

Стек бездрайверного сканирования и печати ОС Linux

Александр Певзнер, pzz@apevzner.com

Коротко об докладчике

Использую Linux для повседневной работы со времен ядра 1.2.12

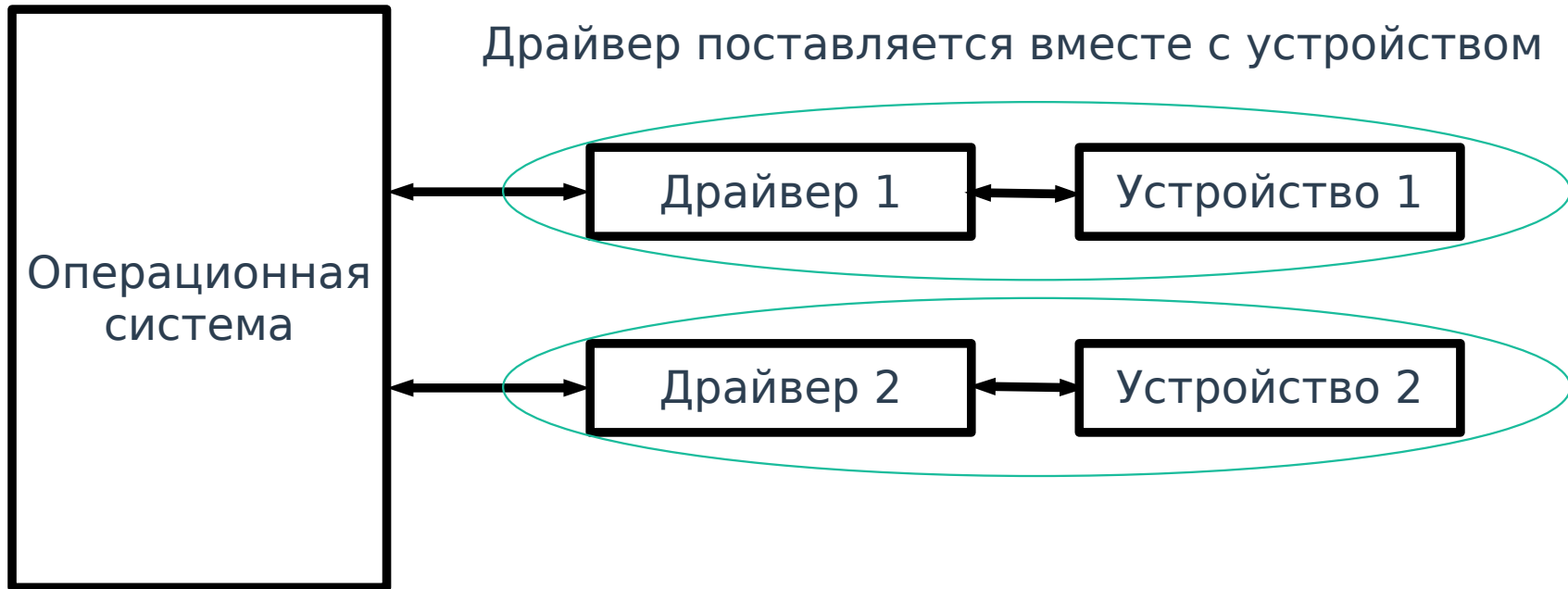
Автор двух популярных пакетов:

- <https://github.com/alexpevzner/sane-airscan> - драйвер SANE, реализующий поддержку «бездрайверных» сканеров
- <https://github.com/OpenPrinting/ipp-usb> - демон, реализующий поддержку «бездрайверного» сканирования и печати для USB устройств

Что такое «бездрайверное» сканирование и печать?

Традиционная ОС, «как в учебнике»:

Драйвер поставляется вместе с устройством



Что такое «бездрайверное» сканирование и печать?

FLASH-память - привычное «бездрайверное» устройство:

