

Сканирование в Linux



Работа с периферией

ОТ А ДО Я

У вас есть сканер, и вам хотелось бы работать с ним в Linux? Вы хотите дать доступ к нему другим пользователям сети, которые используют не только операционные системы Unix-like, а, например, ОС Windows? У вас это, скорее всего, получится, но придется немного потрудиться на этапе начальной настройки сканера в Linux.

Стандартный набор SANE в теории...

Первое, что нужно сделать: забудьте, что ваш сканер совместим с TWAIN. В Linux это вам ничем не поможет, хотя инициативная группа TWAIN и имеет планы начать портирование своей разработки на Linux в конце текущего года. Стандартным средством сканирования в Linux является SANE (www.mostang.com/sane), что расшифровывается как Scanner Access Now Easy и переводится как «доступ к сканеру теперь легок». SANE представляет собой интерфейс между сканером и программой сканирования, то есть вы можете использовать разные программы для сканирования или эксплуатировать одновременно несколько сканеров через одинаковый интерфейс, что, согласитесь, является скорее плюсом, нежели минусом,

с учетом широкого перечня программ сканирования для Windows от разных производителей сканеров.

В поставку SANE входят три программы сканирования:

- ▶ `scanimage` — консольная утилита, способная пригодиться на начальном этапе настройки сканера, а также незаменимая вещь для создания различных скриптов;
- ▶ `sane-find-scanner` — очень полезная программа на этапе настройки сканера;
- ▶ `saned` — сетевой сервер сканирования: именно с его помощью и предоставляется доступ к сканеру в сети.

...и на практике

Для настройки сканера в Linux вначале нужно убедиться в том, что он поддерживается SANE. Это легко сделать, ознакомившись с со- »

» держанием статус-страницы на сайте разработчиков пакета. Отметим сразу, для SCSI-сканера шанс того, что он поддерживается в Linux, заметно выше, чем для сканера с разъемом LPT. USB-сканеры по параметру совместимости находятся между этими двумя стандартами, и возни с их настройкой много, да и результат может оказаться отрицательным, особенно если поддержка этого сканера была добавлена совсем недавно.

Убедившись, что сканер поддерживается, загружаем SANE с сайта разработчиков. Даже если ваш дистрибутив Linux имеет уже готовый пакет, все равно SANE лучше скачать и скомпилировать, поскольку он довольно динамично развивается и поддержка новых и старых сканеров в нем постоянно совершенствуется. Распаковываем загруженный архив и компилируем SANE:

```
tar xfvz sane-x.x.x.tar.gz
cd sane-x.x.x
./configure --prefix=/usr; make; make install
```

Так же поступаем и с XSane. Теперь начинается самый сложный процесс — настройка.

SCSI-сканеры

Вначале опишем процесс для SCSI-сканера. Откроем консоль и набираем:

```
sane-find-scanner
```

И получаем в результате примерно следующее:

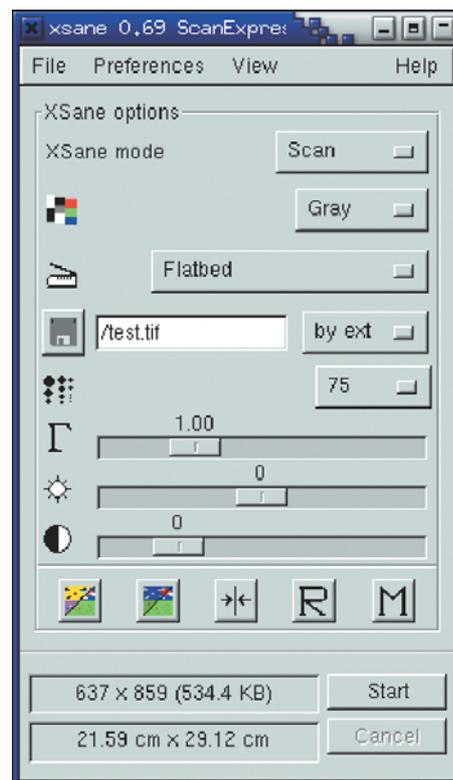
```
# Note that find-scanner will find
any scanner that is connected
# to a SCSI bus. It will even find
scanners that are not supported
# at all by SANE. It won't find
a scanner that is connected to a
# parallel, USB or other non-SCSI port.
find-scanner: found scanner «SCANNER
2.02» at device /dev/scanner
find-scanner: found scanner «SCANNER
2.02» at device /dev/sg0
find-scanner: found scanner «SCANNER
2.02» at device /dev/sga
```

