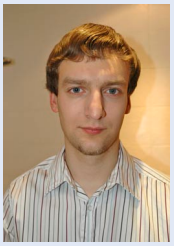
Колонка главного редактора



Тема, поднятая в редакторской колонке предыдущего выпуска, то есть непосредственно сама эта рубрика, насколько ее вообще можно таковой назвать, породила интересную совокупность отзывов и мыслей. Оказалось, что некоторые, несмотря на мои постоянные изыски сделать что-то сколь-нибудь оригинальное, находят результат банальным и стандартным, а другие, наоборот, не понимают, к чему попытки придумать нечто непонятное, и хотят увидеть привычное: краткий обзор содержимого выпуска. Синтезируя все эти данные, я пришел к в некотором роде парадоксальному выводу...

В ближайших номерах «Open Source» данная колонка будет слабо коррелировать с содержимым конкретного выпуска приложения. Представляемый в ней материал станет прямым отображением свободных мыслей на тему тенденций, событий и даже просто «вещей» из области информационных технологий. Разумеется, по возможности с акцентом в сторону программного обеспечения, но ограничиваться этим ни в коем случае нельзя.

И напоследок краткое обращение к потенциальным авторам. На странице http://osa.samag.ru/todo не только постоянно обновляются примеры тем для статей, но теперь представлены и возможные новые рубрики для «Open Source». Проявляйте активность – пишите!

Главный редактор
Дмитрий Шурупов
(osa@samag.ru)

«Open Source»
электронное приложение к журналу «Системный администратор»
№27, 11 июня 2008 г.

РЕДАКЦИЯ
Исполнительный директор
Владимир Положевец
Главный редактор
Дмитрий Шурупов
Верстка и оформление
Владимир Лукин
Сайт электронного приложения:
http://osa.samag.ru

Новости мира Open Source

Качество Open Source-кода выросло на 16%

Компания Coverity, специализирующаяся на анализе исходного кода программных продуктов, опубликовала результаты своего очередного исследования, посвященного Open Source-проектам, – Open Source Report 2008. В отчете представлены результаты анализа качества исходного кода, написанного на языках программирования C, C++ и Java. Данные отчеты спонсируются Министерством национальной безопасности США (Department of Homeland Security), поддерживаются Стэнфордским университетом и являются частью американского государственного проекта Open Source Hardening Project. Для отчета были использованы данные, которые собирались более двух лет в рамках инициативы Coverity Scan.

Статистика по проанализированному коду такова: всего было просмотрено 55 миллионов строк кода из 250 Open Source-проектов, проведено 14238 анализов, что породило 10 миллиардов строк проанализированного кода. Итоги исследования показывают общий рост качества кода Open Source-проектов на 16 процентов.

Появился альтернативный DNS-сервер с открытым кодом

Стало известно о публичном релизе нового DNS-сервера с открытым кодом – Unbound 1.0. Unbound, созданный усилиями NLnet Labs, VeriSign, Nominet и Kirei, является рекурсивным, кэширующим DNS-сервером, соответствующим стандартам. Он позиционируется как высокопроизводительная альтернатива наиболее распространенному решению в этой области – BIND. Unbound – это единственная (кроме BIND) Open Source-реализация DNS-сервера, поддерживающая стандарт DNSSEC. Поддержкой DNS-сервера Unbound будет заниматься некоммерческая голландская организация NLnet Labs.

«Мы выпустили программное обеспечение под лицензией BSD, которая позволяет использовать его в других продуктах без значительных ограничений, – заявил Олаф Колкман (Olaf Kolkman), директор NLnet Labs. – Надеемся, что публикация нашего ПО в свободный доступ будет способствовать использованию DNSSEC».

Обнародована статистика использования устаревшего Linux-ядра 2.4

Уилли Тэрро (Willy Tarreau), занимающийся поддержкой ветки 2.4 ядра Linux, провел в почтовой рассылке опрос, посвященный тому, как сейчас используется Linux 2.4. Собрав информацию от пользователей, он подготовил небольшой статистический отчет на основе этих данных.

Около половины пользователей Linux-ядра 2.4 применяют его на серверах общего назначения и регулярно обновляют. Основные причины такого использования – слабый интерес к новым возможностям (появившимся в 2.6) и недостаток времени, а в некоторых случаях – неудачная попытка перехода на 2.6 на ранних этапах. 20% пользователей работают с Linux 2.4 на серверах со специфичными приложениями, где важнейшим фактором является надежность.

Еще у 10% пользователей Linux-ядро 2.4 запущено на старых роутерах, брандмауэрах, системах обнаружения атак (IDS), которые функционируют по несколько лет. Другие 10% пользователей применяют устаревшее Linux-ядро во встраиваемых системах, где определяющим фактором является стабильность, а используемая сборка может быть достаточно специфичной.

Около 5% применений Linux 2.4 приходится на старые лаптопы, КПК и тонкие клиенты, которые уже не нуждаются в обновлениях. И последние 5% – на десктопы и, например, станции мониторинга, которые не обновляются, потому что «просто работают».

Проект FreeBSD перешел на Subversion взамен CVS

Ivan Voras со ссылкой cvs commit от 1 июня объявил о том, что проект разработки свободной операционной системы FreeBSD перешел на систему управления версиями Subversion вместо устаревшей CVS.

CVS-хранилище исходного кода FreeBSD – одно из старейших и крупнейших: его история насчитывает около 12 лет и около 180 тысяч commit (что в среднем составляет более 41 в день). Ветвь последнего релиза, RELENG_7, содержит более 42 тысяч файлов (482 Мб).

Тема перехода с CVS на Subversion (SVN) активно обсуждалась на DevSummit, проходившем в рамках конференции BSDCan 2008.

И некоторые существенные недостатки CVS (например, невозможность переименовывать/перемещать файлы, плохая работа с ветвями при непрекращающемся процессе разработки и добавлениях в дерево, неатомарные коммиты) окончательно убедили разработчи-

ков в том, что от этой системы пора отказываться.

Mozilla мигрирует на Mercurial вместо CVS

Новостной сайт Mozilla объявил о том, что миграция с системы управления версиями CVS на Mercurial подошла к завершающей стадии.

Теперь разработчики могут вносить свои изменения в исходный код в mozilla-central – новое дерево Mozilla, управляемое Mercurial. Впрочем, рекомендуется, чтобы в течение первых нескольких дней разработчики перед внесением изменений в mozilla-central получали одобрение у управляющих во избежание непредвиденных проблем. Сообщается, что новый репозитарий mozilla-central на базе Mercurial предназначен только для кода Firefox, XULRunner, Gecko. Остальные проекты Mozilla (Thunderbird, Calendar, SeaMonkey...) вольны самостоятельно решать, какую систему контроля версий они будут использовать, и если они тоже выберут Mercurial, то для них будут созданы новые репозитории.

Австрийская Вена вынуждена использовать Vista наряду с GNU/Linux

На этой неделе городской совет Вены, столицы Австрии, принял решение перенастроить 720 компьютеров с GNU/Linux, используемых в городских центрах дневной медицинской помощи для детей, для возможности запуска там альтернативной операционной системы – Microsoft Windows Vista.

Как сообщают австрийский и немецкий новостные сайты, это решение было вызвано необходимостью запускать специализированное приложение для обязательных тестов по языковым навыкам, которое работает только с веб-браузером от Microsoft.

На инициативу будет потрачено около 105 тысяч евро. Это вызвало обеспокоенность у Марии Ринглер (Marie Ringler) из Партии зеленых Вены: «Даже части этих средств было бы достаточно, чтобы ускорить адаптацию этого тестирующего ПО для Firefox компанией-разработчиком».

По мнению Ринглер, этот случай лишь доказывает, что Вена отходит от своих инициатив по использованию Open Source. В качестве другого подтверждения приводится февральское решение городского совета Вены о покупке новых лицензий от Microsoft на 7,6 миллиона евро. Впрочем, другой член городского совета, представляющий партию социал-демократов (Social Democratic Party), считает, что эти обвинения необоснованны: «Вена использовала программное обеспечение с открытым кодом на протяжении 20 последних лет и будет продолжать это делать».

Acer делает серьезную ставку на GNU/Linux

Тайваньский компьютерный производитель Acer объявил о намерении серьезно использовать в своих продуктах Linux и программное обеспечение с открытым кодом (Open Source).

Acer уже продвигает Linux для своих дешевых ультрапортативных компьютеров (netbook), первый из которых был представлен в начале июня, а теперь руководство объявило о том, что будет устанавливать Linux и на лаптопы.

Как пояснил Джанпьеро Морбелло (Gianpiero Morbello), вице-президент по маркетингу и бренду Acer, намерение активнее продвигать и использовать Linux вызвано самой корпорацией Microsoft: «Мы смещаемся в сторону Linux из-за Microsoft. Microsoft очень сильна, и это будет трудно, но мы будем усердно работать над развитием Linux-рынка».

В Acer видят два основных преимущества Linux: функционирование и цена. Предустановленный Linux-дистрибутив будет запускаться за 15 секунд по сравнению с минутами, которые уходят на старт Windows, а время работы устройства от батареи увеличится с 5 до 7 часов. А разница в цене, достигаемая путем предварительной установки на компьютеры Linux вместо Windows, позволит Acer сделать более выгодные предложения на рынке недорогих устройств.

Финансовую выгоду Linux пояснил Дэвид Драммонд (David Drummond), управляющий директор в британском отделении Acer: «Операционная система от Microsoft обычно стоит порядка 50 фунтов стерлингов за штуку. Для ПК стоимостью в 1000 фунтов это несущественно, но для компьютера ценой в 200 фунтов это очень важный фактор».

Дмитрий Шурупов,
по материалам www.nixp.ru
(osa@samag.ru)

Привет, Fedora 9! Обзор дистрибутива

Выпуск девятого релиза популярного Linux-дистрибутива Fedora (http://www.fedoraproject.org) планировался еще на конец апреля, но из-за задержек с локализацией дата выхода была перенесена. И вот 13 мая на всех зеркалах и BitTorrent-сетях появилась долгожданная Fedora 9 под кодовым названием Sulphur, что в переводе на русский язык означает «сера».

