

g2k14: inside

Содержание

1 g2k14	1
2 Вадим Жуков	1
2.1 World of KDE	1
2.2 Отзывы по самому мероприятию	3
3 Theo de Raadt: безопасность и конфигурашность	4
4 Матье Херб (matthieu@) о развитии X	4
5 Марк Эспи (espie@) о портах и пакетах	5
6 Мартин Пеликан (pelikan@) об ext4 и о файловых системах	6
7 Jonathan Gray об улучшении драйверов для X	7
8 Об этом документе	8

Все персонажи являются вымышленными, а любое совпадение - случайным
от автора

1 g2k14

g2k14 - это хакафон OpenBSD, который проходил в городе Любляна (Словения) с 8 по 14 июля.

Из отзывов участников мы можем узнать, в каком направлении развитие OpenBSD движется и будет двигаться дальше. Это своего рода неформальные программные статьи, которые отвечают на вопрос "Как живёшь, OpenBSD?"

На этом мероприятии были замечены и два русских разработчика OpenBSD: Вадим Жуков (zhuk@) и Кирилл Бычков (kirby@)



Важно

Прорабатывается вопрос о представлении их к правительственным наградам **за участие!**

2 Вадим Жуков

2.1 World of KDE

По горячим следам успешного хакафона, Вадим Жуков (zhuk@) отчитался о своих успехах:

Я прибыл на хакафон с коротким, но суровым списком задач:

- Закончить KDE 4.13.2 и приготовить 4.13.3 (официальный анонс - 15 июля*)
- Наконец-то портировать несколько приложений из openbsd-wip в официальный cvs
- Исправить давнюю проблему с усиленным поеданием процессора в kded4
- Продолжить работу над Samba 4.x
- Исправить проблему с отсутствием ext2fs в установщике для amd64 (RAMDISK_CD)
- Некоторые вещи, которые я разбабывал последние месяцы для ports/infrastructure, занести в CVS
- Занести в CVS порт man-pages-posix

Замечание

* Разработчики KDE дают возможность мейнтейнерам пакетов с KDE в той или иной ОС иметь т.н. предварительный доступ, где-то дней за 3-5 до официального релиза. Это позволяет выпускать "родные" пакеты с KDE для ОС одновременно с официальным анонсом релиза.

Но прежде всего хакафон для вас начинается со знакомства с людьми, с которыми вы не были прежде знакомы. Учитывая, что до этого единственным мероприятием, связанным с OpenBSD, которое я посещал, была конференция EuroBSDCon 2013, на хакафоне было много новых лиц.

Боюсь, что не запомнил их всех, но не потому, что я не уважаю их или их работу, это просто мой недостаток :)

Итак, хакафон начался. Мы с kirby@ - другим портером OpenBSD из России - сели друг напротив друга. И это нам очень помогло - он помог мне тестить сборку ядра с ext2fs и дал мне идею насчёт *libinotify* (см. ниже), я помог ему обновить порт rawtherapee.

Мой первый коммит на этом хакафоне был занесением в CVS *books/man-pages-posix*. Это полезная вещь для разработчиков, и я получил положительные отзывы ещё до того, как начал это импортирование.

Это был не столько мой труд, сколько schwarze@ и других разработчиков, давших большое количество отзывов и замечаний. Я узнал много нового о mandoc, groff и pkg_create во время работы над этим портом. Но, опять же, это было только для разгона.

Большую часть времени я сидел и делал четыре вещи: запускал make, твикал патчи, пушил их в апстрим и засыпал landry@ новыми портами. благодарен ему за терпение. Благодаря его отзывам*, у нас теперь есть следующие приложения из KDE4: *Calligra Suite*, *Digikam*, *K3b*, *Kdenlive*, *KDevelop*, *KMyMoney*, *KTorrent*, *Tellico* и *Yakuake* (вместе с зависимостями, типа Eigen 3.x).

Замечание

* Отзывы (ревью) могут быть и без замечаний, но без ревью занесение в порты не делается.

Из портов, связанных с KDE4, в openbsd-wip осталась только audio/cantata: она имеет несколько кривую интеграцию с KDE4, так что мне быстро это надоело - плееров, в том числе для KDE4, и так хватает. Надеюсь, что Рафаэль Садовски, который постоянно мне помогал, не обидится. :)

Обновление KDE 4.13.2 само по себе скучно и неинтересно. Имеем 200+ портов, значит, 200+ раз пишем *make configure update-plist port-lib-depends-check package clean*, отправляем несколько патчей в апстрим, закончили упражнение. Вот и всё. Реально всё. Трудными были задачи собственно портирования KDE4, а также совместного существования KDE3 и KDE4, а поддержка портов KDE4 не так сложна.

