Соотношение свободы и приватности с безопасностью

Арс Либрев

Влияние свободы программного обеспечения на безопасность

Свободное программное обеспечение дает пользователю возможность его использовать, изучать, изменять, копировать и распространять. Такая возможность имеет место за счет определенной лицензии. Иными словами, свободные лицензии дают пользователю возможность контролировать используемые им программы.[1] Такому подходу противостоит практика создания проприетарного программного обеспечения, которое оставляет контроль над ним разработчику. Поскольку пользователь не может проверить, что именно делает программа и исправить ее, это дает возможность разработчику внедрить в него вредоносные функции для эксплуатации пользователя, такие как слежка и лазейки для удаленного контроля. Поэтому, когда вопрос касается информационной безопасности, ориентир на свободное ПО является необходимым.[2]

Существует, впрочем, возражение, согласно которому, использование свободного ПО, само по себе, вовсе не прибавляет безопасности. В свою очередь, отсутствие доступа к исходному коду, не говорит о том, что программное обеспечение обязательно содержит проблемный функционал. Таким образом свободная лицензия не является поводом счесть программу под ней более безопасной, чем ту, что распространяется под проприетарной лицензией. Насколько обоснованна подобная позиция? Чтобы ответить на этот вопрос необходимо всесторонне рассмотреть те следствия, которые вытекают из использования как свободных, так и проприетарных лицензий, а также проанализировать конкретные факты, связанные с безопасностью свободного программного обеспечения и проприетарного.

Особенности программного обеспечения

Одним из аргументов апологетов проприетарного ПО является утверждение, что открытость кода сама по себе еще не обеспечивает независимую проверку. Указывается, что ввиду объемности и сложности кода, часто никакая проверка не проводится, а если и проводится, то не гарантирует выявления проблем. В качестве примера приводят случаи, когда в свободных программах обнаруживали опасные уязвимости, в том числе позволяющие

злоумышленнику получать полный контроль над системой, которые присутствовали в них на протяжении нескольких лет. И за столь долгий период никто их не замечал.[3]

В качестве крайнего случая, также приводится пример того, как одна из исследовательских групп Миннесотского университета, проводила эксперимент, в ходе которого создавала патчи для ядра Linux с бэкдорами — программными лазейками. Похожие патчи от них были приняты в разных проектах, разрабатывающих свободные программы — далеко не все замечали коварные лазейки. Когда университет признался, что проводил исследование того, возможно ли внедрить бэкдор в свободное ПО и создавал свои патчи с лазейками именно для этого, включение их программных решений было отменено.[4]

Таким образом, сама по себе открытость, не является гарантией отсутствия уязвимостей и лазеек. Но лучше ли с ними обстоят дела у проприетарного ПО? В системах Windows и MacOS неоднократно находили уязвимости, которые также могли привести к полной компрометации системы и также сидели в них годами.[5] Что касается лазеек, то их коварство в том, что крайне сложно доказать их намеренное введение — разработчик всегда может заявить, что это лишь незамеченная уязвимость. Тем не менее иногда разработчики добавляют функционал, для осуществления своего контроля над системой. Например в шифровании диска Windows имелась лазейка, оставляющая Microsoft ключ.[6] В системе Android есть функционал, позволяющий Google менять настройки пользователя.[7] В iPhone есть функционал, позволяющий Apple удаленно отключать приложения пользователей.[8] Это только несколько примеров — подобные особенности неоднократно обнаруживались во многих проприетарных программах.[9] И даже это явно не все, что можно обнаружить в таких программах.[10]

Нелишним будет напомнить, что слежка в проприетарных системах, это установленный факт, который их разработчики не скрывают — прямо расписывая в пользовательских соглашениях то, какие данные о пользователях собираются.[11] Составляются эти соглашения не редко так, что невозможно сказать, ограничивается ли сбор данных только тем, что указано конкретно, точно также как и ограничиваются ли используемые методы слежки только теми, что описаны. Многие корпорации используют единые политики конфиденциальности для своих разработок, а потому невозможно сказать, в каких именно программных решениях, что применяется, и насколько каждая отдельная программа или сервис реализует те или иные гнусные по отношению к пользователю практики. Само расписывание подобного функционала в этих соглашениях делается для того, чтобы снять с себя ответственность — ведь

пользователи фактически сами дают согласие вести за ними слежку и осуществлять удаленный контроль.