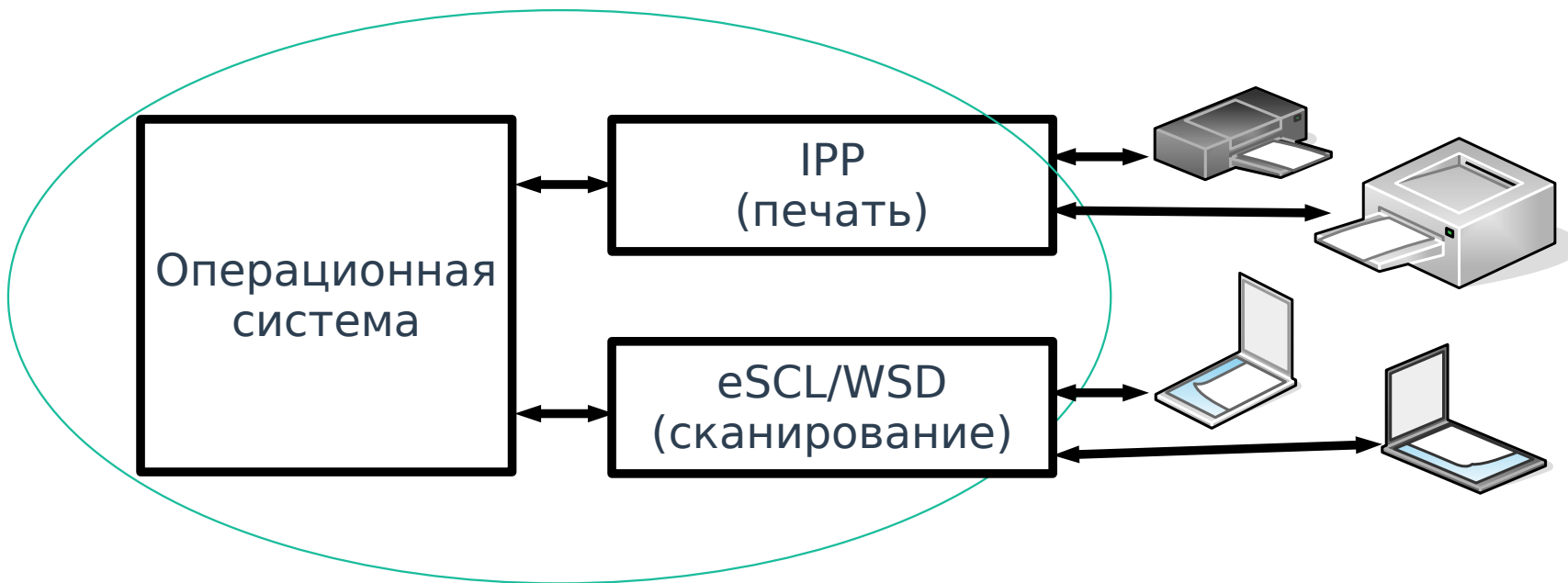
Общий драйвер стал частью ОС



Все флешки реализуют единый стандартный протокол и обслуживаются общим драйвером

Что такое «бездрайверное» сканирование и печать?

И то же самое для принтеров и сканеров:



Функциональные компоненты стека

Компоненты стека обеспечивают следующие функции:

1. Автоматическое обнаружение сетевых устройств
2. Поддержку протокола печати
3. Поддержку протокола сканирования
4. Поддержку несетевых (USB) устройств

Стандарты и протоколы

Существует два основных набора стандартов:

1) Стандарты от IETF/Apple:

- DNS-SD для поиска устройств
- IPP для печати
- eSCL для сканирования
- Этот набор стандартов известен как IPP everywhere (в части печати), Bonjour Printing/Airprint (начиная с 1.4 включает сканирование), Mopria printing/scanning

2) Стандарты от Microsoft (Web Services For Devices):

- WS-Discovery для поиска устройств
- WS-Print/WS-Scan для печати и сканирования
- Проведен, как набор стандартов Web Consortium (w3c) и немного отличается от спецификации от Microsoft

Актуальность стандартов

- Практически все бездрайверные принтеры поддерживают протоколы IETF/Apple (DNS-SD, IPP). Поэтому поддержка WSD для печати не имеет особого смысла
- Сканеры активно используются оба набора протоколов (DNS-SD+eSCL, WS-Discovery+WS-Scan), и разные устройства поддерживают разные варианты. Поэтому в драйвере приходится поддерживать оба стандарта

Поддержка «бездрайверных» протоколов устройствами

- Практически все сетевые принтеры и многие с USB подключением поддерживают бездрайверную печать
- Поддержка сканерами бездрайверного сканирования пока не настолько полная, но имеет тенденцию к росту
- Часть устройств реализуют полную функциональность (высокие разрешения изображений, расширенное управление сканером и т.п.) только через проприетарные протоколы
- В перспективе нескольких лет, вероятно, всё сканирование и печать будут бездрайверными

Основные компоненты стека в ОС Linux

- **Avahi** — реализует DNS-SD (автоматическое обнаружение сетевых устройств)
- **CUPS** — реализует протокол печати IPP
- **sane-airscan** — протоколы сканирования eSCL и WS-Scan, обнаружение WSD устройств с помощью протокола WS-Discovery
- **ipp-usb** — поддержка USB устройств

sane-airscan — драйвер сканера с поддержкой двух протоколов (eSCL/WSD)