Что нам пообещали разработчики?

Каждый новый релиз Fedora всегда радует свежими наработками в сфере свободного программного обеспечения, и Sulphur здесь впереди планеты всей. Среди ключевых заявленных новшеств в дистрибутиве:

- ☑ **Новая версия ядра системы** – Linux 2.6.25.
- ☑ **Система инициализации Upstart пришла на смену классической SysVinit.** Ее основное отличие от SysVinit заключается в том, что она может вычислять зависимости между сервисами и, основываясь на этих данных, запускать некоторые сервисы параллельно, что положительно сказывается на времени загрузки системы. Подробнее можно прочесть на официальном сайте (http://upstart.ubuntu.com).
- ☑ **PackageKit** – независимое от дистрибутива решение для управления пакетами, у которого имеется готовый бэкенд для консольного пакетного менеджера yum. Теперь PackageKit является основной системой управления программным обеспечением в Fedora 9. Подробнее о нем можно прочитать, например, в моем блоге (http://eveel.blogspot.com/2008/06/packagekit.html).
- ☑ **GNOME 2.22** – последняя версия популярной графической среды, содержащая множество исправлений и улучшений. В частности, подверглась изменениям ее архитектура: теперь для доступа к ресурсам используются GIO и GVFS (вместо старой GNOME VFS).
- ☑ **KDE 4.0.2** – наконец-то в Fedora включена новая ветка KDE. KDE 4 разработана на основе библиотеки Qt 4 и радует многочисленными нововведениями. (**Прим. ред.:** Подробный обзор KDE 4.0 можно найти в «Open Source» 021.)

- **Улучшенный NetworkManager** – утилита для настройки и управления сетевыми подключениями – получила расширение функциональности в виде лучшей интеграции с мобильными устройствами.
- **Firefox 3** – в поставку Sulphur входит Firefox 3.0 beta 5, последняя версия на момент выхода дистрибутива. Для «беты» он работает весьма неплохо.
- **Наконец-то поддержка технологии Flash доступна «из коробки»** благодаря подключению бэкенда swfdec к фреймворку GStreamer.
- **TeXLive** – замена старому, неподдерживаемому пакету TeX, содержит как новые стили оформления, так и исправления многих ошибок.
- **Perl 5.10** – первый существенный релиз интерпретатора языка Perl за последние несколько лет.
- **Множество улучшений в установщике Anaconda**, который теперь умеет менять размеры разделов ext2, ext3 и NTFS, определять устройства средствами HAL и udev, а самое главное – появилась возможность сетевой установки системы.

Полный список изменений можно найти в документе «Release Notes» (http://docs.fedoraproject.org/release-notes/f9).

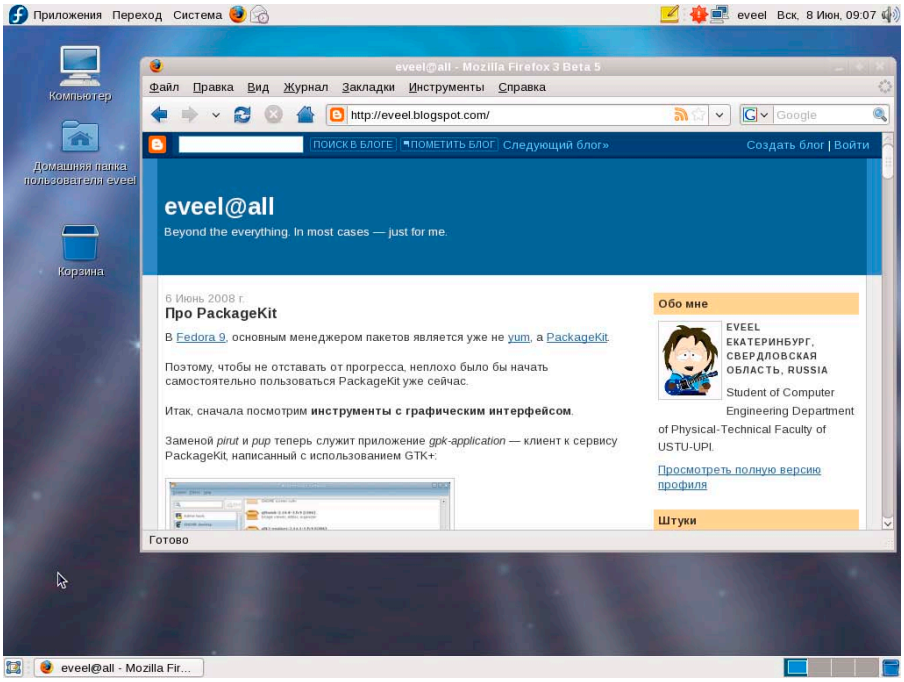
И что мы получили?

При обновлении системы с Fedora 8 на Fedora 9 меня порадовал тот факт, что пришлось только вставить установочный диск и сделать пару щелчков мыши. Остальное установщик Anaconda сделал сам, без моего вмешательства.

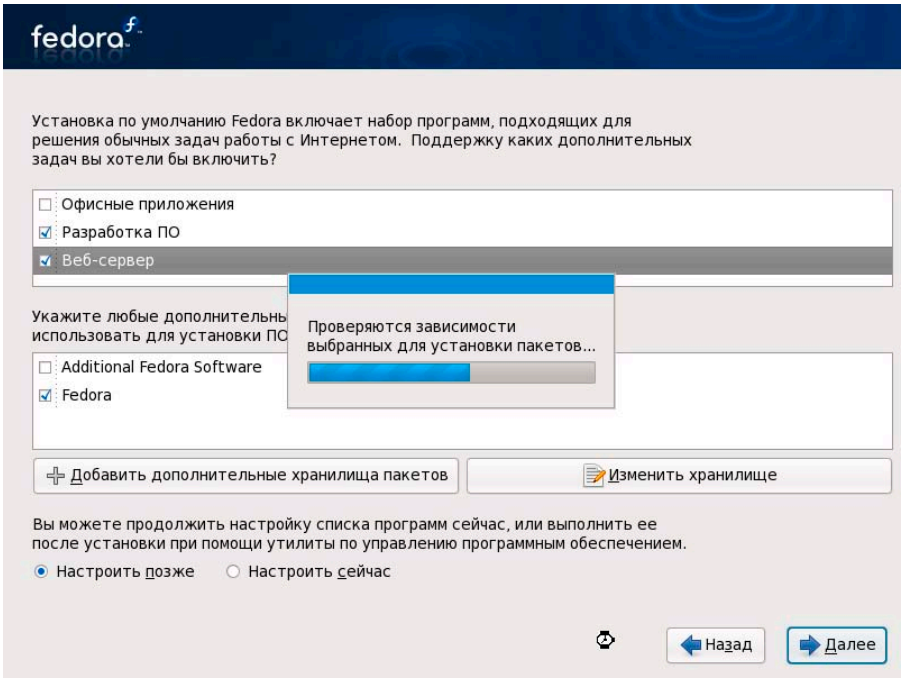
Однако как корабль назовешь, так он и поплывет: по-хорошему «допиливать» дистрибутив службе Quality Assurance следовало еще один-два месяца. И сейчас я все объясню.

Во-первых, как выяснилось, хваленый Upstart работает не как самостоятельная система инициализации, а просто как надстройка к старым скриптам загрузки. Ни о каком распараллеливании речи быть не может, и это меня огорчило. Но еще больше расстроило то, что при завершении работы системы не отображается информация о ходе процесса.

Во-вторых, на момент выхода дистрибутива сервер Xorg 1.5 находился в состоянии предварительного релиза, поэтому проприетарные драйверы от NVIDIA и ATI не работают. К счастью, 28 мая NVIDIA выпустила новую версию драйверов, уже совместимых с Xorg 1.5, поэтому сейчас с ними проблем нет.



Внешний вид Fedora 9 после установки



Установщик Anaconda строит дерево зависимостей пакетов

В-третьих, по ряду непонятных причин, приложения, обращающиеся напрямую к ALSA, выдают «потрескивающие» басы, что неприятно сказывается на звуке (например, драм-машина Hydrogen).

Следующая проблема обнаружена в том, что Firefox 3.0 beta 5 просто отказался запускаться: что-то сломалось при обновлении с Fedora 8. Впрочем, в новых инсталляциях Fedora браузер Firefox работает нормально. Заявленная поддержка Adobe Flash не так уж хороша. Следует помнить, что swfdec – это свободный Flash-плеер, имеющий определенные проблемы с поддержкой этого формата.

PackageKit, несмотря на запрет проверки обновлений, все равно постоянно желает обновиться и из-за этого постоянно блокирует процесс yum.

Еще расстроила неполная, а временами и вовсе отсутствующая локализация некоторых приложений. Например, gpk-application, графический интерфейс к вышеупомянутому PackageKit, совсем не переведен. Отдельным пунктом в проблемах локализации стоит отметить крайне неприятный инцидент: если вы выбрали установку на русском языке и указали на необходимость уточнить список устанавливаемых пакетов, то при попытке выбрать некоторые группы пакетов программа установки «вылетит». Проблема решается либо выполнением установки на английском языке, либо посредством

обновления установщика (см. https://fedoraproject.org/wiki/Ru_RU/Releases/9/InstallationFailed).

Даже если допустить, что половина проблем проявилась при обновлении, все равно остается неприятный осадок. Тем не менее эти неполадки не настолько критичны, чтобы пользоваться системой было неприятно.

Учитывая стремление разработчиков Fedora Project сделать использование дистрибутива законным во всем мире, из стандартной поставки традиционно исключена поддержка MP3 и DivX, но это совсем не страшно: проблема решается подключением репозитариев livna (http://rpm.livna.org), tigro (http://www.tigro.info), adobe (http://linuxdownload.adobe.com/linux/i386) и установкой соответствующих пакетов.

Приговор

Достоинства нового выпуска перекрывают недостатки, большая часть которых уже исправлена: видеодрайверы NVIDIA уже работают, а ATI скоро подоспеет, проблема с инсталляцией решена, а вместо swfdec можно установить официальный Flash Player от Adobe.