И вот пришло время для действительно интересенького. kded4. Если вы не в курсе подробностей: kded4 (что означает "KDE 4 Daemon") обычно запускается с kdeinit... то есть, либо в самом начале сессии kde, либо когда вы запускаете первое приложение KDE. Этот демон хостит так называемые модули KDE - Если вы видели services.exe в Windows, то вы понимаете, о чём я, это почти то же самое. Другая задача kded4 - мониторить файлы конфигурации, особенно связанные с MIME файлы .desktop. При установке/настройке/удалении приложения .desktop-файлы могут изменяться, как системные (в /usr/local), так и ваши личные (в \$KDEHOME). Многие программы, особенно различные виджеты рабочего стола (читай: KDE-меню и подобное), заинтересованы в уведомлениях о таких изменениях. Таким образом, kded4 мониторит некоторые директории на предмет добавления/изменения/удаления .desktop-файлов.

В OpenBSD этот процесс был очень неэффективен. А причина в том, что kded4 внутри использует KDirWatch, который по умолчанию использует inotify в Linux и QFSWatch в других операционных системах. Он также поддерживает FAM, но я уже пытался его использовать, но результаты меня не удовлетворили. Я уже начал было думать о реализации бэкэнда на базе kqueue(2), и тут я вспомнил, что kirby@ работает над libinotify. Это ведь то, что нужно - inotify API на базе kqueue. Так что я написал FindInotify.cmake который должен работать и в Linux и вне Linux, сделал несколько #ifdef в коде, пересобрал kdelibs ... и вот оно! Теперь kded4 проверяет файлы при запуске, и дальше живёт абсолютно не напрягаясь!

Ещё после этого akonadi_maildir_resource перестал жрать ресурсы: похоже, он страдал той же проблемой. Две проблемы по цене одной! Покупайте наши *libinotify*! *



Важно

* все пользователи KDirWatch теперь едят намного больше файловых дескрипторов (до нескольких тысяч - по сути, по дескриптору на каждый отслеживаемый каталог и файл). На Linux эта проблема не так заметна, так как там обычно банально не стоит никаких лимитов, или они задраны очень высоко.

Кроме того, за время этого хакафона я успел закончить:

- новую утилиту `portbump(1)`, в связке с `sqlports` она позволяет сэкономить много времени на масштабных обновлениях.
- добавил переменные `TEST_ENV` и `ALL_TEST_ENV` в `bsd.port.mk`: одного `TEST_FLAGS` было явно недостаточно, поскольку некоторые порты на CMake (читай: использующие Ninja) не понимают `TEST_FLAGS` вообще.
- документацию для `devel/cmake` и `x11/kde4`. Не имею намерения документировать `x11/kde`, потому что его больше никто не собирается поддерживать, а кто поддерживает сейчас, и так всё знает.

К сожалению, не хватило времени на *samba4*. Есть хитрые проблемы, связанные с `ld.so` и компилятором, которые я надеялся исправить на хакафоне... но не всё сразу. Так или иначе, KDE был приоритетной задачей.

Также я не раз участвовал в дискуссиях и протестировал несколько патчей, летающих повсюду. И, даже в случае ошибок, мне доставлял радость тот факт, что я могу помочь другим разработчикам OpenBSD напрямую, что, как правило, проблематично в обычной жизни.

В заключение, я хочу (чувствую необходимость, обязан) сказать спасибо Мите Муженичу и гостевому дому "Табор" за организацию этого чудесного мероприятия. Это был мой первый хакафон, и было удивительно, сколько всего произошло за несколько дней. И Любляна - прекрасный город... Я надеюсь что кто-то, кто знает английский язык лучше меня, сможет ярче живописать этот уютное место и его жителей. Всё было просто классно - спасибо, спасибо и еще раз спасибо!

2.2 Отзывы по самому мероприятию

Ааа, завидую! В хорошем смысле! А Тео там тоже был?

Естественно. Это был ежегодный большой "всеобщий" хакатон.

Можно ли поподробнее описать как проходят хакафоны?

1) как туда попасть?

Получить приглашение. :)

2) много ли людей участвует

Когда как. Статистика на <http://www.openbsd.org/hackathons.html>

3) где все спят

Обычно устроитель хакатона обеспечивает (сам или при финансовой помощи OpenBSD Foundation) места в каком-нибудь общежитии (ныне их хостелами кличут). Кто не хочет - селится сам в какой-нибудь гостинице.