- **Написан на C (около 20К строк кода)**
- **Два протокола в одном драйвере**
- **Более 170 поддерживаемых устройств (оттестированных пользователями). Примерно 1/3 из них - WSD-only**
- **Входит во все дистрибутивы Linux, в *BSD, в ChromeOS**

sane-airscan — технические особенности

- Одно физическое устройство может быть обнаружено в сети в нескольких ипостасях: IPv4/IPv6, HTTP/HTTPS, eSCL/WSD и т.п. Сведение их вместе, так, чтобы показывать пользователю именно физические устройства — достаточно непростая задача, занимает около 1/4 кодовой базы драйвера
- SANE позволяет запустить сканирование в фоновом режиме, но требует чтобы драйвер мог вернуть фактические параметры изображения сразу после запуска сканирования. Однако драйверу они доступны только после получения изображения. Приходится возвращать не фактические, а запрошенные параметры, а потом приводить изображение в соответствие им. Это позволяет корректно реализовать функцию Cancel

Поддержка USB — техническая сторона

- До сих пор мы говорили больше о сетевых устройствах. Однако есть устройства без сети (с USB-подключением)
- Для них разработан стандарт IPP over USB (Class 7, subclass 1, protocol 4)
- Его вернее было бы назвать HTTP over USB, и он поддерживает все основанные на HTTP протоколы: IPP, eSCL и даже веб-консоль
- Суть протокола в том, что HTTP-обмен производится прямо через USB endpoint

iprusbxd — ранняя реализация

- **Первая попытка реализовать IPP over USB в Linux называлась iprusbxd. Подход был очень простой: программа принимала TCP-соединения и пробрасывала их в USB**
- **В отличие от TCP, в USB нет понятия «закрытия соединения». Поэтому если клиент обрывал HTTP-транзакцию «на полуслове», в буферах USB застревала или часть запроса или часть ответа, и терялась синхронизация между устройством и хостом**
- **Поэтому работало это не очень надежно. Печать и сканирование иногда работали, веб-консоль не работала совсем**

`ipr_usb` — современная реализация

- В отличии от `iprusbxd`, `ipr-usb` «понимает» протокол HTTP и является полноценным HTTP-проxy
- Если устройство видит начало HTTP-транзакции, транзакция всегда корректно доводится до конца, при любом поведении HTTP-клиента
- Написан на Go
- Входит во все дистрибутивы Linux. Недавно появился порт на FreeBSD. В ChromeOS не входит из-за большого размера исполняемого файла, но послужил основой их собственной реализации

С какими проблемами приходилось сталкиваться при разработке

- Отладка по переписке — необходимо очень тщательно структурировать логи
- Во время работы над sane-airscan, спецификация протокола eSCL еще не была опубликована (приходилось полагаться на reverse engineering)
- Много ошибок в прошивках устройств. Вплоть до того, что немного слишком «вольнo» отформатированный XML-запрос (с незначащими пробелами) может перезагрузить устройство

Сканирование: статус и перспективы

- **sane-airscan** добавил в Linux/BSD поддержку целого класса устройств
- Несовместимые (не поддерживающие eSCL/WSD) устройства пока встречаются, но новые устройства все чаще и чаще поддерживают бездрайверные протоколы
- Пока встречаются устройства, которые реализуют более полную функциональность через проприетарные протоколы, чем через eSCL/WSD. Например, более высокие разрешения, управление яркостью и т.п.
- В Windows 11 появился драйвер eSCL-сканера. Поддержка WSD в перспективе, вероятно, умрет
- В перспективе нескольких лет, вероятно, все устройства будут бездрайверными

IPP over USB: проблемы, статус и перспективы

- Специфическая для IPP over USB проблема: `ipp-usb` блокирует доступ к устройству проприетарных драйверов. Приходится выбирать, или `ipp-usb`, или они
- Теоретически, `ipp-usb` мог бы занимать USB-интерфейсы только когда он их на самом деле использует. Но существуют устройства, которые нестабильно работают в «смешанном» режиме, и долю их трудно оценить. Поэтому «смешанный» режим сознательно не реализован
- Из-за этого `ipp-usb` иногда «ломает» доступ к устройству, которое без него работало (с проприетарными драйверами)
- Хорошего решения пока нет
- Но в целом, существует консенсус, что польза от `ipp-usb` превышает вред
- Как и со сканерами, новые устройства все лучше и лучше поддерживают этот протокол

Как попасть во все дистрибутивы Linux

- **Востребованный проект, решающий существующую проблему пользователей**
- **Наличие бинарных пакетов под популярные дистрибутивы (я использовал OpenSUSE build services для их сборки)**
- **Участие в форумах пользователей и разработчиков**
- **Активная поддержка пользователей с проблемами**
- **После того, как попадете в Fedora/Debian/Ubuntu, дальше «пойдет само»**
- **Это занимает примерно 1 год**

Спасибо за внимание

Здесь доклад более формальный. Я так же описал эту историю, как личный опыт в статье на Хабре:

<https://habr.com/ru/articles/751214/>



Спасибо за внимание, готов ответить на вопросы