Сама система работает весьма быстро, а замечаний к ее стабильности у меня нет. Желаю вам успехов при работе с Fedora 9!

Дмитрий Усталов
(eveeel@gmail.com)

FOSS Review 003

Сообщество Open Source своевременно реагирует на все изменения. Вот и у менеджера буфера обмена Glipper, который вдруг обзавелся зависимостями от сопровождающих компонентов среды

Lybniz

- ☑ **Версия:** 1.3.2.
- ☑ **Лицензия:** BSD.
- ☑ **Размер:** 70 Кб (tar.gz).
- ☑ **Сайт:** http://lybniz2.sourceforge.net.

Маленькая, но очень полезная каждому студенту программка. Lybniz строит графики различных математических функций, совмещая в одной рабочей области до трех разных графиков. Последние будут выделены разными цветами, также доступны параметры координатной сетки и увеличение. А еще программа по заданному числовому значению аргумента сумеет посчитать значение функции. Но радость была бы неполной без возможности экспорта графика в формат PNG. Lybniz написана на языке Python, а интерфейс программы – на PyGtk.

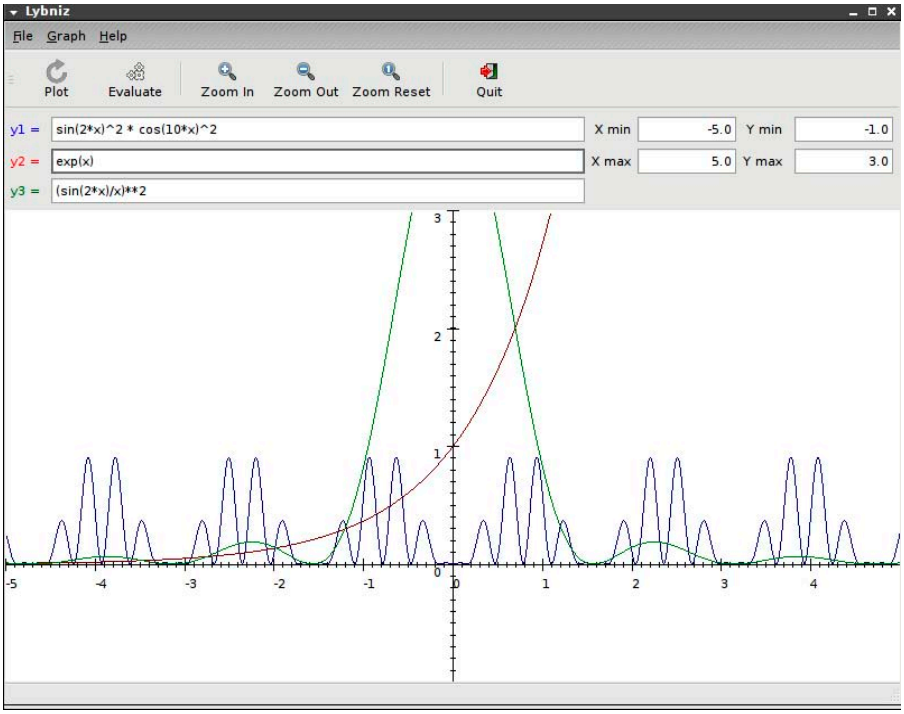
Incollector

- ☑ **Версия:** 1.1.
- ☑ **Лицензия:** GPL.
- ☑ **Размер:** 300 Кб (tar.gz).
- ☑ **Сайт:** http://incollector.devnull.pl.

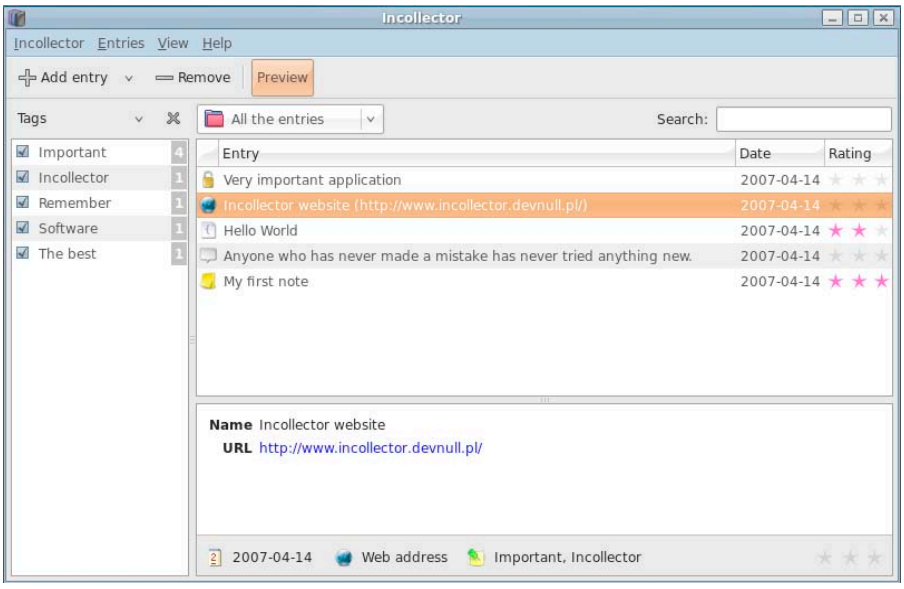
Мы часто даже не задумываемся, как много мелкой информации нам приходится хранить в персональных компьютерах. Incollector – программа, предназначенная для хранения и организации различной небольшой информации, будь то заметки, цитаты, онлайн-беседы, слова, ссылки, исходный код программ или серийные номера. Данные можно сортировать по каталогам, присваивать им метки и оценку. У Incollector красивый интерфейс на базе Gtk# (Mono), программа умеет сворачиваться в системный трей, причем поиск и добавление записей может осуществляться прямо оттуда.

Parcellite

- ☑ **Версия:** 0.7.
- ☑ **Лицензия:** GPL.
- ☑ **Размер:** 113 Кб (tar.gz).
- ☑ **Сайт:** http://code.google.com/p/xyhthyx.



Lybniz



Incollector

GNOME, появился хороший заместитель – Parcellite. Такого рода программы позволяют обращаться к предыдущим вставкам в буфер обмена, а не только к последней. Программа удобная в обращении, быстрая и легкая, а главное – совсем не зависит от GNOME.

GNOME Commander

- ☑ **Версия:** 1.2.6.
- ☑ **Лицензия:** GPL.
- ☑ **Размер:** 2.2 Мб (tar.bz2).
- ☑ **Сайт:** http://www.nongnu.org/gcmd.

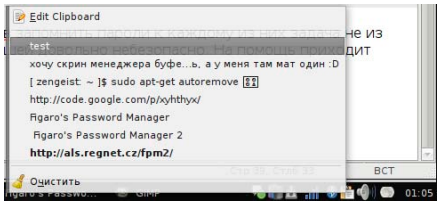
Двухпанельные файловые менеджеры (так называемые ортодоксальные) очень удобны для работы с файлами. У пользователей Windows есть Total Commander, у KDE – Krusader. Но как быть с GNOME? Для этой среды есть Gnome Commander – двухпанельный файловый менеджер, реализующий весь необходимый набор операций над файлами, включая работу с архивами, поиск, массовое переименование, отправку файлов (E-mail, IM, Bluetooth), удаленные соединения (FTP и Samba). Программе может быть оснащена и другими возможностями с помощью дополнительно подгружаемых модулей. Интерфейс GNOME Commander поддается тонкой настройке и интуитивно понятен пользователю. Нельзя не отметить хорошую интеграцию непосредственно с библиотеками и приложениями GNOME.

Figaro's Password Manager 2

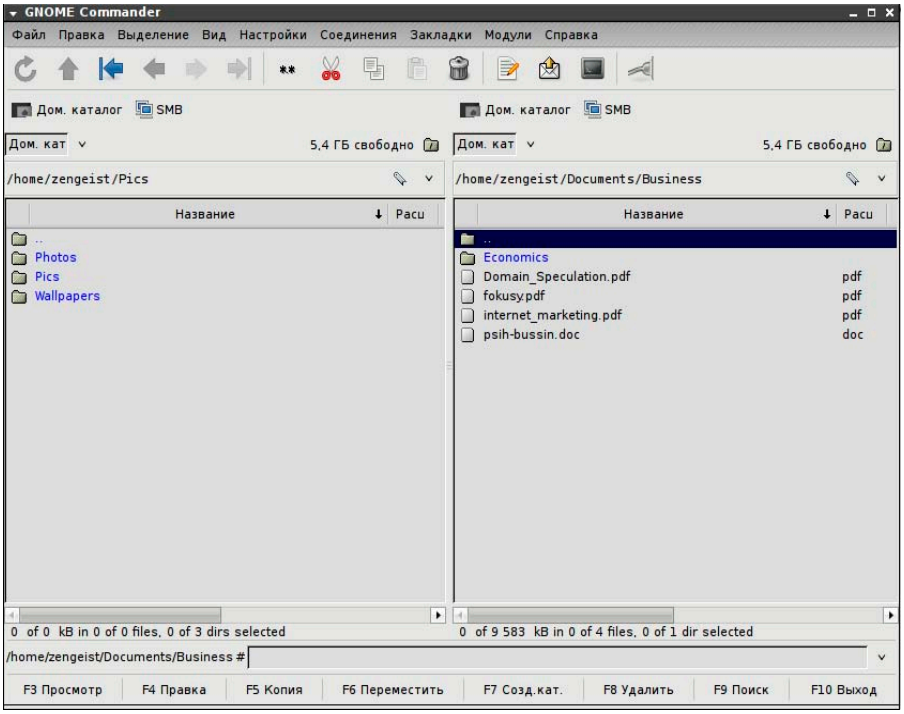
- ☑ **Версия:** 0.71.
- ☑ **Лицензия:** GPL.
- ☑ **Размер:** 184 Кб (tar.bz2).
- ☑ **Сайт:** http://als.regnet.cz/fpm2.

