

# так ли открыты открытые исходники?

крис касперски ака мышцх

**затхлый воздух давно непрветриваемых корпоративных секретов уже ощущается и в россии. открытые исходники встречаются с большим энтузиазмом, словно глоток свежего воздуха. они активно продвигаются и позиционируются как идеальное средство от всех проблем. попробуем разобраться насколько это так...**

## введение

Начнем с повторения заезженных истин. Компьютер состоит из программного и аппаратного обеспечения, граница между которыми настолько размыта, что вряд ли кто-то рискнет ее провести. Любой из компонентов современного ПК сам по себе нехилый компьютер. Вот, например, жесткий диск. В нем есть сигнальный процессор и управляющий микроконтроллер, работающий под управлением специализированной операционной системы и несущий на своем борту СОМ-порт для передачи технологических команд. При большом желании, на него можно заливать собственные программы, например, выполняющие некоторые математические расчеты и возвращающие результат. Такие "компьютеры" даже можно объединить в сеть, собрав мини-кластер. А что! Чем он хуже других?!

Прошивки винчестеров — это программы, доступные для дизассемблирования, модификации и изучения. При всем нежелании производителей разглашать анатомические подробности своих девайсов, на абсолютную закрытость информации жаловаться все же не приходится.

Некоторые микроконтроллеры имеют внутреннее ПЗУ, защищенное от дампа, и прочитав его можно только на специальном (и притом весьма дорогостоящем!) оборудовании, да и то не всегда. Вот хорошая страница по хардварному хаку: <http://www.cl.cam.ac.uk/~sps32/> с кучей познавательных статей и фотографий. А вот несколько эффективных идей взлома: большинство производителей так или иначе распространяет прошивку в том или ином виде, а, значит, ее можно скачать и хакнуть. Даже если распространяются лишь обновленные фрагменты, в которых нет ничего интересного, можно внедрить "свою" подпрограмму, которая, попав на микроконтроллер считает всю прошивку и сдампит ее.

Короче, даже такое экзотическое программное обеспечение нельзя считать полностью закрытым. Что же тогда говорить про обыкновенный софт. Отсутствие исходников еще никогда и никого не останавливало. Если нужно узнать как программа работает — берем в руки дизассемблер и вперед!

Было бы нелепо сводить концепцию Open Source к одному лишь факту доступности исходных текстов. Здесь все намного глубже и... интереснее.

## о тайнах и секретах

Существует понятие "тайны", например, государственной или коммерческой. Это вполне нормальное явление (особенно в информационную эпоху). Античные мастера обладали множеством продвинутых технологий, передающихся из поколения в поколение или умирающих вместе с их обладателем. Многие секреты оказались безвозвратно утеряны во времени. С развитием индустрии, такой путь "производства" оказался весьма проблематичным и потому государство предложило концепцию патента. Внимание! Сейчас начинается самое интересное!

Что же такое патент? Патент это добровольное разглашение тайны в обмен на охрану исключительных прав использования данной технологии. В практическом плане это означает, что описание любой запатентованной технологии (а на западе сейчас патентуется каждая мелочь) можно свободно (и притом бесплатно!) найти на сайте Patent Full-Text and Full-Page Image Databases (<http://www.uspto.gov/patft/index.html>). Зачастую, это единственная информация, доступная по современным технологиям (устройству тех же жестких дисков, например).

Патенты стимулируют систему образования и вообще полезны во всех отношениях за исключением полной несвободы их использования. Да, мы можем узнать как устроен транслятор адресов винчестеров типа Maxtor (что очень полезно для их восстановления), но не можем выпускать ничего подобного самостоятельно. Владелец патента вправе решать: выдавать нам лицензию или нет. Сначала патенты выдавались на вполне конкретные изобретения, которые могли использоваться в производстве. Например, на хитроумную

конструкцию часов с самозаводящимся механизмом. Изучив чужой патент, остальные пытались усовершенствовать его, разработав совсем другую конструкцию, чтобы выпускать ее без всяких там отчислений, что было вполне логично. Но в последнее время наметилась неприятная тенденция к патентованию фундаментальных основ на которых держится пуп мироздания. Несмотря на то, что информация по-прежнему остается открытой, она уже не может быть использована посторонними.