Но это еще далеко не все. Утилита определяет все сканеры, подключенные к шине SCSI, даже те, которые не поддерживаются SANE, и игнорирует сканеры, подключенные к другим шинам — LPT и USB. Об этом, как

можно видеть, сообщается в начале вывода в строках, начинающихся с символа «#».

Если в выводе ваш сканер будет упомянут, то это значит, что ядро Linux как минимум его «видит» и настройку SCSI-подсистемы ядра можно пропустить. Если его там нет, то нужно настроить SCSI-адаптер. К сожалению, большинство моделей сканеров поставляются с дешевыми ISA-контроллерами, имеющими массу недостатков, начиная практической невозможностью использования этих контроллеров в Linux и заканчивая невозможностью использования прерываний в своей работе, что сводится к зависанию компьютера в процессе сканирования. Мы пытались заставить нормально работать подобную карту, поставляемую со сканерами фирмы HP, но, к сожалению, не получилось — пришлось покупать дешевый PCI-контроллер. Вероятно, сделать это придется и вам, причем вместе со SCSI-кабелем, поскольку на новых контроллерах чаще всего используется 50-пиновый разъем, в то время как поставляемый со сканером кабель обычно имеет 25 контактов.

Если же у вас есть PCI-контроллер, то нужно найти поддерживающий его модуль. Это можно сделать, обратившись к документации ядра и SCSI-HOWTO. Настроив SCSI-контроллер (то есть добившись того, чтобы ядро его «видело» и правильно определяло), необходимо также убедиться в том, что в ядре есть поддержка SCSI-сканеров и прочего SCSI-оборудования. За это в ядре отвечает модуль sg. После этих манипуляций ваш сканер должен найти программа sane-find-scanner.



▲ Главное окно программы XSane: все на месте и, кажется, нет ничего лишнего

Теперь определяем backend (драйвер вашего сканера), обратившись к сайту SANE. Давайте посмотрим конфигурационный файл этого backend. В нем обычно ничего не нужно править, но просмотреть полезно. Его синтаксис представлен ниже:

```
scsi Производитель Модель Тип
Шина Канал ID LUN
```

Причем можно использовать символ «*» в значении «любой», также можно опускать



Для справки...

SCSI в Linux

Немного полезной информации о SCSI-подсистеме Linux. Допустим, вы включили компьютер, а сканер включить забыли. Следовательно, он не определился ядром и вы не можете его использовать. Будете перезагружаться? Конечно, нет! Это вам не Windows. Следует включить сканер и от суперпользователя набрать в консоли команду:

```
echo «scsi add-single-device 0 0 5 0» ->
/proc/scsi/scsi
```

Цифры в этой команде аналогичны четырем последним параметрам для конфигурационного файла backend и означают соответственно:

Шина Канал ID LUN.

Есть и противоположная команда:

```
echo «scsi remove-single-device 0 0 5 0» ->
/proc/scsi/scsi
```

Она, наоборот, заставляет ядро «забыть» о том, что такое устройство вообще было в системе.