Во всем многообразии электронных и онлайн-сервисов запомнить пароли к каждому из них – задача не из легких, а иметь один пароль ко всему – по меньшей мере довольно небезопасно. На помощь приходит специальная программа – Figaro's Password Manager 2 (FPM2). Это легкий менеджер паролей, написанный на Gtk2. Для безопасного шифрования паролей используется алгоритм blowfish. Записи можно разделять по категориям, а имя пользователя и пароль вставлять в буфер обмена. Помимо этого FPM2 может автоматически запускать программы, где будет вводиться пароль (например, Firefox или консоль). Присутствует и неплохой генератор паролей.

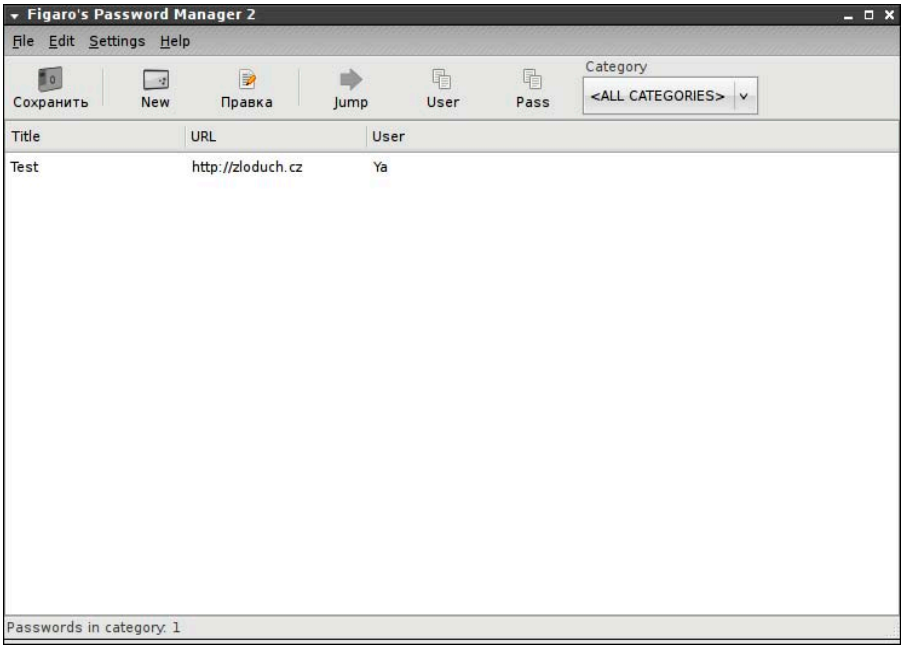
Роман Комков (r.komkov@gmail.com)



Parcellite



GNOME Commander



Figaro's Password Manager 2

Обзор грядущего релиза Firefox 3

Вот и подходит к концу полуторагодовая работа над новой, третьей по счету, версией межплатформенного веб-браузера Mozilla Firefox. Это второй по популярности браузер в мире, и его доля постепенно растет.

Ожидать релиз стоит к концу этого месяца – в так называемый День закачки (Download Day), который пока не объявлен (http://www.spreadfirefox.com/ru/worldrecord). Он примечателен тем, что пользователи со всего мира будут пы-

таться поставить рекорд по количеству закачек программы в сутки.

Итак, новая версия браузера от Mozilla основывается на HTML-движке Gecko версии 1.9 (http://wiki.mozilla.org/Gecko_1.9_Roadmap), который, по словам разработчиков, содержит более 14 тысяч исправлений, включая изменения, направленные на увеличение производительности и стабильности, поправки в обработке кода. Соответственно, и браузер должен стать быстрее, стабильнее, надежнее. Разработчики также уверяют, что находящееся «под капотом» очень многое может предложить создателям веб-сайтов и разработчикам дополнений для Firefox.

Что же нового для пользователя предлагает Mozilla на этот раз?

Проблемы безопасности

Среди улучшений, направленных на повышение безопасности, стоит выделить встроенные средства борьбы с различными вредоносными программами. Firefox предупреждает об опасности, когда вы попадаете на сайт из «черного списка», который устанавливает вирусы, шпионское ПО, трояны и др. Помимо этого, браузер передает антивирусу информацию о том, что скачивается исполняемый файл (впрочем, ни ClamAV в Kubuntu, ни avast! в Windows XP не проявили дополнительной активности). Этими мерами снижается вероятность того, что файл, который был скачан, окажется опасным для ОС.

Что же касается кроссплатформенных улучшений в безопасности, то были исправлены ошибки, возникающие при удалении пользовательских данных. Помимо этого, теперь закладки, история активности, cookies и настройки хранятся в защищенном формате, который должен спасти от кражи даже в случае краха системы. К сожалению, разъяснений по поводу «защищенного формата» Mozilla не дала, а при использовании профиля в разных версиях Firefox ни в Windows XP, ни в Kubuntu проблем не возникло. Изменилась система запоминания паролей: теперь вы можете сохранять их после того, как успешно пройдет авторизация и начнет загружаться нужная страница.

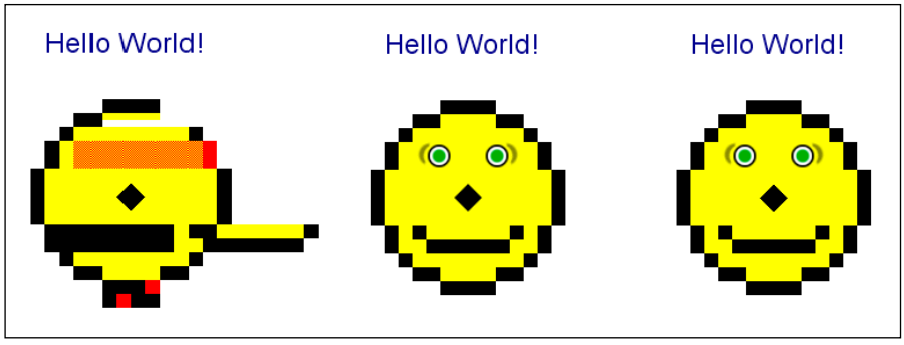
Интерфейс

Тема оформления третьей версии использует иконки GTK+, что сделало Firefox «роднее» для пользователей графической среды GNOME. А вот Windows-версия стала более похожей на Internet Explorer 7. Кнопки в веб-формах и выпадающие меню стали более сглаженными, менее угловатыми. С отрисовкой дела тоже улучшились: огрехов в Windows не замечено, но в Kubuntu есть проблемы, которые я отнес на счет версии. Так, например, кнопка с выпадающим списком всех открытых вкладок не отрисовывалась, пока не наведешь на нее курсор. (**Прим. ред.:** В Firefox 3 RC1 для Ubuntu подобных проблем замечено не было.)

Адресная строка в Firefox 3 представляет особый интерес. Иконка с левой стороны обзавелась функционалом: после нажатия на нее всплывает окно с информацией о SSL-сертификате. А с помощью «звездочки» в правой части адресной строки можно легко добавить страницу в закладки. Повторное нажатие откроет окно свойств вновь созданной закладки с возможностью редактирования. Что касается ввода адреса, то и тут есть изменения: ввод стал более интерактивным, браузер запоминает не только адреса, но и заголовки посещенных сайтов, позволяя осуществлять по ним быстрый поиск при вводе URL.

Другое новшество – полное масштабирование страницы: если раньше изменялся только размер шрифтов, то теперь увеличиваются/уменьшаются и все остальные элементы сайта (в том числе и изображения).

Изменился и менеджер закачек: улучшилось управление загрузками – можно посмотреть ссылку закачиваемого файла и восстановить закачку после разрыва или завершения работы браузера. Появился поиск по истории, в базе которого хранятся и URL скачанных файлов. Внешний вид стал лаконичнее и удобнее, в нем появились кнопки паузы и остановки закачки, присущие менеджерам загрузок. Еще одной приятной особенностью стало появление в правой части строки состояния статуса закачек. И последнее: заработали «горячие» клавиши при включенной русской раскладке в Kubuntu.



Результаты теста Acid 2: слева – Firefox 2; посередине – Firefox 3, справа – эталон

Производительность

Теперь о производительности: время «холодного» старта, равно как и «теплого» (повторный запуск программы без перезагрузки ОС, когда часть информации находится в оперативной памяти), в Kubuntu не изменилось, а в Windows XP оно, по личным ощущениям, сократилось на доли секунды. В RC2, по словам разработчиков, было снижено количество используемой оперативной памяти и последующая ее утечка, что потребовало немалых усилий. Переключение вкладок, прокрутка – все работает быстрее; особенно это заметно при больших объемах текста. Теперь Firefox работает с множеством вкладок без каких-либо проблем (вторая версия испытывала с этим трудности).