Иногда заявляется, что сбор данных в таких системах можно отключить. Во-первых, это сильное преувеличение. Далеко не все проприетарные программы предоставляют возможность отключения слежки. Во-вторых, даже если предположить, что отключение действительно будет полным — следующее обновление может это «исправить» и включить все обратно.[12] Кроме того, никакие методы отключения не дают гарантий, что не осталось функционала, не затронутого этим отключением. Могут возразить, что в свободных системах тоже нет таких гарантий. Но в первом случае разработчик не дает пользователю возможность проверять и изменять программу, во втором дает. Тут мы подошли к вопросу, а насколько реализуема эта возможность? Ведь проверка кода дело не простое, и как уже говорилось выше, нередко эта возможность, несмотря на ее наличие, не позволяет быстро выявить проблему. В свою очередь, апологеты проприетарного ПО указывают на то, что проверять такое ПО тоже можно и, якобы, не менее эффективно, чем свободное. Так ли это? Чтобы это узнать необходимо рассмотреть какие существуют методы проверки проприетарного ПО.

Возможности изучения программного обеспечения

Один из методов проверки ПО, это внешний анализ. Например того трафика, который идет из программы.[13] В этом плане проприетарное ПО неотличимо от свободного. И то и другое можно одинаково эффективно проверить подобным методом и выявить сливает ли оно какую-то информацию через сеть. Полезно сравнить результаты таких проверок популярных проприетарных программ и свободных. Например, проприетарные браузеры Google Chrome,[14] Opera,[15] Yandex[16] сливают информацию огромнейшему количеству ресурсов. Популярный свободный браузер Firefox также сливает информацию.[17] Однако в нем отключить всю слежку значительно проще за счет доступа к конфигурации.[18] В проприетарных такой возможности просто нет. Если же посмотреть на менее популярный, но также свободный LibreWolf, то слежки там значительно меньше, и опять же присутствует возможность ее полного отключения.[19] В иных свободных браузерах слежки нет вообще.[20] Подобным же образом можно сравнить и другие программы, как правило картина одинакова — среди свободных есть те, что не отправляют никакой информации, среди проприетарных найти таковые крайне сложно.[21] Таким образом, свободные программы выглядят явно более предпочтительными.

Еще существует анализ системных запросов, а также данных состояния системы при работе и сбое. Он позволяет выявлять как та или иная программа

взаимодействует с ядром, к каким компонентам системы обращается, что позволяет делать выводы о ее функционале.[22] Разумеется, ничто не мешает проверять этим методом и свободное ПО. Иногда результаты таких проверок показывают обращение проприетарных программ к критическим системным файлам и попытки считать из них данные.

Иногда также в качестве способа проверки отмечают специализированные сборки, которые поставляются с файлами для анализа поведения приложений.[23] Эти файлы с программами работают сами по себе и автоматически выявляют ошибки при работе, сторонние запросы, неинициализированные чтения пакетов и т.д. Для того, чтобы использовать их не нужен исходный код. Однако он нужен, чтобы написать такую программу. Учитывая это, использование данного способа для анализа проприетарных программ вряд ли можно приравнять к независимой проверке, даже если проверка сама по себе возможна, благодаря предоставлению разработчиком таких сборок, что имеет место далеко не всегда.

Также есть метод позволяющий добиться сбоя программы, путем подачи в нее случайных или неправильных данных.[24] Данный метод действительно позволяет выявлять уязвимости гораздо эффективнее, чем непосредственное чтение исходного кода. Но даже несмотря на то, что его возможно производить и в отношении несвободных программ, при доступности исходного кода такое проведение гораздо эффективнее. Разработчики проприетарного ПО редко выпускают сборки, дружественные к этому методу. Фильтровать результаты гораздо труднее без глубокого знания дизайна программы.[25]