4) как общаются

В основном на английском. :) Все в одной комнате, плюс есть общий чат (чтобы ор постоянный не стоял и друг друга не отвлекать от важных процессов слишком часто). Ну а локальные группки нередко формируются по языковому принципу, и там уже болтают на родном языке.

5) сколько дней всё это длится

Когда как, обычно - около недели. Опять же, статистику можно собрать по ссылке выше.

3 Theo de Raadt: безопасность и конфигурашность

Лидер проекта OpenBSD Тео де Раадт (deraadt@) пишет об g2k14:

Две недели перед Словенией я работал с Бобом Беком (Bob Beck) над заменяющими `getentropy(2)` функциями. В начале хакафона были внесены последние штрихи, нужные Бобу и Бренту Куку (Brent Cook) для дальнейшей работы.

Затем пришло время разбираться с очередной проблемой безопасности, о которой мне стало известно. К нашему прискорбию выяснилось, что исчерпание ограничения на количество одновременно открытых файлов может быть использовано для сокрытия уведомлений о переполнении стека от соответствующего механизма защиты. Защитнику стека требуется файловый дескриптор, чтобы сообщить об ошибке. Те, кто уже читал заметки об `arc4random` и `getentropy`, уже в курсе данной ситуации. (*)

Замечание

* Речь об исчерпании лимита на количество открытых файлов, которое в случае использования `syslog(3)` могло привести к тому, что сообщения об ошибках не попадут в системный журнал. Дело в том, что `syslog(3)` оперировал через открытие файла `/dev/log`, которое, в случае исчерпания оных лимитов, становится невозможным.

Проблема стала очевидной из-за технологии "песочницы", используемой ныне в SSH-утилитах, которая закрыла `syslog_r()` доступ к `socket()`, `connect()`, `sendto()`... всем системным вызовам, необходимым для сообщения об ошибке, но потенциально опасным - что как раз "песочница" и должна предотвращать.

Задача была решена путём создания нового системного вызова, который может отправить сообщение в `syslogd` без использования лишних ресурсов; `syslog_r(3)` теперь использует его напрямую: один щелчок, выстрел, поехали дальше. Данный системный вызов имеет более чем узкое применение, и поэтому был назван `sendsyslog(2)`, и при этом он также подходит для специфических условий, таких как использование "песочницы".

В этом плане, ситуация схожа с тем, как `getentropy(2)` была вынесена из `sysctl`. Забавно, как одно приводит к другому.

В качестве передышки от пространства ядра, пришла пора для небольшой уборки и, надеюсь, улучшения в `/etc`, `sysmerge` и инструментах установки. Роберт и Антуан помогли спланировать практически пустой `/etc/rc`, эта работа ещё не окончена, но приведёт к улучшенному `sysmerge`. На других фронтах я работал с теми, кто занимается установочными скриптами и DRM, чтобы в нашем следующем релизе можно было автоматически по возможности прикрывать прямой доступ к оборудованию для X на поддерживаемых в этом плане чипсетах

Замечание

по сути это современные Intel - прим. ред.

В остальное время хакафона я мелькал тут и там, как обычно, участвуя в проектах других разработчиков. Очень приятная и производительная неделя!

4 Матье Херб (matthieu@) о развитии X

Матье "бешеный француз" Херб (matthieu@), поддерживающий Xenocara, хочет поделиться своими впечатлениями о g2k14:

Я так и ничего и не сделал по моим остальным проектам (мультиач, DHCPv6), поскольку был отвлечен на твики наборов для X, по просьбе нескольких других участников. Результатом долгой дискуссии стало лишь добавление `usr` в базовую систему (после недолго пребывания в `/usr/xenocara/app/xrdb-cpp`) под именем `/usr/libexec/auxcpp`.

Причина в том, что `xrdb` (часть необходимой многим портам `xbase`) требует препроцессор `C` для запуска. Но, начиная с `gcc4`, `/usr/bin/cpp` находится в наборе `comp`, потому что это просто часть `gcc`. Получается, набор `xbase` требует установленного набора `comp`.