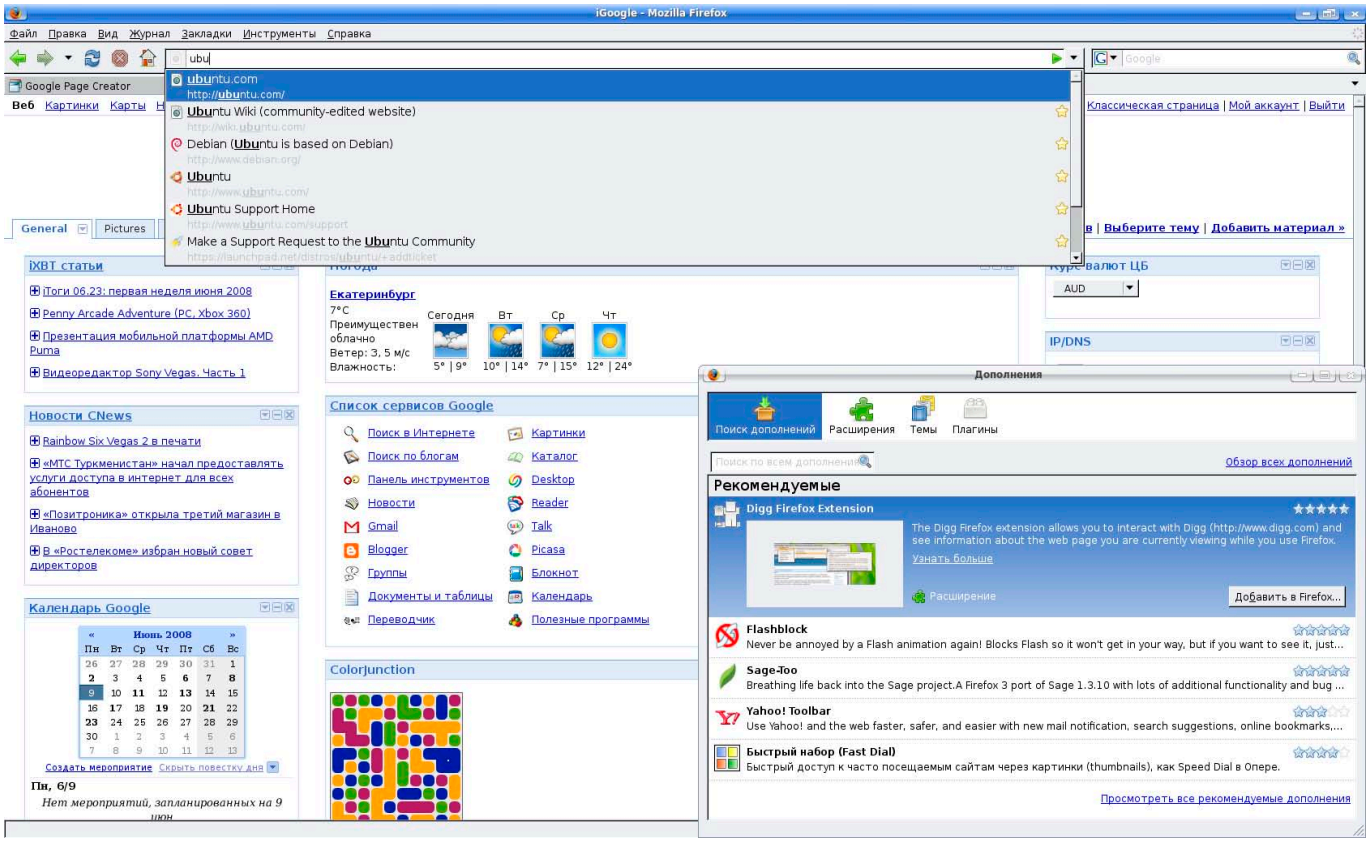
Отдельным пунктом повышения производительности у разработчиков значится улучшенная поддержка AJAX-приложений, в том числе и от Google. Перепробовав несколько приложений для работы в on-line, включая Gmail, я с этим скорее соглашусь. Однако о двойном или даже тройном приросте производительности (по сравнению с Firefox 2), как гласит «What's new» (http://www.mozilla.com/en-US/firefox/3.0/whatsnew), речи не идет.

Дополнения

Как я уже сообщал выше, Firefox при старте проверяет дополнения на совместимость, а затем на наличие обновлений. При возможности они обновляются сразу, как это было еще во второй версии. Старые, небезопасные дополнения, которые содержат ошибки, Firefox блокирует. Также он не дает обновлять с неизвестных источников уже установленные. Установка же упростилась за счет появления новой возможности – дополнения теперь устанавливаются и из менеджера дополнений. Причем по умолчанию некоторые там уже рекомендуются, а остальные можно подобрать с помощью поиска: либо по названию, либо по описанию. К сожалению, около 60% дополнений, используемых мною каждый день, работать отказались. В их числе – Google Toolbar и ImgLikeOpera. Впрочем, к выходу финальной версии ситуация должна улучшиться.

Стабильность

Под конец, пару слов о стабильности. Ни-



iGoogle в Firefox 3

каких нареканий нет: все отлажено, сбоев или вылетов не было, никаких зависаний за все время работы. И это несмотря на то, что RC2 позиционируется как нестабильная версия. (**Прим. ред.:** Лично у меня на amd64 при использовании Firefox 3 Beta5 наблюдались регулярные «падения» при работе с Gmail, которые пропали с выходом RC1.)

Резюме

Итак, продолжительная работа явно не пропала даром: браузер стал заметно стабильнее, безопаснее и быстрее. Работать с ним по-прежнему удобно и даже удобнее, чем было раньше. Приоритетные задачи, которые ставили перед собой разработчики, выполнены. Остался не менее важный этап – отладка. Мое заключительное мнение таково: Firefox 3 готов к использованию – с поправкой на работоспособность дополнений. Ведь расширяемость – это именно то, чем Firefox в свое время прославился. Миллионы пользователей, и я в их числе, ждут Download Day.

Никита Лялин
(tinman321@gmail.com)

xmonad: функциональный оконный менеджер

Сегодня доступно множество самых разных оконных менеджеров, но большинство из них вопреки названию (manage – «управлять» по-английски) перекладывают работу по управлению окнами на пользователя. Действительно, чтобы создать удобную для работы конфигурацию из нескольких окон, максимально освободив при этом пространство экрана, пользователь должен потратить ощутимое время на перемещение окон и изменение их размеров. Более того, с появлением очередного окна всю работу приходится проделывать заново.

Многим это надоедает настолько, что они просто распахивают окна на весь экран и переключаются между ними, но даже такая задача, как распахивание каждого нового окна, требует дополнительных телодвижений. Если какое-то действие приходится выполнять множество раз, хороший программист стремится его автоматизировать. Так возникли современные «тайловые» (от англ. tile – «черепица», «плитка») оконные менеджеры, которые берут на себя всю рутинную работу по управлению окнами. В каждый момент окна занимают весь экран, не перекрываясь и не оставляя зазоров.

Как раз об одном из таких оконных менеджеров мы собираемся рассказать. Имя ему – xmonad (читается «икс-монад», официальный сайт – http://www.xmonad.org). Конечно, это не единственный тайловый оконный менеджер, но комбинация следующих характеристик и возможностей делают его уникальным:

- ☑ стабильный, быстрый, лёгкий и простой;
- ☑ написан на чисто функциональном языке Haskell;
- ☑ возможность работы без использования мыши;
- ☑ поддержка Xinerama (многоэкранные конфигурации);
- ☑ поддержка плавающих окон, вкладок (табов) и декораций;
- ☑ интеграция со средами GNOME и KDE;
- ☑ индивидуальные компоновки окон на каждом рабочем столе;
- ☑ огромная и постоянно растущая библиотека расширений;
- ☑ развёрнутая и подробная документация;
- ☑ большая и активная команда разработчиков, дружелюбное сообщество.

xmonad состоит из двух частей: собственно ядро xmonad (которое может быть использовано как минималистский оконный менеджер) и постоянно растущая

библиотека расширений XMonadContrib (на момент написания статьи она насчитывала 127 отдельных модулей). Такое разделение позволяет ядру оставаться компактным и стабильным, и в то же время не ограничивает фантазию разработчиков.

В xmonad установлены высокие стандарты по отношению к качеству кода и документации. Каждая функция содержит четкое описание, а все расширения сопровождаются примерами использования. Перед тем как какой-либо патч поступит в репозиторий, он автоматически проверяется на наличие ошибок компиляции. Учитывая сильную статическую типизацию Haskell, это само по себе с большой вероятностью гарантирует, что код правильный. Кроме того, функции из ядра xmonad автоматически тестируются с помощью QuickCheck – более 100 различных инвариантов функций проверяются на случайным образом сгенерированных тестовых данных. Это обеспечивает высокую надёжность xmonad, так что пользователь не рискует, даже работая с нестабильной версией из репозитория darcs.

О чем мы не расскажем

Об xmonad можно говорить долго (что мы и собираемся делать), и тем не менее всё охватить невозможно. Дадим сразу список вопросов, которые не будут освещены в статье, со ссылками на полезную информацию по ним:

- **Интеграция со средой GNOME** – http://haskell.org/haskellwiki/Xmonad/Using_xmonad_in_Gnome.
- **Интеграция со средой KDE** – http://haskell.org/haskellwiki/Xmonad/Using_xmonad_in_KDE.
- **Использование трея.** Например, можно использовать stalonetray (http://stalonetray.sourceforge.net)
- **Использование многоэкранных конфигураций.** Особой настройки для неё не требуется, если правильно собрана библиотека X11 для Haskell. Отметим, что многие пользователи находят модель xmonad (каждый монитор – отдельный рабочий стол) намного удобнее, чем модель других оконных менеджеров (один большой рабочий стол).
- **Композитные расширения** – http://haskell.org/haskellwiki/Xmonad/Frequently_asked_questions#How_do_I_use_compositing_with_xmonad.3F.
- **Установка «обоев» (фона рабочего стола).** Для этого можно использовать любую программу, которая умеет выставлять фон корневого окна (root window), например, qiv или feh.
- **Использование urgencyHook** (http://www.xmonad.org/xmonad-docs/xmonad-contrib/XMonad-Hooks-UrgencyHook.html) для задания определённого действия оконного менеджера в ответ на то, что окно «хочет привлечь внимание пользователя». Это может быть полезным для отслеживания уведомлений различных программ.

Вообще, не стоит забывать, что в разделе часто задаваемых вопросов (http://haskell.org/haskellwiki/Xmonad/Frequently_asked_questions) вы сможете найти не только вопросы, но и ответы на них.

