Screen - дисконнекты теперь не страшны!

Опубликовано muff в Сб, 2009-08-15 03:00

Пора уже привыкнуть работать в screen... Удобная штука, рекомендую всем.

Для начала мой .screenrc (должен лежать в домашнем каталоге пользователя)

```
# less .screenrc
vbell off
activity 'activity in window %n'
bell_msg 'bell in window %n'
nethack on
autodetach on
startup_message off
defscrollback 10000
caption always
caption string "%{rk} %c %{dd} %{+b M}%n %{-b dd}%-w%{+b B.}%n* %t%{-}%+w%<"
```

Немного информации о screen:

SCREEN - полноэкранный и достаточно мощный консольный оконный менеджер с поддержкой скроллинга и поиска в окне и функцией копирования-вставки между ними. Наиболее интересная функция данного менеджера заключается в том что Вы можете в любой момент отсоединиться от своего screen`а и закрыть сеанс работы в шеле. После этого Вы можете присоединившись к screen`у вновь продолжить свою работу с того места где Вы остановились.

Устанавливаем screen из портов.

```
# cd /usr/ports/sysutils/screen
# make install clean
```

Выбираем необходимые опции

```
Options for screen 4.0.3_6


[ ] CJK   Treat CJK ambiguous characters as full width[X] INFO  Build and install in
fo documentation[X] MAN  Build and install man pages[X] NETHACK Enable nethack-style
 messages[X] XTERM_256 Enable support for 256 colour xterm[X] HOSTINLOCKED Print use
r@host in locked message[X] SHOWENC Show encoding on the status line
```

И по окончанию установки не забываем перечитать пути

```
# rehash
```

Итак, разберем поподробнее, как же работать с данным чудом. Для начала заглянем в конфигурационный файл .screenrc, который находится в домашнем каталоге. Если его там нет, можно скопировать общесистемный файл screenrc который находиться в каталоге /usr/local/etc.

Что же полезного можно включить в конфигурационном файле? Все опции можно изменить во время работы. Для этого нажмите Ctrl+a: и введите название параметра и его значение.

vbell off - управляет визуальным звонком. Если данный параметр будет включен (on) то звонок будет отображаться как вспышка на экране.

activity 'activity in window %n' - сообщение которое будет выводиться при включенном

режиме мониторинга за окном. Полезно если Вы ждете какого либо действия в окне.

bell_msg 'bell in window %n' - сообщение которое выведется на Ваш экран в случае получения screen`ом звукового сигнала в каком либо окне.

nethack on - изменяет стиль текста выводимых сообщений на стиль знаменитой игрушки NetHack. Почувствуйте себя в подземельях... ;)

autodetach on - если по какой то причине соединение с управляющим процессом будет потеряно, то после востановления работа в screen может быть возобновлена. В обратном случае (off) - screen будет уничтожен со всеми дочерними окнами и процессами.

startup_message off - выключает сообщение об авторских правах при первом запуске screen`a.

defscrollback 10000 - количество строк по умолчанию для буфера прокрутки.

caption always - показывает заголовки окна в строке статуса.

caption string "%{rk} %c %{dd} %{+b M}%n %{-b dd}%-w%{+b B.}%n* %t%{-}%+w%<" - форматирование строки статуса. Данный набор символов приведет к тому что в строке статуса будет отображаться время и цветом выделяться активное окно.

После запуска screen создаст одно окно с Вашим shell. В последствии вы сможете создать дополнительные окна. Все нажатия клавиш передаются текущей программе в окне. Ограничение накладывается только на управляющую последовательность самого менеджера. Данная последовательность Ctrl+a. Для того что бы передать приложению данную последовательность Вам нужно нажать Ctrl+a и сразу a. В остальном - абсолютно никаких ограничений. Единственно что тип терминала должен быть VT100 совместим для правильной передачи нажатий при удаленной работе.
Тип терминала передаваемый приложению в окне screen - так и называться screen. Если Ваше приложение не поддерживает данный тип - его всегда можно изменить путем изменения переменной TERM.

КРАТКАЯ СВОДКА КОМБИНАЦИЙ КЛАВИШ ПРИ РАБОТЕ

Для создания нового окна - **Ctrl+a c** (create).

Для переключения между окнами - **Ctrl+a a** - между последним активным.

Ctrl+a <НОМЕР> - выбор окна по номеру. **Ctrl+a (p|n)** - циклическое перемещение между окнами. p - prev, n - next. Ctrl+a " - список окон для переключения.

Управление окнами - **Ctrl+a A** - изменить заголовок окна. Аналогично вводу команды title при нажатии Ctrl+a :

Ctrl+a C - очистить окно.
Ctrl+a F - подогнать размер окна под текущий размер терминала.
Ctrl+a H - протоколирование окна в файл screenlog.<НОМЕР ОКНА>
Ctrl+a K - уничтожить окно.
Ctrl+a M - режим слежения за активностью в окне. Если в момент этого вы находитесь в другом окне - в подсказке будет выведено:activity in window <НОМЕР ОКНА>
Ctrl+a r - переключение режима переноса по словам. (wrap)
Ctrl+a S - очень интересный режим работы. Сплит. То-есть текущее окно разделяется на две части и в обоих можно открыть по новому окну.
Переключение между окнами Ctrl+a; TAB, выход из режима сплит - Ctrl+a Q.
Общие команды

Ctrl+a ? - помощь

Ctrl+a Esc - режим скроллинга. Он же режим копирования. Для копирования подведите курсор к нужному месту и нажмите пробел.
Ctrl+a] - Вставка выделенной области.
Ctrl+a x - Запереть менеджер. При вкомпиленной поддержке PAM - для разблокировки нужно ввести пароль пользователя от которого запущен менеджер. В обратном случае пароль для разблокировки будет запрошен при блокировании.

НАИБОЛЕЕ ЧАСТО ПРИМЕНЯЕМЫЕ ОПЦИИ КОМАНДНОЙ СТРОКИ.

rd - подключиться к screen. Сделать deatach для остальных сессий.
list/-ls - список запущенных менеджеров.
dm - запуск screen в режиме deatach. Полезно для init скриптов или скриптов вообще.
wipe - удалить сведения о запущенных менеджерах. Полезно в случае потери менеджера, но сохранения информации о нем.
x - присоединиться к screen. Присоединение осуществляется даже в случае существующих соединений. Полезно при работе с одним screen из разных окружений. Например один screen и на X и на консоль. ;)

Пример: коннектимся к серверу, запускаем screen (набиваем команду "screen"). Запускаем на выполнение программу. Отключаемся от запущенного скрина сочетанием клавиш Ctrl+a d (detach). Снова коннектимся к серверу и подключаемся к
запущенному скрину командой screen -rd.

Источник (получено *2026-11-01 23:29*):
http://muff.kiev.ua/content/screen-diskonnekty-teper-ne-strashny