А это значит, что программисты не могут писать программы, не опасаясь судебного преследования. Сегодня куда ни плюнь, обязательно попадешь во что-то запатентованное. Написать программу сложнее "hello, world!", не нарушив при этом пары десятков патентов — невозможно! Точнее возможно, но только тихо. Без лишнего шума. Пока мы не будем никому мешать, нас никто не тронет. Скорее всего, никто не тронет. Если программа попадет на глаза ретивому менеджеру крупной компании, он разорвет нас в пух и прах. Куча прецедентов — яркое тому подтверждение.

Со временем ситуация будет только ухудшаться. Затем, когда она дойдет до логического абсурда, угробив три четверти индустрии, случится что-то радикально. Такое, что все изменит. Но до этого еще далеко. На сегодняшний день от патентов страдают все. Компании, вкладывающие миллиарды долларов в разработку перспективных исследований, каждый раз рискуют потерять все, потому что конкурент может добежать до финиша на день раньше и сорвать банк. Мелкие компании находятся еще в более проигрышном положении, во-первых, они вынуждены лицензировать кучу технологий, которые могли бы разработать и самостоятельно, но... увы! нельзя! на них, и всю прилегающую к ним смежную область, уже выданы патенты! Во-вторых, запатентовав что-то свое, мелкая компания вынуждена сидеть и ждать у моря погоды, пока пираты сменят гнев на милость и заплатят ну хоть что-то (а ведь регистрация патента немалых денег стоит к тому же необходимо платить постоянные отчисления за его "защиту"). В общем, мрак.

Тем не менее, у патентов и Open Source много общего. И в том, и в другом случае информация хранится в открытом виде, доступном (для ознакомления) всем желающим. А на ее использование наложены достаточно жесткие ограничения. Да! да! Open Source это отнюдь не свобода! Это сплошные ограничения, с которыми настала пора познакомимся поближе.

## **свободное несвободное Open Source**

Начнем опять с тривиальных истин и определений. Программой называется последовательность инструкций для управления процессором. Не стоит путать ее с программным продуктом, представляющим совокупность программы, документации, упаковки, технической поддержки и т. д. То есть программа — это просто набор байт, а программный продукт — услуга.

По нашим (и не только нашим) законам, программа является объектом авторского права, поэтому, на ее использование наложены определенные ограничения, возникающие в силу объективных психических расстройств в голове у законников. К слову сказать, авторское право уже распространяется даже на картриджи к принтерам, так что удивляться ничему не приходится. Власть захватили в руки медиа-магнаты и софтверные гиганты, проталкивающие свои законы и занимающие первые строчки хит-парада самых богатых людей планеты, опередив при этом даже нефтяных магнатов. Вообще-то, по логике, законы должны служить на благо подавляющего большинства, а не жалкой кучки "имущих", но... имущие имеют нас и ставят раком, да еще при этом показывают рекламу, убеждающую нас, что все идет верным путем и вообще забота о состоянии Била Гейтса нас гражданский долг. Как будто Microsoft ведет честную игру и честный бизнес! Так почему же мы должны поступать иначе?!

Кстати говоря, половина опрошенных музыкантов/авторов равнодушно относится к тому, что их произведение можно стыбнуть в сети, другая половина даже благословляет это, поскольку свободное копирование увеличивает их известность. И лишь жалкие несколько процентов вроде Войновича вопят, что Интернет это палач писателя. Я Войновича не читал, а теперь уже и не прочту. Если он пишет только ради личного обогащения, то читать там нечего. Продукт брожения, продукт перегонки, творческий продукт...

Как человек, неоднократно издававшийся зарубежом, скажу, что американский книжный рынок (да и рынок медиа-продукции) переживает не лучшие дни. Книги практически не продаются. А все потому, что народ не хочет покупать по старому, а "маркетодиды" не умеют/не хотят торговать по новому. Издатель (или в случае музыкантов "лейб") покупает права на произведение и кладет его под сукно. Типа собака на сене — сам не продаю, не раскручиваю, но и другим не даю. Как следствие, все больше и больше творческих людей

посылают авторское право на хрен и свободно выкладывают себя в сеть, что приносит намного больший доход (не говоря уже о глубоком моральном удовлетворении), чем обращение к издателям, которые есть крысы по определению.

Программное обеспечение еще находится в той стадии, когда все понимают, что авторское право несет только вред, но еще никто реально не готов от него отказаться. Даже сторонники Open Source. Фактически, с точки зрения авторского права нет совершенно никакой разницы в каком виде распространяется программа — в виде двоичного файла или исходного текста. В любом случае оно обеспечивает охрану, что является очень большой ошибкой законников. Вспомним ситуацию с патентами — обмен тайны на охрану. А здесь что? Вот если бы авторское право защищало только исходные тексты, большинство программ распространялись бы именно так, и общество от этого только выиграло бы!