Установка

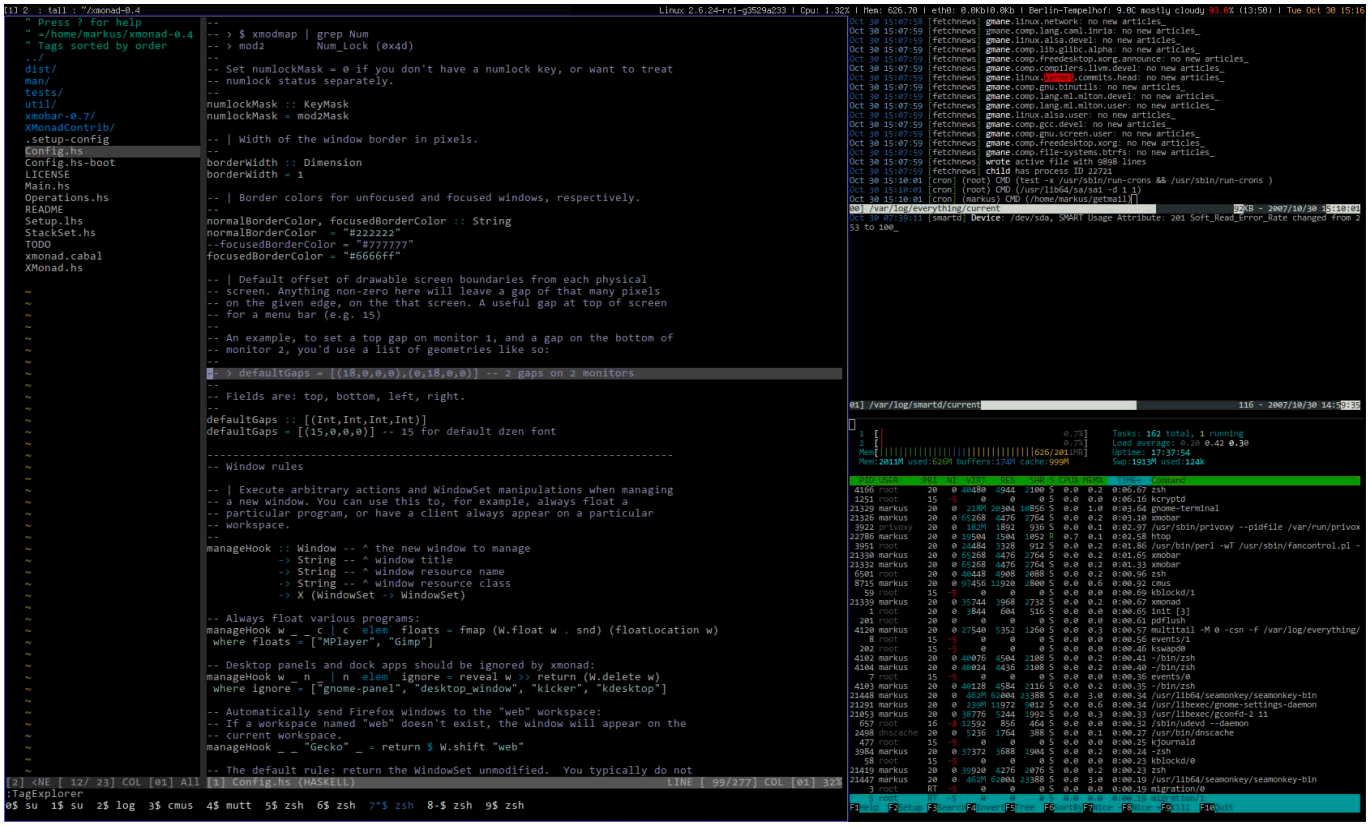
Сегодня многие операционные системы предлагают xmonad в виде готовых пакетов.

Среди систем, которые дружат с xmonad, числятся OpenBSD/NetBSD/FreeBSD, Gentoo, Debian, Ubuntu, Arch, Fedora, Slackware и другие. Обратите внимание, что для полноценной конфигурации необходимо установить GHC (компилятор Haskell) даже в том случае, когда сам xmonad распространяется в бинарном виде.

Если для вашей системы xmonad не упакован, есть отличный повод собрать его из исходников. Подробные инструкции приведены на сайте (http://xmonad.org/intro.html).

В этом случае, как и при сборке многих других программ, будет стоять выбор: собирать последний релиз (сейчас это версия 0.7, а версия 0.8 вот-вот выйдет) или разрабатываемую версию из darcs-репозитория (darcs – распределённая система контроля версий, используемая для разработки xmonad).

Как уже было сказано, разрабатываемая версия достаточно стабильна, а старые ошибки исправляются в целом чаще, чем привносятся новые.



Оконный менеджер xmonad

Ещё один важный шаг – установка документации. Если вы устанавливаете xmonad из пакетов для дистрибутива, туда скорей всего уже включена сгенерированная документация.

В противном случае её можно сгенерировать прямо из исходных текстов командой:

```
runhaskell Setup.lhs haddock
```

Для этого необходимо предварительно установить программу haddock.

В случае использования стабильного релиза документацию по нему можно найти в Сети (http://xmonad.org/xmonad-docs).

Конфигурация

Итак, мы установили xmonad. Теперь хорошо бы попробовать его в действии!

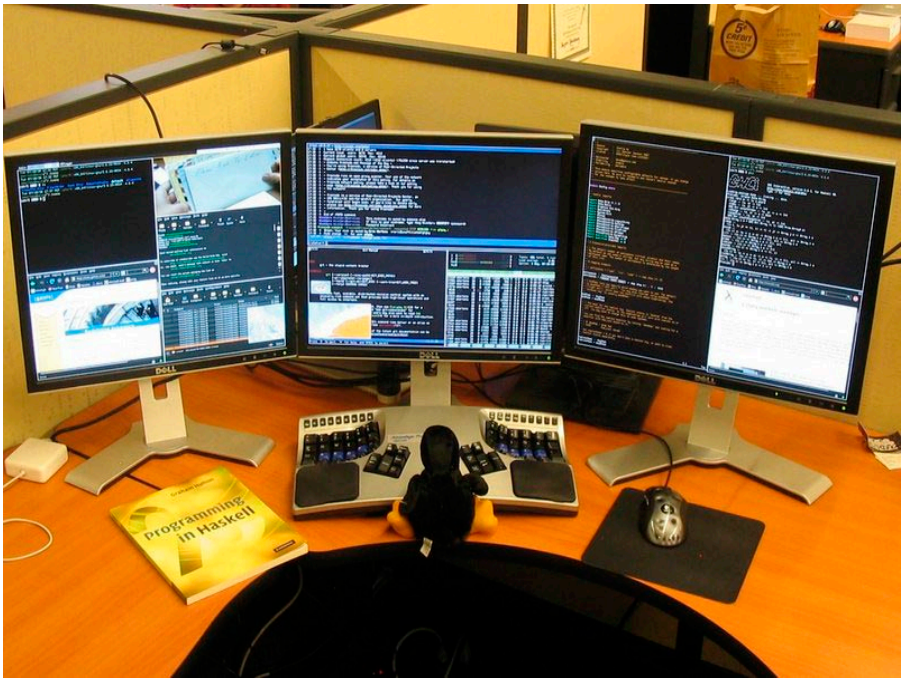
Если запуск X-сервера осуществляется из консоли с помощью команды startx или аналогичной, достаточно в $HOME/.xinitrc заменить вызов оконного менеджера на xmonad (с указанием полного пути к программе, если это необходимо). Если же используется менеджер дисплея (XDM/GDM/KDM/...), аналогичные изменения надо внести в файл $HOME/.xsession (если такого нет, то достаточно создать его и вписать туда xmonad). Создатели пакета могли позаботиться о том, чтобы в меню KDM или GDM появился тип сессии xmonad, но мы рекомендуем использовать файл запуска .xsession, для чего надо выставить в GDM/KDM тип сессии Default.

Поскольку xmonad ориентирован на работу с клавиатурой, полезно ознакомиться с основными сочетаниями клавиш (о том, как их можно изменить, см. ниже). Их перечень доступен в man xmonad (или по адресу http://xmonad.org/manpage.html). Основные из них:

- ☑ **Alt+Shift+Return** – запуск терминала;
- ☑ **Alt+Shift+C** – закрытие текущего окна;
- ☑ **Alt+J** и **Alt+K** – переключение между окнами;
- ☑ **Alt+Shift+Q** – выход из xmonad.

Если предварительно никакие особые действия не предпринимались, после запуска xmonad вы, вероятно, увидите однотонный фон рабочего стола и никаких признаков наличия оконного менеджера. Сделать вывод об успешном запуске xmonad можно, запустив терминал указанным выше сочетанием клавиш. Дальнейшие инструкции помогут сделать xmonad более приветливым.

Вся конфигурация xmonad производится в конфигурационном файле $HOME/.xmonad/xmonad.hs. XMonad исповедует принцип «создай свой оконный менеджер сам» не только в переносном смысле (предлагая возможность гибкой настройки), но и в прямом – конфигурационный файл представляет собой программу на языке Haskell. Будучи скомпилированной, эта программа и будет вашим менеджером окон, а пакеты xmonad и xmonad-contrib, которые вы установили, – это по сути библиотеки, которые делают создание своего оконного менеджера практически тривиальным.



Многоэкранная конфигурация

Тем не менее даже если вы первый раз слышите о языке Haskell, не стоит пугаться – о вас разработчики xmonad думают в первую очередь! Декларативность языка и широкие синтаксические возможности делают Haskell отличным языком для описания конфигурации, в чём мы и предлагаем убедиться.

Нет никакой необходимости начинать с нуля. Вашему вниманию предлагается тщательно документированный образец xmonad.hs (http://haskell.org/haskellwiki/Xmonad/Config_archive/Template_Config.hs) (если вы устанавливали xmonad из исходников, то этот же самый файл можно найти в директории с исходниками по адресу man/xmonad.hs). Кроме того, на wiki-странице http://haskell.org/haskellwiki/Xmonad/Config_archive можно найти десятки конфигурационных файлов от пользователей xmonad.

Дадим минимум знаний о синтаксисе Haskell, который понадобится для написания конфигураций. Блочные комментарии заключаются между последовательностями «{-» и «-}» (фигурные скобки и дефис), а однострочные начинаются с "--" (два дефиса). Язык – регистрозависимый, то есть путать большие и маленькие буквы нельзя. Кроме того, как и в языке Python, отступы и переводы строк играют определенную роль. Поэтому для уменьшения количества ошибок рекомендуется всегда использовать пробелы вместо табуляций. Если у вас возникли проблемы, не стесняйтесь зайти в IRC на #xmonad (в сети Freenode) и попросить помощи.

Теперь – об основных частях конфигурационного файла. Если не считать комментарии, конфигурационный файл начинается с объявлений import. Каждое такое объявление должно располагаться в начале строки и говорить компилятору о необходимости подключить соответствующий модуль. Объявление:

```
import XMonad
```

должно присутствовать в каждом конфигурационном файле. Кроме того, каждое расширение, которое вы будете подключать, как правило, потребует от добавления еще одного import.

