba не будет следовать по символическим ссылкам вне экспортируемой области. Сначала Samba следует по символической ссылке, а затем выполняет так называемый `directory path lookup` (системный вызов, определяющий, где завершилась ссылка). Данная операция подразумевает на 6 системных вызовов больше, нежели в случае, если

`wide links=yes`.

Учитывая, что подобных операций делается очень много, то выключение `wide links` снижает производительность Samba приблизительно на 30%.

2. При использовании TCP/IP, если размер запросов и ответов не фиксирован (как в случае с Samba), рекомендуется применять протокол TCP с опцией `TCP_NODELAY`. Для этого в файл `smb.conf` нужно добавить строку:

```
socket options = TCP_NODELAY
```

Тесты показывают, что Samba при больших нагрузках работает в 3 раза быстрее, чем без указания этих опций.

3. Если Samba используется в локальной сети (в большинстве случаев так оно и есть), рекомендуется еще указать опцию `IPTOS_LOWDELAY`:

```
socket options = IPTOS_LOWDELAY TCP_NODELAY
```

4. Если есть желание «выжать» из Samba еще больше, тогда установите следующие параметры буферизации: `SO_RCVBUF=8192` `SO_SNDBUF=8192`. Например:

```
socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF= 8192
```

или

```
socket options = TCP_NODELAY IPTOS_LOWDELAY SO_RCVBUF=65536  
SO_SNDBUF=65536
```

Смысл параметров:

- TCP_NODELAY отвечает за задержку пакетов. Начиная с Samba 2.0 данный параметр устанавливается по умолчанию, в более старых версиях его установка может привести к ускорению работы на 30%.
- IPTOS_LOWDELAY— еще один параметр для оптимизации пропускной способности. Но он затрагивает работу роутеров и конечных систем, а не сервера. Этот параметр должен использоваться вместе с TCP_NODELAY и может обеспечить прирост производительности до 20%.
- Опции SO_RCVBUF и SO_SNDBUF определяют максимально возможный размер буферов приема и передачи Samba. Уменьшение размера буферов приводит к увеличению фрагментации пакетов, увеличение размера — к уменьшению фрагментации.

PS. В версии 4.7.x — включение последней опции, как минимум, не дало производительности, а в некоторых случаях — понижало ее.

*** Без опции ***

```
svm@note:/mnt/share$ time cp test.file_1G ~  
real 1m32,656s  
user 0m0,017s  
sys 0m4,358s
```

```
svm@note:/mnt/share$ time cp test.file_1G ~  
real 1m31,091s  
user 0m0,029s  
sys 0m4,387s
```

```
svm@note:/mnt/share$ time cp test.file_1G ~  
real 1m34,893s  
user 0m0,029s
```

sys 0m4,434s

```
svm@note:/mnt/share$ time cp test.file_1G ~
real 1m32,069s
user 0m0,000s
sys 0m4,482s
```

***SO_RCVBUF/SO_SNDBUF=8192

```
svm@note:/mnt/share$ time cp test.file_1G ~
real 1m43,103s
user 0m0,005s
sys 0m4,411s
```

***SO_RCVBUF/SO_SNDBUF=65536

```
svm@note:/mnt/share$ time cp test.file_1G ~
real 1m32,687s
user 0m0,017s
sys 0m4,526s
```

```
svm@note:/mnt/share$ time cp test.file_1G ~
real 1m45,322s
user 0m0,012s
sys 0m4,531s
```