Впрочем, не будем предаваться мечтаниям, а лучше разберемся с тем, как обстоят дела на практике. Итак, программа — объект авторского права и ее распространение полностью регулируется правообладателем (во всяком случае в теории все обстоит именно так). Нормальные авторы выпускают произведения в свет, позволяя им жить собственной жизнью, но таких очень немного. Подавляющее большинство хочет все, причем сразу. Движение Open Source не стало исключением...

Что такое свобода? Допустим я создаю программу и выкладываю ее в Интернет со словами: пусть каждый использует мое творение так, как считает нужным, вознаграждение и сохранение строки с инициалами автора — необязательны. Вот это, действительно, *свобода*, при которой автор программы не имеет никаких преимуществ перед всеми остальными. Open Source проповедует совсем другую идеологию. Лицензия GPL, под которой распространяется подавляющее большинство открытых продуктов, разрешать использовать компоненты программы в своих продуктах, но при этом требует, чтобы они распространялись по все той же лицензии GPL! А это значит, что: а) всем пользователям необходимо передать безвозмездное право на модификацию и дальнейшее распространение программы; б) предоставить доступ к исходным текстам.

Что ж, это логично. Если я не сплю ночами и под бурчание возмущенного желудка пишу свободно распространяемую программу, за которую не требую вознаграждения, скромно прося милостыню подаяния (или по-английски donate), то, разумеется, я не хочу, чтобы какой-то там билл гейтс включил мое творение в свою галимую windows и делал на язве моего желудка money! Но ведь в мире существуют не только гейтсы! Для большинства программистов лицензия GPL равносильна приговору и с гораздо большей охотой они приобретают коммерческие "несвободные" компоненты, поскольку они более "свободы"!

Соблазнившись лицензией GPL и написав какой-нибудь продукт для себя, программист становится заложником "свободного" Open Source и вынужден распространять его на "свободных" основаниях. Формально GPL не запрещает продавать программу, но при этом накладывает столько ограничений, что делает успешный бизнес практически невозможным. Для обхода (или от освобождения) от GPL разработчик вынужден переписывать тучу кода, поэтому GPL лицензирую еще называют GPL-вирусом, который заражает все, к чему прикоснется.

Оставив нравственно-политический подтекст в стороне, попробуем ответить на вопрос: что же такое свобода? Open Source образует свое сообщество (community), а любое сообщество это уже несвобода. Вступая в любое, повторяюсь, в любое (co)общество вы поступаетесь частью своих прав и свобод в обмен за гарантию защиты оставшихся. Лицензия GPL защищает разработчиков, препятствуя использованию использования продуктов их труда на традиционной коммерческой основе. Этим GPL создает принципиально иную нишу рынка, успешно конкурирующую даже с такими монстрами как Microsoft и IBM. Использование программ, распространяемых по лицензии GPL, как бы втягивает разработчика внутрь сообщества, после чего он уже вынужден работать на его благо. Конечно, имеются и другие лицензии, более демократичные, например, лицензия BSD, требующая всего лишь обеспечить открытость кода, но не обязывающая передавать пользователям все права. Однако, они не получили такого широкого распространения.

Короче, подытожим. С точки зрения конечных пользователей, GPL это действительно свобода. Можно бесплатно скачивать продукты из сети, дорабатывать их по своему усмотрению, записывать на CD-R, и продавать по всему периметру сети магазинов, ведь GPL не препятствует продажам! Главное, чтобы были исходные коды плюс право их дальнейшей модификации.

А вот программисты в GPL никакой свободы не видят. Если это и свобода, то только внутри сообщества, "свобода" только для избранных, вроде нашего коммунизма.

## **закрытая открытость Open Source**

Какие мотивы могут побудить создателя программы к распространению исходных текстов? Какие мотивы могут этому воспрепятствовать? Решение о "закрытости" или "открытости" как правило применяется на бессознательном уровне. Логика здесь отдыхает.