Следующей важной частью xmonad.hs является строчка:

```
main = xmonad defaults
```

Поскольку xmonad.hs является программой, это объявление указывает, что при её запуске надо исполнить функцию xmonad (которая и отвечает за управление окнами) с аргументом defaults – структурой данных, содержащей все настройки, такие как сочетания клавиш, компоновки окон и прочие.

Вот как выглядит определение defaults:

```
defaults = defaultConfig {
        terminal           = myTerminal,
        focusFollowsMouse  = myFocusFollowsMouse,
        borderWidth        = myBorderWidth,
        modMask            = myModMask,
        numlockMask        = myNumlockMask,
        workspaces         = myWorkspaces,
        normalBorderColor  = myNormalBorderColor,
        focusedBorderColor = myFocusedBorderColor,
        keys               = myKeys,
        mouseBindings      = myMouseBindings,
        layoutHook         = myLayout,
        manageHook         = myManageHook,
        logHook            = myLogHook,
        startupHook        = myStartupHook
    }
```

Здесь использована стандартная форма записи структур данных в Haskell. Слева от знака «=» стоят названия параметров, а справа – значения.

В данном случае все значения представляют собой переменные, которые определены в других местах конфигурационного файла. Смысл параметров таков:

- **terminal** – строка, содержащая команду для вызова предпочтительного терминала (например, «xterm»).
- **focusFollowsMouse** – отвечает за то, будет ли фокус ввода перемещаться, если вы перемещаете указатель мыши.
- **borderWidth**, **normalBorderColor** и **focusedBorderColor** – относятся к рамке, которая рисуется вокруг каждого окна.
- **modMask**, **numlockMask**, **keys** и **mouseBindings** – относятся к сочетаниям клавиш и будут объяснены ниже.
- **workspaces** – список строк, которые будут идентифицировать рабочие столы. По умолчанию они просто пронумерованы от 1 до 9. Можно выделять рабочие столы под отдельные задачи. Например, ["irc", "mail", "www", "work"].
- **layoutHook** – содержит настройки компоновок окон. manageHook позволяет производить определённые действия при открытии нового окна. logHook поможет сделать строку статуса. startupHook может выполнять действия при старте xmonad. К этим параметрам мы ещё вернёмся.

Что дальше?

На этом мы заканчиваем первую из трёх частей цикла об оконном менеджере xmonad. В следующих частях вы узнаете о том, как изменить стандартные сочетания клавиш, как воспользоваться всем богатством алгоритмов компоновки окон, как использовать строку статуса, а также о многих других возможностях, предоставляемых xmonad.

Иван Веселов
(veselov@gmail.com)
Роман Чепляка
(roma@ro-che.info)

Защищаем себя средствами GnuPG

Предисловие

В последнее время очень остро стоит проблема сохранения конфиденциальности информации. Особенно в Интернете, где риск перехвата секретных данных весьма высок. В этой статье будет представлено описание работы пакета GnuPG (GNU Privacy Guard, GPG) (http://www.gnupg.org) вкупе с несколькими примерами применения.

GnuPG служит для создания цифровых подписей и шифрования данных. Например, вопрос идентификации писем всегда был актуальным. С помощью GnuPG можно «вложить» в письмо электронную подпись. И таким образом получатель определит подлинность отправителя и принадлежность ему этого письма. Процесс работы GnuPG весьма прост: за сложнейшими алгоритмами шифрования скрыта простая логика: используется пара ключей, один из которых является приватным (вы его держите у себя), а второй – публичным (он свободно бороздит просторы Сети).

Файл второго содержит публичный ключ и подписи ваших респондентов. Получается, что после доставки подписанного письма получатель сравнивает публичные ключи и таким образом идентифицирует отправителя.

Особенности GnuPG

Основные технические особенности GnuPG таковы:

- полноценная альтернатива PGP;
- не использует патентованные алгоритмы;
- распространяется под лицензией GPL;
- полная реализация OpenPGP (RFC4880);
- расшифровывание и аутентификация сообщений, созданных с помощью PGP 5, 6 и 7;
- поддержка электронной подписи с помощью алгоритмов ElGamal, DSA, RSA и хеш-функций MD5, SHA-1, RIPE-MD-160 и TIGER;
- работа с асимметричным шифрованием ElGamal и RSA (длина ключа от 1024 до 4096 бит);
- поддержка блочных алгоритмов симметричного шифрования AES, 3DES, Blowfish, Twofish, CAST5, а также IDEA с помощью модуля;
- лёгкая реализация новых алгоритмов с помощью дополнительных модулей;
- многоязыковая поддержка (в том числе и русского);
- on-line-система помощи;
- поддержка просроченных ключей и подписей;
- встроенная поддержка HKP-серверов ключей.

Как уже было отмечено, GnuPG был разработан в соответствии со стандартом OpenPGP, а это значит, что подписи и зашифрованные данные, созданные другими программами, совместимыми с OpenPGP, будут работать с GnuPG. Использование различных криптографических алгоритмов, таких как симметричные шифры, шифрование с открытым ключом и смешанные алгоритмы, позволяет надёжно защищать секретные данные и передавать их. Длины ключа в 1024 или 2048 бит достаточно, чтобы не беспокоиться о взломе зашифрованной информации.

GnuPG (http://www.gnupg.org) – исключительно консольная программа, но уже сейчас существует несколько графических оболочек для неё: Seahorse (http://www.gnome.org/projects/seahorse) и GPG-Crypter (http://gpg-crypter.sourceforge.net – которые упрощают работу с программой посредством интуитивно понятного графического интерфейса.

Работа с GnuPG

Первое взаимодействие с пакетом GnuPG начинается с генерирования ключей:

```
$ gpg --gen-key
```

Программа задаст несколько вопросов о длине ключей, имени и адресе электронной почты. Затем нужно будет ввес-

ти пароль для защиты ключа. Таким образом будет создана пара ключей, один из которых будет основным. Его стоит использовать для шифрования самых важных данных.

Поскольку вероятность взлома есть всегда, основной ключ лучше использовать для подписи в крайних случаях. Также можно создать ещё несколько подключей, которым по усмотрению пользователя могут быть заданы другие алгоритмы шифрования, если не требуется повышенного уровня секретности данных.

Такие подключи будут зависеть от основного и могут использоваться для шифрования документов или переписки. Другими словами, для каждого способа связи – свой ключ.

У каждого из них есть срок использования (так же, как и у кредитных карт). Хорошим тоном является установка срока использования для подключей 1-2 года. GnuPG ведёт собственную базу, которая находится в файле ~/.gnupg/pubring.gpg. Туда и заносятся открытые (публичные) ключи ваших респондентов.

С помощью команды:

```
$ gpg --list-keys
```

можно просмотреть все ключи, находящиеся в базе. Будет выведен список ключей, показывающий их статус (pub – публичный, sub – второстепенный), длину и метод шифрования, дату создания и, главное, уникальный идентификатор (ID), представляющий собой 8-значное 16-ричное число.

Для основного ключа можно (и нужно) создать отзывающий сертификат:

```
$ gpg --gen-revoke $KEY
```

где $KEY – ID основного ключа.

Отзывающий сертификат нужен для уничтожения ключа. Это может потребоваться, например, если ключ будет украден или утерян. Даже если ключевая фраза очень надёжна, стоит заранее, еще на этапе создания ключа, подумать о возможности его уничтожения в будущем. После создания сертификата его содержимое будет выведено в stdout. Его нужно сохранить в надёжном месте (желательно на другом носителе или вообще в печатном виде), т.к. любой, завладевший этим сертификатом, может сделать ключ недействительным и удалить его из базы данных сервера открытых ключей (тогда никто не сможет получить ваш ключ). Такие базы данных хранят публичные ключи совершенно свободно. Это сделано для удобства обмена ключами: вам необязательно постоянно передавать ваш публичный ключ лично. Можно воспользоваться несколькими способами: разместить у себя на домашней странице, на портале, в котором есть поле для публичного ключа, либо воспользоваться более централизованной базой – копилкой ключей, из которой достать ваш ключ будет всегда удобно.

Чтобы использовать сертификат, нужно просто импортировать его в базу, как и любой открытый ключ:

```
$ gpg --import revoke-certificate.asc
```

а затем отправить на сервер:

```
$ gpg --send-keys $KEY
```

где $KEY – D ключа, который будет отправлен.

Точно так же можно отправлять публичные ключи и хранить их на серверах баз данных. Для этого в первую очередь их нужно экспортировать в общую локальную базу командой:

```
$ gpg --import $FILE
```

где $FILE – файл ключа или keyring («связка», несколько ключей в одном файле).

После этого командой:

```
$ gpg --sign-key $KEY
```

где $KEY – ID ключа респондента, нужно подписать желаемый ключ. При подписывании к нему добавляется ваш публичный ключ для того, чтобы ваши сообщения/письма могли идентифицировать другие. Затем нужно отправить подписанный вами ключ его владельцу:

```
$ gpg --export $KEY > userkey.gpg
```

Эта команда извлекает подписанный ключ отдельно для удобства отправки.

Можно сделать то же самое в виде ASCII-текста, который легко разместить в Сети:

```
$ gpg -a --export $KEY > userkey.asc
```

где $KEY – ID ключа владельца.