» параметры с конца. То есть, например, для сканера Agfa SnapScan 1236S (да и для любого сканера Agfa) строчка в конфигурационном файле `snapscan.conf` будет выглядеть следующим образом:

```
scsi AFGA * Scanner
```

Для любого сканера UMAX в `umax.conf`:

```
scsi UMAX * Scanner
```

Необходимые данные можно получить из файла `/proc/scsi/scsi`. Также поддерживается прямое указание файла устройства:

```
/dev/scanner
```

или, например:

```
/dev/sga
```

Но делать этого не рекомендуется, так как имена устройств могут меняться, и файл конфигурации тоже придется менять.

С файлом конфигурации `backend` мы разобрались. Рассмотрим еще два файла: `dll.conf` и `net.conf`. В первом содержится список всех используемых `backend`. Неиспользуемые в данный момент комментируются (во всех конфигурационных файлах SANE символом комментария является решетка — «#»). В этом файле желательно закомментировать все строки, кроме вашего `backend` и, возможно, `net`, так как их загрузка и определение того, что сканера, который ими поддерживается, в системе нет, могут занять довольно много времени. Второй файл как раз и является конфигурационным файлом для `backend net`, который предоставляет доступ другим сканерам в вашей сети. Так что, если у вас нет сети или в ней нет других сканеров, то

можете закомментировать и его — за исключением случая подключения LPT-сканера.

Теперь набираем

```
$ scanimage -L
```

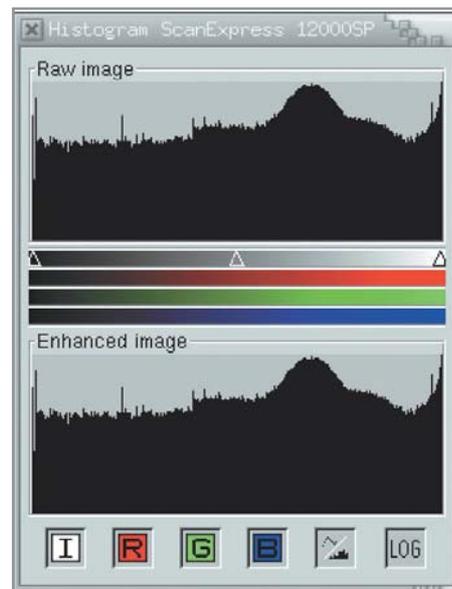
и видим примерно следующее:

```
device 'mustek:/dev/sga' is a Mustek
ScanExpress 12000SP
flatbed scanner
```

Теперь ваш сканер можно использовать. В случае со SCSI-сканерами иногда помогает информация о других типах сканеров.

LPT-сканеры

Теперь о настройке LPT-сканеров. С ними обычно все происходит не так просто, как со SCSI-сканерами. Вначале стоит проверить, установлен ли в BIOS Setup для соответствующего порта какой-нибудь расширенный протокол обмена данными (ЕСР или ЕРР). Какой именно нужен вам, можно выяснить в документации к сканеру. Следующим шагом является правка конфигурационного файла нужного `backend`. Следует внимательно прочитать `man`-страницу соответствующего `backend` и комментарии в конфигурационном файле. Всю настройку следует производить с правами суперпользователя (`root`). Связано это с тем, что большинство LPT-`backend` используют библиотеку для прямого доступа к LPT-порту, которая требует прав суперпользователя для своей работы. Более конкретная информация об этой части подключения LPT-сканеров находится в документации к каждому отдельному `backend`.



▲ Окно коррекции изображения: это не Photoshop, но с его помощью можно хорошо исправлять огрехи изображения

После того как сканер заработает с правами суперпользователя, нужно разрешить сканирование для остальных пользователей. Делается это следующим необычным образом: сканер «расшаривается» по сети, даже если у вас ее нет, а потом с применением `backend net` используется обычными пользователями.