Часто приходится слышать, что исходные тексты содержат корпоративные секреты, которые ни в коем случае нельзя разглашать, иначе компания просто рухнет. Что за ерунда?! Как уже говорилось выше, в современном мире секреты уступили место патентам. Если некоторая фирма создала действительно революционный продукт (например, безупречный распознаватель речи) и выбросила его на рынок, машинный код тут же дизассемблируют конкуренты и восстановят алгоритм. Никто даже пункнуть не успеет! Помешать им может только патент, а патент требует раскрытия секретности. В России и Европе, правда, алгоритмы еще не патентуются... Ну, как бы не патентуются, но это ограничение легко обойти, например, описав абстрактное устройство с процессором, выполняющее такие-то и такие-то действия. Устройство уже патентовать можно. И ведь патентуют! Причем в промышленных масштабах!

Еще существует мнение, что наличие исходных текстов позволит создать аналогичный (или даже сильно улучшенный) продукт на основе уже существующего. Отчасти это действительно верно, однако, только отчасти. Взгляните на Open Source. Исходные тексты доступны, но... продукты не плодятся как кролики. То есть, плодятся конечно, но побочные ветви быстро сдыхают. А если права модифицировать исходники нет, то никому и в голову не придет что-то там дорабатывать, ведь не сегодня-завтра придут судебные исполнители и зарубят проект к чертовой матери. Лучше быстро выпустить "пиратку" по демпинговой цене и уползти с наваром. Для это даже исходные тексты иметь не нужно!

Единственная объективная причина — это кривизна исходников, демонстрирующая тупость их создателей, умышленно оставленные закладки и ворованные компоненты, плюс еще стадный признак. Если никто не показывает исходных текстов, то и мы не будем этого делать. Особенно меня умиляют начинающие программисты, объявляющие, что их продукт "запатентован", а его исходные тексты покрыты мраком тайны. На самом деле, исходные тексты недоступны потому, что программисты боятся показать их народу, и все сразу увидят какие они корявые и неструктурированные. Дураки. Если коллеги укажут тебе на ошибку, то ты сможешь выявить ее и исправить, ликвидировав еще один пробел в своих знаниях. Чтобы научиться писать хорошие и конкурентоспособные программы, необходимо не раз ткнуть носом в собственное дерьмо. Воняет? Конечно воняет. Но более чистых путей увы не бывает.

Если исходные тексты не открывают, то воруют. За примерами ходить далеко не надо. Только ленивый не найдет в сети сырцы MS-DOS 6.x и W2K. Ну и что? Какой от этого ущерб? Аналогичных продуктов на основе уже существующих так и не появилось, корявость рук программистов конечно стала видна (даже несколько дыр новых нашли), но о том, что в Microsoft умных людей практически не осталась и так все знают (достаточно взглянуть на Longhorn, beta-версия которого занимает ~450 Мб памяти), так что ровным счетом никакого ущерба. Сплошной пиар. А вот программистом это помогло лучше понять некоторые туманные места в документации. Теоретически, Microsoft должна быть заинтересована в том, чтобы привлечь как можно больше разработчиков на свою платформу. Практически, она вкладывает миллиарды убитых енотов в рекламу и устраивает разные акции, зазывающие разработчиков, так что распространение исходных текстов идет на пользу как самой Microsoft, так и программистам вместе с пользователями.

А вот другой случай. Несколько лет назад были украдены исходные тексты CISCO IOS. Какой разразился скандал. Руководитель компании драл задницу в поисках похитителя, вылив на него столько грязи, сколько не увидишь даже в российском придорожном сортире. А ведь IOS представляет собой смесь Линуха и BSD. "Свое", "родное" там только драйвера и кое-что еще. Для этого, кстати говоря, даже не нужно заглядывать в исходные тексты. Как и любой цельнодернутый продукт, CISCO наследует все дыры своего "донора". Так что здесь все просто: вор у вора лошадку увел. Какой смысл закрывать исходные тексты, если они и так доступны? Их даже воровать не нужно! Стек протоколов во всяком случае уж точно писан не CISCO! Но это не мешает компании выпускать самые популярные роутеры, ведь роутер это не исходный текст. Это железка. Наличие/отсутствие исходных текстах операционной системы на объем продаж никак не влияет. И это не единичный пример!

Теперь разоблачим миф, связанный с безопасностью. Дескать, имея на руках исходные тексты, можно обнаружит любые баги и закладки или, говоря научным языком, провести аудит. А вот и ни хрена подобного! Типичная современная программа представляет собой миллионы строк сложным образом взаимодействующих с "окружающим миром" и с друг с другом.