Существует также метод обратной разработки. Он состоит в том, чтобы анализировать бинарный код. Это позволяет реконструировать исходный код из бинарной последовательности.[26] Этот метод сам по себе крайне сложен, а кроме того, его применение никак не устраняет проблему объемности и сложности самого кода, которая свойственна и свободному ПО. По этой причине даже если такая разработка проводится, она порой касается только отдельных функций программы, по поводу которых сами разработчики заявляли о закрытии уязвимостей или реализации каких-то методов безопасности. Нужно ли говорить, что это весьма ограниченный метод, завязанный на той информации, которую решили предоставить сами разработчики несвободного ПО. В добавок, его применение возможно только в том случае, если разработчик не вознамерился воспрепятствовать ему, например не применил методы обфускации кода, затрудняющие выявление из последовательности чисел исходники, или не применил технологии DRM.[27] В этих случаях все сильно усложняется, иногда практически до полной невозможности осуществить разработку.

Как видно, даже для некоторых из этих методов проприетарное ПО действительно в значительной мере ограничивает возможности установления действительного функционала, тогда как свободное его облегчает. Кроме того, и само изучение исходного кода, даже не будучи основным способом проверки программ, также позволяет выявлять проблемы и дополняет методы не связанные с чтением кода. Еще кроме самой по себе доступности исходного кода, свободному ПО свойственна прозрачность разработки. А это позволяет видеть стандарты, которых придерживается код, в том числе относящиеся к безопасности. Таким образом, свободное ПО несравнимо доступнее для изучения, чем проприетарное.

Преимущества и недостатки программного обеспечения

Порой можно услышать возражение, что преимущества свободного ПО в отношении проверки ничего не значат, поскольку для установления наличия проблем достаточно анализа трафика и системных запросов, а также изучения данных о состоянии системы при сбое. Если они не выявили ничего неприятного или незаявленного, то значит ПО безопасно, даже если оно несвободное. Это спорное утверждение, тем более, что последний метод в случае проприетарного ПО нередко затруднен. И конечно, если эти методы не выявили проблем, это не значит, что их нет. Могут возразить, что если в свободном ПО не удалось выявить проблем, это тоже не значит, что их там нет. Это так. Но во-первых, для свободного ПО возможности проверки заметно больше. Во-вторых, выше уже были приведены примеры сравнения свободного ПО и проприетарного. Они однозначно демонстрируют, как власть развращает разработчиков, подталкивая их внедрять вредоносные особенности и препятствовать противодействию им. И хотя в свободном ПО тоже иногда наблюдаются неприятные моменты, в нем они присутствуют в меньшей степени, а возможностей для их устранения гораздо больше.

Существует позиция, согласно которой, сама по себе слежка безобидна, пока она исходит от корпораций и власти, а вот реальную опасность представляют мелкие злоумышленники. И потому ориентироваться нужно не на то, реализована или нет в ПО слежка, а на то, как это ПО защищено от проникновения со стороны. Нередко заявляется, что в проприетарном ПО эффективнее закрываются уязвимости, а также реализуются новые уникальные методы безопасности, которые гораздо позже вводятся в свободное ПО. Иногда даже заявляется, что открытый код, это проблема, потому что злоумышленникам в нем проще искать дыры. А значит, проприетарное ПО, якобы, более безопасно.

Для начала рассмотрим вопрос, действительно ли слежка со стороны крупных корпораций безобидна? Ведь порой указывают, что корпорации собирают сведения, чтобы узнать подробности использования их ПО и сделать его лучше. Чем больше они знают о взаимодействии пользователей с их разработками, тем яснее будут для них потребности клиентов, а значит они смогут переработать свое ПО в соответствии с ними. Это и впрямь так, однако неосмотрительно игнорировать факт чьего-то доступа к чувствительной информации. Данные пользователей могут быть использованы против них самих.[28] Например некоторые компании, разрабатывающие приложения для отслеживания фертильности женщин и течения беременности, делятся с работодателями — которые продвигают их приложения — данными о том, кто из их сотрудниц забеременел или планирует беременность. К сожалению, не редки случаи, когда работодатели отказывают в повышении или даже увольняют беременных или стремящихся к беременности женщин. Разработчики этих приложений также делятся информацией со страховыми компаниями, что позволяет тем навязывать женщинам свои услуги.[29] Данные пользователей могут быть использованы также, например, для проведения мошеннических схем с онлайн-банкингом. Конечно подозревать сами корпорации в ведении подобной деятельности безосновательно. С другой стороны, у сотрудников, имеющих доступ к пользовательским данным, намерения могут быть самыми разными. Факт такого доступа нередко игнорируется, и совершенно необоснованно. Например Apple записывала разговоры своих клиентов, при этом сотрудники прослушивали и анализировали их, чтобы использовать в маркетинговых целях.[30] Аналогичную практику проводила и Microsoft.[31] Клиенты компаний ничего не подозревали о том, что за ними постоянно ведется прослушка, и потому говорили порой на весьма личные темы. Но ведь и такую информацию можно использовать, чем компании не брезгуют, что уже было показано выше. Кроме того, доступ к собранным сведениям могут получить, как раз злоумышленники. Известно немало случаев сливов данных.[32]