Есть два типа людей, которых это раздражает: люди с маленькими дисками, и люди с фобией "компилятор на сервере? непостижимо!" (хотя эти люди правы: <http://www.welivesecurity.com/2014/03/18/operation-windigo-the-vivisection-of-a-large-linux-server-side-credential-stealing-malware-campaign/>)

Так что теперь `auxcpp` стал частью набора `base`. Прощай, зависимость `xbase` от `comp`. Текущее состояние наборов X Window сохранится и в 5.6. Помимо этого, я обновил несколько компонентов `xenocara`. Репозиторий `xenocara` практически готов для 5.6.

Но всё равно, мне понравился хакафон. Спасибо Мите и его команде за организацию, и всем благодетелям за пожертвования!

5 Марк Эспи (espie@) о портах и пакетах

Ещё один отчёт с завершившегося недавно хакатона g2k14, от Марка Эспи:

В Словении я был в первый раз. За несколько часов - к счастью, удалось избежать гроз - осмотрел столицу. Очень интересное соединение: никогда не видел подобной смеси из восточной Европы, южной Европы и туристических мест.

Что до самого хакафона, я прибыл на него вскоре после крупного изменения (переупорядоченные пакеты), и был практически готов исправлять проблемы в случае необходимости. К моему удивлению, всё работало как часы... Если я что-то и сломал, то никто этого не заметил; зато всем должно понравиться ускорение процесса обновления пакетов.

После продолжительного подпнывания, Вадим Жуков таки закоммитил `digikam-kde4` (порт `Digikam` для KDE4, - В.Ж.). Я провёл креш-сборку и информировал его о найденных проблемах, которые он быстро исправил.

Я работал над немногочисленными мелочами... столкнулся же со многими, и исправил чуть меньше после обычного слома дерева портов, связанного с нашими бесстрашными ломателями исходников (в основном негативные последствия были из-за `libressl`, `endian.h` и обновления `mesa`).

Я работал над новым механизмом, обеспечивающим лучшую целостность репозитория пакетов. Пакет "quirks" теперь сообщает о дате своего подписывания (которая в свою очередь проверяется, поэтому подделать её не получится), благодаря чему теперь можно знать, что срез пакетов достаточно свеж, или же что кто-то помешал ему попасть на ваше любимое зеркало...

...а ещё пакет "quirks" содержит в себе список уязвимых версий пакетов, благодаря чему вы получите сигнал опасности, если вам требуется обновиться из-за уязвимости в старой версии и при этом новой версии пакета на зеркале нет.

Всё это лишь уведомляет пользователя, так как срезы пакетов требуют определённого времени для расплзания по зеркалам... У нас есть идеи, как побороть ЭТУ проблему, но после обсуждения с вовлечёнными сторонами было принято решение отложить внедрение до следующего релиза в связи с необходимостью чересчур серьёзных переделок.

Я также пытался решить проблему с необходимостью наличия исходных текстовой базовой ОС для сборки пакета "pkglocatedb". К моему большому удивлению, Тео согласился, и мы зашли даже дальше, чем изначально планировалось, благодаря чему снапшоты теперь будут включать `locate`-базы как для базовой системы, так и для иксов.

Я провёл немало времени, играя с этими базами: теперь и `pkg_check` использует их, позволяя проверить систему полностью. По-прежнему слишком много лишних сообщений, но прогресс налицо.

Также я провёл немало времени за вычисткой устаревшего кода. Оно не так привлекательно, но, является, пожалуй, важнейшей частью процесса, так как при этом достигается уверенность в том, что мы не запускаем более не нужный и не поддерживаемый код...

Ещё я должен упомянуть о отдельном `cpr` (препроцессоре C - прим. ред.) для `calendar` и `xrdb`, теперь иксы не потребуют для своей работы установки базового набора `comp`.

Как обычно, встречаться лицом к лицу собратьев по разработке очень помогло некоторым проектам продвинуться вперёд.

Спасибо OpenBSD Foundation за спонсирование этого мероприятия, а также Мите за место, всё было организовано настолько хорошо, что нам даже не приходилось о чём-то задумываться.

6 Мартин Пеликан (pelikan@) об ext4 и о файловых системах

Мартин Пеликан пишет в отчёте с g2k14:

Мой первоначальный план состоял в том, чтобы в нашей `base` мог собираться `libcpr` из LLVM, дав нам поддержку C++11. После того, как я читал о последних дополнениях локалей POSIX, другие разработчики прояснили, что будет нужно больше вариантов версии библиотеки, чтобы не сломать порты. После того, как первый `diff` был готов, я поднял сборку базовой системы, чтобы проверить, сломается ли она. **И затем моя жизнь изменилась...**

За несколько дней до хакатона я решил поставить на свой ноутбук Linux, Windows и OpenBSD рядом друг за другом. Одна из связанных с локалями статей была оставлена на разделе с Linux, и я хотел открыть её в Австрии, которая просто полна тоннелей без интернета. Наше ядро не любит ext4; будучи слишком ленивым для перезагрузки, я решил "пришло то самое время", чтобы выяснить, почему.