USB-сканеры

Поддержка этих сканеров появилась совсем недавно, так же как и поддержка самой шины USB. Поэтому она сейчас активно развивается, и для успешной работы вашего сканера следует использовать текущее стабильное ядро. В общих чертах все USB-сканеры можно разделить на три группы: сканеры с внутренним SCSI-интерфейсом, сканеры с внутренним LPT-интерфейсом и сканеры с внутренним USB-интерфейсом. Последние появились относительно недавно. Понимать это следует так: первоначально имелись только SCSI-сканеры и LPT-сканеры. Когда их было нужно переделывать под шину USB, то разработчики, не долго думая, приделали к готовым сканерам чип-конвертер с нужного интерфейса на USB. Такими были большинство первых USB-сканеров, следовательно в Linux они поддерживались в соответствии с их внутренним интерфейсом: для этого в SANE был даже включен код использования USB-шины как SCSI. Однако тут есть одно но: необходимо знать, какой именно чип-конвертер применили разработчики сканера, а они этими данными делятся не особенно охотно. Последние модели USB-сканеров являются настоящими: в них уже не используется чип- »



Sane-frontends

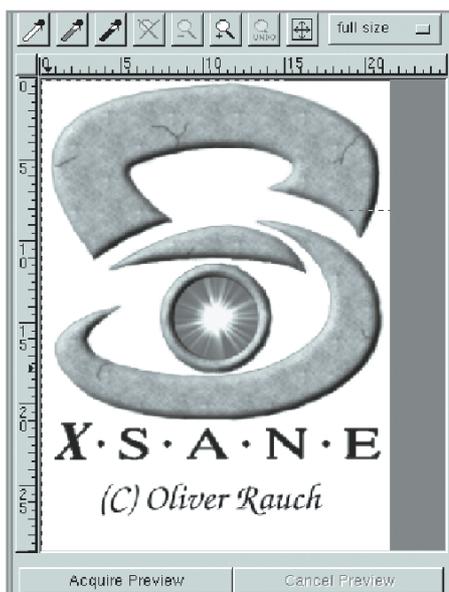
Дополнительные утилиты

В пакете `sane-frontends` находятся следующие утилиты:

- ▶ `scanadf` — утилита для использования ADF (устройства для автоматической подачи бумаги)
- ▶ `xsane` — программа для работы с цифровыми камерами, поддерживаемыми SANE
- ▶ `xscanimage` — утилита для сканирования из-под X Window

Более продвинутой является утилита XSane (www.xsane.org). Она позволяет использовать все возможности SANE, при этом имея вполне дружелюбный пользовательский ин-

терфейс. Автор выполнил перевод этой программы на русский язык и использует ее для сканирования под Linux. Теперь все шансы разобраться с этой программой есть даже у пользователей, слабо знающих английский. XSane может делать предпросмотр сканируемого изображения, изменять настройки цвета, пересылать полученную картинку по факсу или по электронной почте. Эта программа довольно динамично развивается и вполне может конкурировать с аналогами для Windows, в чем-то им проигрывая, а в чем-то опережая.



▲ Окно предварительного просмотра: обидно, что XSane не поддерживает сразу несколько областей сканирования

Клиентом является backend net, конфигурационный файл которого аналогичен файлу конфигурации saned, но в нем называются компьютеры, к которым необходимо подсоединяться, и в нем не используется символ «+».

С теорией покончено — переходим к практике. Предположим, вам нужно использовать сканер компьютера с символическим именем scan-server в демоне и предоставить доступ компьютеру с именем scan-client к вашему сканеру.