Несомненно, исходные тексты упрощают анализ, но не до такой степени, чтобы радикально что бы то ни было изменить. Чтобы разобраться, что в каком файле находится и кто за что отвечает, необходимо угробить кучу времени...

В команде Open Source или в коммерческой фирме существует жесткое разделение. Каждый ведет свою часть проекта и в чужие суется либо от не фиг делать (и за это ему дают по рукам), либо на стыке взаимодействия своей части проекта с остальными. За это ему тоже дают по рукам или ругаются матом. Об этом хорошо сказал Евгений Зуев в статье "Редкая Профессия": *"Возникло тяжелое ощущение того, что это большая темная комната, а у тебя только маломощный фонарик, который в состоянии осветить небольшой аппарат - твои модули. От аппарата тянутся в темноту провода и вереницы зубчатых колес. Что делается в дальних углах, неизвестно. Иногда вокруг раздаются какие-то звуки, из темноты выступают части каких-то движущихся механизмов, назначение которых остается неизвестным, даже если осветить их. Время от времени из темноты раздается голос, настоятельно требующий: "нажми на кнопку с надписью ABC", "переведи рычаг XYZ в правое положение". Что делается в комнате и как все работает вместе, понять совершенно невозможно"*.

В команде разработчиков обязательно имеется несколько человек, своеобразных "носителей знания", которые держат в голове максимум подробностей структуры кода и координируют работу остальных "суб-носителей", окуливающих каждую команду. Все это сделано для того, чтобы обезопасить проект от воли отдельных разработчиков. Если член команды покидает/погибает ряды компании, его место занимает другой, которому "вливают" все старые знания в черепную коробку. В больших фирмах существует развитая инфраструктура документооборота, в Open Source – списки рассылки. Благодаря этому, вновь пришедшие разработчики могут разобраться с вверенной им волостью и навести мосты с ближайшими соседями. Процесс вживления в команду протекает долго и практически всегда очень болезненно.

Сами по себе исходные тексты любой серьезной системы (например, компилятора gcc или ядра linux) полностью лишены смысла. Это только груда файлов, которую, если повезет, с N'ой попытки удастся откомпилировать. Без соответствующей поддержки, разобраться в них практически невозможно. Возьмем тот же gcc. Допустим, мне необходимо добавить в него поддержку новой фишки или исправить ошибку кодогенератора. Поклонники Open Source говорят, что в случае с Microsoft Visual C++ мое дело труба. Все, что я могу — бомбардировать Microsoft факсами и умолять о пощаде. А вот в gcc – просто взял и добавил. Ага, щас! Мне тут программировать нужно, а не ковыряться в исходном коде gcc!!! Да за то время, пока я с ним буду разбираться, можно сто раз найти обходное решение проблемы или дождаться очередного фикса от Microsoft! Так что, чем крупнее проект, тем меньшую пользу можно извлечь из исходных текстов. К тому же, куда мне девать свои изменения при переходе на новую версию? С большой степенью вероятности, перенести их будет очень непросто и потребуются угробить еще одну кучу времени. И так каждый раз.

Интерфейс плагинов в этом смысле намного более привлекателен. Сторонним разработчикам предоставляется более или менее документированный и унифицированный API, дающий им возможность создавать собственные расширения. Заглядывать в исходные тексты при этом не требуется. К тому же снимается проблема переноса расширений во все остальные версии. В Open Source мире ведь как — или сообщество принимает твоё расширение и включает его в поставку (но в любой момент может исключить), либо посылает на хуз, и ты вынужден вести свою версию, что приводит к путанице и всеобщей дисгармонии. А так — конечный пользователь сам решает какой ему плагин ставить, а какой нет.

Еще один миф. О тысячах энтузиастов по всему миру... Это чистейшей воды брехня. Я не знаю ни одного продукта, созданного толпой. Всегда во главе стоит один-два толковых парня, а остальные ходят с понтом, что "мы пахали". Вот, например, DOS Navigator от RIT Labs. Отличный продукт, с нехилым количеством пользователей и фанатов. После того как фирма свернула свои работы и выложила его исходники в сеть, проект сдулся. Да, сейчас существует NDN и еще несколько "отпрысков", но все они находятся в сильно заболоченном состоянии. Конечно, можно привести и контр пример. Скажем Лис (он же FireFox), который развивается настолько бурно, что даже теснит IE, а потеснить IE это не туалет обоссать скажу я вам. Но... как же он тормозит! Вся программа в одном исполняемом файле. Так умные люди не пишут! Так что рая на земле не бывает...