Неудивительно. ext4 использует extents, которые в былые времена не поддерживались. Беглый взгляд на FreeBSD показал, что у них уже есть read-only поддержка, которая стала более-менее функциональной на моём ядре OpenBSD в среду вечером. Нет индексов каталога NTTree, нет 64-битных номеров блоков, нет журналов или снапшотов, или защиты от мульти-монтирования. Но я мог наконец прочитать PDF без перезагрузки и затем даже скопировать файл, больше, чем 4 ГБ, или открыть каталог с 50000 подкаталогов в нём. Никогда прежде не видя исходников файловой системы, я смог сделать рабочий порт за несколько часов? **Жизнь прекрасна!**

Теперь вопрос состоял в том, как это интегрировать. Разработчики OpenBSD не любят гигантские `diff`ы, и на это есть серьёзное основание. После починки формата `inode` и добавления новых флагов, Тед указал на древнее правило "кто последний полез в эту часть - тот теперь её мейнтейнер". Было очевидно, что я должен был получить больше знаний, читая тексты дизайна и код других систем, прежде чем смогу делать ценные и правильные коммиты. Устаревшие остатки, похожие на трудно (и неправильно) закодированный лимит размеров файлов были первыми. Части кода понимались с трудом, но FreeBSD удалось их разделить, так или иначе. После этого наши пути кода выглядели достаточно похожими, **и поддержка ext4 ворвалась в нашу жизнь.**

Несмотря на то, что цель хакафона была в написании кода, я должен был буквально учиться работать с подсистемой, которую я видел до этого только однажды и мельком (с FUSE). Поскольку всё это уже было во FreeBSD, и заставить это работать было так просто, это было очень хорошее место, чтобы научиться писать код файловой системы самому. В течение следующих дней я начал с того, что написал парсинг и чтение журнала, а закончил тем, что сидел и сознательно ломал мою файловую систему, и смотрел, как она будет восстанавливаться и что для этого нужно ещё сделать. В общем, это верный путь, и цель - сделать поддержку записи.

Это было бы невозможно без Филипа Гуентэра, который рассмотрел мои diff-ы после того, как я продолжал отвлекать его от более важных дел. Тед, Тео, Кен и Боб дали мне много ценного и объяснили, как вещи должны работать. Стефан Сперлинг дал мне доступ к одной из его машин `sparc64` и с удовольствием держал дерево актуальным, таким образом, мои поломки были относительно небольшими и незначительными. Обсуждения о сетевом стеке помогли сузить наш фокус в направлении и по поводу других решений. Митя занимался организацией встречи, чтобы дела шли так, как им и должно идти. Каждый всё сделал отлично, спасибо!

Будем надеяться, что этот энтузиазм насчёт файловой системы сохранится, потому что теперь я должен переключиться назад на *GSoC (Google Summer of Code)* - в задачи, которые мне знакомы. Надеюсь, что файловая система, сломанная у меня сейчас, когда-нибудь поможет починить вашу...

Замечание

эти слова оказались пророческими, и после этого мы с Мартином вели обширную переписку о поломках. Р.Я.

7 Jonathan Gray об улучшении драйверов для X

Джонатан Грэй (jsg@) сообщил нам, почему он провёл 30 часов в автобусе, чтобы быть с нами:

Одной из первых вещей, которые я сделал в g2k14, был импорт обновления Mesa, над которым я теперь работал в течении некоторого времени. Я следил за git репозиторием Mesa несколько месяцев и отправлял патчи, чтобы уменьшить всю ту боль, причинённую локальным diffом, который не был таким большим, но приходилось тратить много времени на обновления.

Незадолго до хакафона я столкнулся с проблемой, заставляя Mesa собираться на i386, как бы то ни было. Это происходит только в том куске кода, который с помощью `sysctl` проверяет, включен ли SSE. Это, как оказалось, было проблемой, потому что `sysctl.h` включает в себя `utm_extern.h`, который, в свою очередь, берёт заголовочные файлы ядра, включая `mutex.h`, это означает, что `mtx_init()` из Mesa конфликтует с `mtx_init()` ядра. Тео потратил немного времени, вычищая `sysctl` и заголовки `utm`, таким образом, они не будут включать где-либо много определений. Эту работу уже закоммитили, когда я пришёл на хакафон.