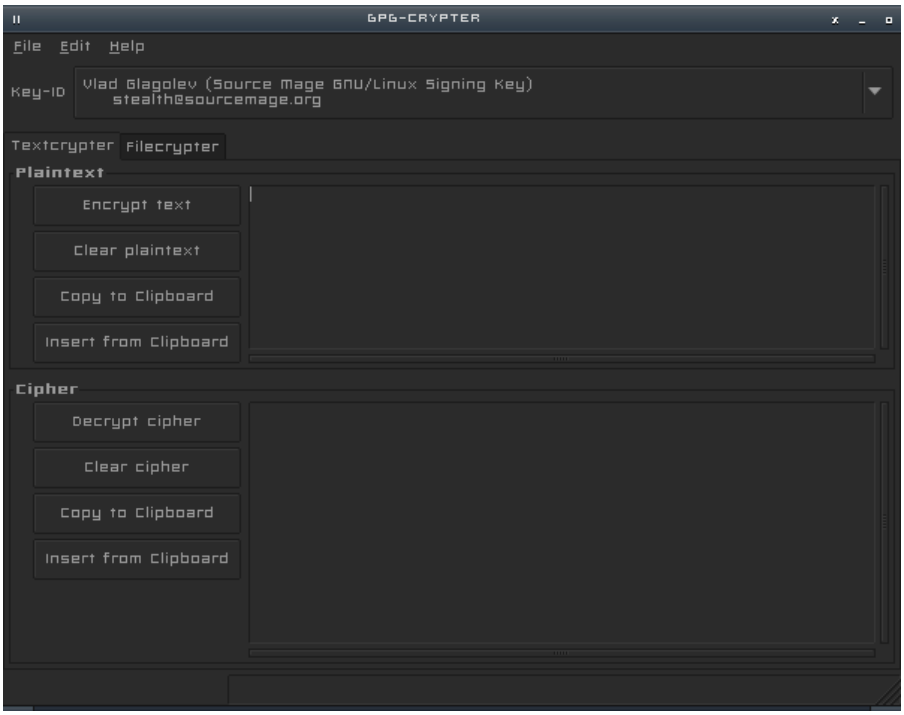
Теперь владелец должен импортировать этот ключ к себе, чтобы ваша подпись находилась у него в базе. Экспорт такого ключа производится точно так же, как и любого другого публичного ключа. Затем владелец отправляет его на сервер баз данных ключей или выкладывает на свой сайт. Теперь его ключ содержит вашу подпись. И идентификация сообщений позволит удостовериться, что они дошли именно от вас.

Иногда может потребоваться хранить ваши ключи (публичные или приватные) на каком-либо носителе (например, на USB flash). Для этого нужно экспортировать ключ, что можно сделать как в бинарном виде:

```
$ gpg --export $KEY > mykey.gpg
```

Так и в текстовом (ASCII armor):

```
$ gpg -a --export $KEY > mykey.asc
```



GPG-Crypter

В обоих случаях $KEY – это ID вашего ключа. Вместо «-a» можно также использовать «--armor».

В итоге для работы с документами доступны следующие команды:

- подписать документ (гарантирует, что документ «от вас»), к нему просто добавляется ваша электронная подпись;
- зашифровать документ (производится шифровка выбранным алгоритмом всего документа);
- подписать и зашифровать документ (сочетает в себе эти действия).

Вне зависимости от типа файла (как было показано выше с ключом) можно получить подпись или зашифрованное сообщение как в бинарном, так и в текстовом виде. Например, есть файл библиотеки – он двоичный. Шифруем и подписываем его, а на выходе получаем текстовый файл. После расшифровки файл приходит в своё оригинальное состояние. Это можно использовать для хранения различных файлов в таблицах реляционных баз данных: в таком случае, несмотря на различные типы файлов, после зашифровки все они будут представлять набор символов в виде строк ASCII.

Можно создавать так называемые прозрачные подписи (в которых будет незашифрованное содержимое документа + ваша цифровая подпись):

```
$ gpg --clearsign $DOC
```

где $DOC – путь к документу. Таким образом, будет создан файл $DOC.asc, в котором само содержание документа открыто и добавлена его цифровая подпись.

А подписи, находящиеся в отдельных файлах в бинарном виде (будет создан файл подписи $DOC.sig), создаются командами:

```
$ gpg --detach-sign $DOC
```

В текстовом (ASCII armor) виде (будет создан файл подписи $DOC.asc):

```
$ gpg -a --detach-sign $DOC
```

Такие подписи (в последних двух примерах) должны распространяться вместе с подписываемым документом.

Любой ключ также можно отредактировать командой «--edit-key». Это позволит изменить некоторые параметры ключа: степень достоверности, если это чужой публичный ключ, секретную фразу, если это ваш приватный ключ, и другое.

Что касается степени достоверности, то в GnuPG существует 5 уровней:

- **1. I don't know or won't say** – я ничего не знаю о владельце этого ключа или не хочу говорить об этом;
- **2. I do NOT trust** – я не доверяю этому человеку;
- **3. I trust marginally** – я знаю этого человека и доверяю ему, но не уверен, что ключ принадлежит ему;
- **4. I trust fully** – я знаю этого человека и лично убедился в том, что ключ принадлежит ему;
- **5. I trust ultimately** – я знаю этого человека, у меня есть доступ к его секретному ключу.

GnuPG и Open Source

GnuPG существует практически в каждом дистрибутиве GNU/Linux, является обязательным пакетом в OpenBSD, NetBSD, FreeBSD и других свободных операционных системах.

Множество Open Source-приложений поддерживают GnuPG посредством различных модулей.

Например, gpgme (библиотека, являющаяся неким посредником между GnuPG и программами) используется следующими приложениями:

- Почтовые клиенты Evolution (входит в состав GNOME) и Sylpheed (http://sylpheed.sraoss.jp/en), а с помощью расширения Enigmail GnuPG работает в почтовом клиенте Mozilla Thunderbird.
- Jabber-клиенты Gajim (http://www.gajim.org) и Psi (http://psi-im.org).
- KDE PIM (Personal Information Management).

У упомянутой библиотеки gpgme есть и модули для скриптовых языков. Например:

- pygpgme (http://launchpad.net/pygpgme) для Python;
- Crypt_GPG (http://pear.php.net/package/Crypt_GPG) для PHP;
- Crypt::GpgME (http://search.cpan.org/dist/Crypt-GpgME) для Perl.

В последнее время всё больше проектов используют GnuPG для подписывания файлов с целью дальнейшей проверки их целостности (вместо использования хешей MD5, SHA1, SHA256, SHA512). Поэтому помимо архивов (особенно в Open Source-среде) на сайтах обычно лежат GPG-подписи в бинарном или ASCII-видах.

Один из основанных на исходниках (так называемых source-based) Linux-дистрибутивов использует GnuPG как основной инструмент проверки целостности файлов в системе управления программным обеспечением – Source Mage GNU/Linux (http://www.sourcemage.org).

Бинарные дистрибутивы Linux также зачастую используют пакеты, подписанные GnuPG. Например, в случае с пакетами DEB – это Debian GNU/Linux и Ubuntu, а с RPM – Red Hat, Fedora, Mandriva и многие-многие другие.

Итоги

В этой статье мы познакомились с GnuPG – свободным средством защиты информации. Теперь можно не беспокоиться о том, что ваши секретные данные будут утрачены или обнародованы. Этот принципиально новый подход (со времен создания OpenPGP) к шифрованию с точки зрения обычного пользователя позволяет и по сей день решать сложные задачи, связанные с передачей особо важных данных.

Влад Глаголев
(stealth@sourcemage.org)

Подписные индексы:

20780*
81655**
по каталогу агентства «Роспечать»

88099*
87836**
по каталогу агентства «Пресса России»

* **годовой**
** **полугодовой**

Стоимость подписки через редакцию:

900* руб. за 6 номеров

1800* руб. за 12 номеров

Подписка на журнал «Системный администратор»

Российская Федерация

- ☑ Подписной индекс: годовой – **20780**, полугодовой – **81655**
  Каталог агентства «**Роспечать**»
- ☑ Подписной индекс: годовой – **88099**, полугодовой – **87836**
  Объединенный каталог «**Пресса России**»
  Адресный каталог «**Подписка за рабочим столом**»
  Адресный каталог «**Библиотечный каталог**»
- ☑ Альтернативные подписные агентства:
  агентство «**Интер-Почта**»
  (495) 500-00-60, курьерская доставка по Москве
  агентство «**Вся Пресса**»
  (495) 787-34-47
  агентство «**Курьер-Прессервис**»
  агентство «**ООО Урал-Пресс**»
  (343) 375-62-74
- ☑ Подписка On-line
  http://www.arzi.ru
  http://www.gazety.ru
  http://www.presscafe.ru

СНГ

В странах СНГ подписка принимается в почтовых отделениях по национальным каталогам или по списку номенклатуры «**АРЗИ**»:

- ☑ **Азербайджан** – по объединенному каталогу российских изданий через предприятие по распространению печати «**Гасид**» (370102, г. Баку, ул. Джавадхана, 21)
- ☑ **Казахстан** – по каталогу «**Российская пресса**» через ОАО «**Казпочта**» и ЗАО «**Евразия пресс**»
- ☑ **Беларусь** – по каталогу изданий стран СНГ через РГО «**Белпочта**» (220050, г. Минск, пр-т Ф. Скорины, 10)
- ☑ **Узбекистан** – по каталогу «**Davriy nashrlar**», российские издания через агентство по распространению печати «**Davriy nashrlar**» (7000029, г. Ташкент, пл. Мустакиллик, 5/3, офис 33)
- ☑ **Армения** – по списку номенклатуры «**АРЗИ**» через ГЗАО «**Армпечать**» (375005, г. Ереван, пл. Сасунци Давида, д. 2) и ЗАО «**Контакт-Мамул**» (375002, г. Ереван, ул. Сарьяна, 22)
- ☑ **Грузия** – по списку номенклатуры «**АРЗИ**» через АО «**Сакпресса**» (380019, г. Тбилиси, ул. Хошараульская, 29) и АО «**Мацне**» (380060, г. Тбилиси, пр-т Гамсахурдия, 42)
- ☑ **Молдавия** – по каталогу через ГП «**Пошта Молдовей**» (МД-2012, г. Кишинев, бул. Штефан чел Маре, 134)

по списку через ГУП «**Почта Приднестровья**» (MD-3300, г. Тирасполь, ул. Ленина, 17)

по прайс-листу через ООО агентство «**Editil Periodice**» (МД-2012, г. Кишинев, бул. Штефан чел Маре, 134)

- ☑ Подписка для **Украины**:

Киевский главпочтамт
Подписное агентство «**KSS**»
Телефон/факс (044)464-0220