Вначале необходимо настроить демон inetd или xinetd. Для этого редактируем файл /etc/services и регистрируем в нем порт, используемый SANE (6566). Для этого прописываем в нем строчку sane 6566/tcp. Затем для демона inetd в файле /etc/inetd.conf вписываем строку:

```
sane stream tcp nowait scanner.scanner
    /usr/local/sbin/saned saned
```

А для xinetd создаем в каталоге /etc/xinetd.d/ файл с именем, например, saned следующего содержания:

```
service sane {
    port = 6566
    socket_type = stream
    wait = no
    user = scanner
    group = scanner
    server = /usr/local/sbin/saned
}
```

В приведенных выше строках scanner — пользователь и группа имеющих право на

сканирование, так как использование root не является хорошей идеей в абсолютном большинстве случаев. После правки файлов перезапускаем соответствующий демон. Теперь редактируем saned.conf и добавляем в него строку scan-client. Все — ваш сканер может использоваться компьютером scan-client после его настройки как клиента. Для доступа же к сканеру, подключенному к компьютеру с именем scan-server, необходимо снять комментарий со строчки, содержащей backend net в файле dll.conf, если она закомментирована, и добавить строчку scan-server в файл net.conf. Теперь вы можете использовать сканер, подключенный к компьютеру scan-server.

Описанный выше процесс «расшаривания» сканера пригоден только для ОС UNIX-like, поддерживаемых SANE напрямую. Для пользователей ОС Windows этот метод не подходит, так как SANE в ней работает иначе. Программа XSane была портирована на ОС Windows, но она в этой своей реализации является скорее «белой вороной». Посудите сами: стандартный интерфейс GTK+ в среде GUI Windows является несколько неуместным.

Однако я не стал бы заводить речь о сканировании в ОС Windows с использованием сканера, подключенного к компьютеру с Linux, если бы не было программы SaneTwain (www.ozuzo.net/sanetwain), являющейся чистым приложением Windows и, соответственно, лишенной всех связанных с этим недостатков XSane.

Впрочем, вы можете попытаться дождаться безвременного обещанного портирования TWAIN под Linux. ■ ■ ■ Иван Устюжанин

» конвертер, и сканер сам осуществляет соединение по шине USB.

В настоящее время поддержка USB-сканеров в Linux, к сожалению, реализована не так хорошо, как хотелось бы. По этой причине дать каких-либо общих рекомендаций мы пока не можем. В каждом конкретном случае вам, помимо использования sane-find-scanner, скорее всего, придется обращаться к документации по соответствующему backend и текущей версии ядра.

Сетевое сканирование

Предположим, вы хотите дать доступ к сканеру другим пользователям сети, причем работающим как с Linux, так и с Windows, или, наоборот, использовать чей-то сетевой сканер. В таком случае все, что вам нужно, у вас уже есть — это пакет SANE.

В SANE входит как сервер сканирования, предоставляющий доступ удаленным пользователям к сканеру, так и клиент, позволяющий использовать чей-то сетевой сканер. Сервером сканирования является saned, настройки которого находятся в файле saned.conf. Синтаксис файла довольно прост: в нем по одному на строку записываются имена компьютеров, имеющих право на сканирование. Также есть специальный символ «+», который указывает на то, что сканировать смогут все пользователи. Чаще всего этот сервер запускается через какой-либо домен — inetd или xinetd, — и права на доступ к сканеру обычно задаются через его конфигурационные файлы.



SCSI

В заключение

Стоит отметить, что если ядро «видит» сканер, но scanimage его не находит, то необходимо проверить настройки параметров для сканера в драйвере контроллера SCSI. Дело в том, что большинство сканеров используют асинхронный доступ и не поддерживают Disconnect/Connect протокола, а ядро пытается включить эти функции для сканера, что приводит к зависанию последнего. Обычно об этом написано в man-странице данного backend. О том, как изменить эти параметры, читайте в документации к драйверу вашего

SCSI-адаптера. Еще одним важным замечанием является то, что по умолчанию на файлы SCSI-устройств устанавливаются довольно жесткие права, как, например, в моем дистрибутиве — crw----- и владельцем root, — что не позволяет сканировать обычным пользователям. Я рекомендую сделать группу scanner или аналогичную, включить в нее всех пользователей, имеющих право сканировать, определить владельцем файла устройства созданную группу и задать права на него crw-rw----.