На следующий день я сделал немного сборок хепосара, чтобы найти любые дополнительные проблемы. Проблема, которую я нашёл, происходила из-за симлинка в файл дистрибутива Mesa, который игнорировался `cvns import`, что починили ссылками из Make-файлов в другую директорию. Я также проверил дважды работоспособность сборок Mesa со включенным LLVM, который всё ещё работал через программный рендеринг LLVMpipe.

Другая проблема со сборками Mesa заключалась в том, что `sys.mk`, автоматически включаемый через `make` в `Makefile`, добавляет `CFLAGS` к `CXXFLAGS`. Поскольку Mesa является смесью C, которая включает и код C99, с C, я ругался на то, что к нему попадал C-специфичный флаг `-std=c99`. diff, который исправляет это в системных Make-файлах и некоторых других местах, будет потом отправлен по почте.

Я также проконтролировал, чтобы дерево исходников собиралось с `OPENSSL_NO_DEPRECATED`, который в большинстве случаев добавлял инклюды, которые больше автоматически не брались из других инклюдов. Для некоторых вещей, таких как `nginx`, которые поддерживаются извне, есть патчи, которые уже доступны в следующих версиях. Мы их потом возьмём, но пока что ещё не так уж и стоит патчить нашу версию, когда есть другие места в дереве (`libkeynote/bind/sendmail` и. т.д.), где требуется сделать изменения. Я также вскользь посмотрел на компиляцию с `OPENSSL_NO` но после того, как увидел, что `dc` и `gzsigs` поломались во время сборки, я решил посмотреть в другие места.

Я посмотрел на обновление некоторых патчей `clang`, которые искал пару лет, и коммитил некоторые вещи, касающиеся этого.

Xorg теперь может работать без необходимости предоставлять прямой доступ из пространства пользователя к памяти ядра/устройства, если режим `modesetting (KMS)` ядра поддерживается. Проблема ещё заставляет некоторые устройства требовать доступ к этому окну памяти, чтобы запустился Xorg. Установщик задаёт вопрос, если находит `vga` устройство, которое включает окно через `machdep.allowaperture` в `sysctl`. После обсуждения с парой людей на g2k14 я написал небольшие скрипты, чтобы забирать номера поставщика/продукта PCI из драйверов `radeonrm` и `inteldrm`, которые используются `pci` вложением в `vga` драйвер, чтобы написать строчку в `dmesg`, если это окно памяти требуется для запуска Xorg. Установщик был изменён `halex@` и `gre@`, чтобы проверить эту строку и теперь будет только спрашивать, нужно ли человеку запускать X11 (который включает окно), если оно найдено. Вопрос X11 не будет задан теперь на многих серверах, так как есть чёрный список серверных графических устройств в коде, решающем, нужен ли `aperture`.

Проблема, с которой я столкнулся теперь несколько раз, - это недостаток заголовка `cpuid.h`, который идёт из `gcc >= 4.3` и `clang`, чтобы обеспечить интерфейс, запрашивающий `cpuid` на `i386` и `amd64`. Mesa из `git` теперь требует `cpuid.h` для сборки. Intel-овский драйвер Xorg отключает код, включённый в решение, если SSE присутствует, и делает решения, основанные на размерах кэша, если его нет. И, по крайней мере, некоторые порты (т.е. `OpenXCOM`), кажется, теперь его ждут. Таким образом, я взял `cpuid.h` из `clang`, чтобы включить его в нашу версию `GCC 4.2.1`. Сначала я изменил определения `SSE_4_1` и `SSE_4_2` на `SSE_41` и `SSE42`, чтобы соответствовать именам, используемым в `GCC`, но, вероятно, они оба будут включены, когда это закоммитят.

Большое спасибо OpenBSD Foundation и Мите за g2k14!

8 Об этом документе

Перевод дополнительными лицензионными ограничениями не ограничивается, и равен лицензии оригинального текста. Какая лицензия на оригинальные интервью - понятия не имею, скорее всего - общественное достояние. Тогда и этот текст можно использовать, как вашей душевненьке угодно: копировать, удалять, ругаться на него, заполнять им пустые места на веб-сайте, петь его хором. Нет, петь хором, пожалуй, нельзя, и пусть это будет единственное ограничение на распространение данного текста.

Авторы перевода: Роман Яковлев, Вадим Жуков, Виктор Феденев