Что там у нас еще? Ах да, бэк-доры. Дескать, в Open Source их не внедришь. Как бы не так! Хорошо продуманную закладку обнаружить практически невозможно. Достаточно "случайно" допустить ошибку переполнения буфера, проявляющуюся только при стечении множества маловероятных обстоятельствах на стыке разных модулей. Ну, вот например, такой

сценарий: один модуль делает удар по памяти, искажая данные другого так, что буфер переполняется на строках в 69 символов, но не переполняется при всех остальных (срабатывают дополнительные проверки). Чтобы обнаружить эту ошибку необходимо удержать в голове работу двух разных модулей, связав их воедино, что не сильно легче чем дизассемблировать машинный код. А переполнение буфера это уже shell. Со всеми вытекающими...

## **ЗАКЛЮЧЕНИЕ**

Мне очень нравятся открытые исходные тексты и я ненавижу несвободное программное обеспечение, которое мне запрещается модифицировать или дизассемблировать. Из двух свобод "свободы социалистического лагеря" и "свободы одноименной статуи" я не колеблясь выбираю первую. Я ненавижу корпоративные секреты и прочую ахинею. Под Windows вообще нельзя программировать! Достаточно вспомнить разбирательство между Microsoft и Stacker. Фирма Stacker написала свой компрессор, а Microsoft его спи... Обиженная Stacker подала в суд и выиграло дело, но Microsoft выдвинула встречный иск. Дескать, Stacker увела у нее... протокол загрузки MS-DOS!!! Обе компании разошлись "полюбовно". Никто никому ничего не должен. (Кстати, "правосудие" это не от "правого судьи", а от "права судить")

Приобретая диск с лицензионной Windows, я не приобретаю даже носителя. Поскольку на нем находятся логотипы Microsoft и прочая X, на его использование наложено множество ограничений. Что же касается самого программного обеспечения, то лицензия делает его использование практически невозможным (речь, конечно, идет в первую очередь про написание программ). Тем не менее, Windows (и большинство коммерческого программного обеспечения) как бы работает. Производители действительно прилагают колоссальные усилия, чтобы я остался доволен и не вернул диск в магазин.

Линух и большинство остальных Open Source проектов работают только с бубном и только рядом с тем, что это лабал. Только не надо кричать о кривых руках и нести прочий бред. Качество кода в том же Линухе в разы ниже, чем в Windows и прежде, чем с ним будет можно работать, придется долго драчить напильник, дорабатывая его под свои нужды. Зато потом... потом будет кайф и ништяк.

Вердикт: в России в условиях не соблюдающихся законов, закрытое программное обеспечение оказывается в значительном выигрыше, поскольку, его покупают (то есть рiЗдят...), а не лицензируют. Один и тот же диск ставят на столько машин, на сколько это только возможно, дизассемблируют машинный код, вносят в него любые мыслимые и немыслимые изменения. Короче, делают, что хотят. Open Source все еще остаются в загоне. Действительно, зачем ездить на Запорожце, когда за эту же сумму можно приобрести Мерседес?!

Неверно думать, что нарушение лицензии идет нам на пользу, а Microsoft не в убыток. Народ освоил Windows и Word, программисты создали целую инфраструктуру. Теперь, когда в Microsoft "спохватились" и "вспомнили" про пиратство, что-то менять стало поздно. Намного дешевле купить лицензионную ось, чем переносить все на Линух. Но! Чем жестче будет политика лицензирования, тем более привлекательным окажется Линух! Лично я, увидев Windows Longhorn, твердо сказал себе: Windows 2000 будет моей последней осью из семейства оконных и если я буду куда-то переходить (а переходить как ни крути все равно придется), это будет либо BSD, либо Линух. У этих систем есть будущее. У программистов, сидящих под Windows, будущего нет. Это сейчас вы молодые и кажется, что все круто и хорошо. Но задумайтесь, что будет когда вы состаритесь? Программы, написанные для Линуха, создаются так же, как и тридцать лет назад и на них имеется небольшой, но устойчивый спрос, а под Windows постоянно приходится осваивать кучу никому не нужных технологий, меняющихся с каждым днем. Сейчас, это легко. Но через десять-двадцать лет смертельно надоест и захочется совершенствоваться в чем-то одном, а чувствовать себя постоянно начинающим...