Иногда возражают, что ведь не сами корпорации сливают данные, поэтому их обвинять неуместно. Данное возражение сложно воспринимать всерьез, — если бы компании не собирали данные, то и сливать было бы нечего. Кроме того, известны случаи, когда злоумышленники обманом заставляли корпорации самим предоставлять им данные.[33]

Вредоносные функции проприетарного ПО не ограничиваются одной лишь слежкой. В нем также могут присутствовать лазейки. Нередко эти лазейки могут использоваться для настоящего саботажа. Например удаленного

отключения программного обеспечения,[34] удаления пользовательских файлов,[35] препятствования работе с продуктами конкурентов,[36] и еще много другого.

Что касается вопроса с уязвимостями. Здесь часто указывают на пример более свободных прошивок чем обычный Android — таких как LineageOS — отмечая, что в них уязвимости порой закрываются медленнее, чем в самой системе от Google.[37] Но ведь не вина разработчиков более свободных систем, что у них нет таких мощностей, как у корпораций. И самое главное, более быстрое закрытие уязвимостей в системе, где навалом слежки и лазеек, не делает ее более безопасной, чем ту, из которой этот функционал устранен. Погоня за обновлениями порой приводит к совершенно неадекватным попыткам использовать самые свежие версии, игнорируя другие проблемы программного обеспечения.[38] Игнорируется и то, что порой обновления сами содержат ошибки, которые могут повлиять на стабильность работы системы, а также могут открывать новые уязвимости.[39] Не стоит забывать и про случаи, когда обновления отменяют настройки ранее сделанные пользователем, в том числе касающиеся безопасности,[40] добавляют новую слежку,[41] вводят новые ограничения на возможности работы.[42] Все сказанное, конечно, не означает, что стоит пренебрегать обновлениями. Но важно понимать, что гнаться за ними, пренебрегая другими параметрами ПО, неосмотрительно. Что касается количества уязвимостей в свободных и проприетарных системах, то выстраивание аргументации на основе данных о нем, мягко говоря, сомнительно. Если просто посмотреть некоторые исследования, то данные по этому вопросу оказываются противоречивы. За одни года, исследования находят больше уязвимостей в Debian, чем в MacOS,[43] за иные — больше обнаруживают в MacOS, чем в системах GNU/Linux и даже Windows.[44] В одно время верхние строки по количеству уязвимостей, вместе с Windows и Android, занимает только ядро Linux, а сами свободные операционные системы далеко отстают и от них, и от MacOS.[45] В иное, напротив, именно популярные системы GNU/Linux имеют наибольшее количество уязвимостей по сравнению с ядром Linux и с проприетарными системами.[46] Нередко указывают на сомнительность данных, поскольку корпорации могут скрывать статистику по ошибкам, тогда как в свободных системах разработка открыта. Кроме того, сам по себе факт установления наличия большего числа уязвимостей не говорит, что система менее безопасна. Наоборот, это может говорить о том, что их выявляют качественнее, а значит и закрывают их больше. Существуют и исследования, как раз показывающие, что в свободных системах уязвимости закрывают быстрее.[47] В общем, объявлять некоторую систему более уязвимой по причине обнаружения в ней наибольшего количества уязвимостей безосновательно.

Также указывают на пример свободных магазинов приложений, отмечая, что те позволяют скачивать приложения для старых версий ОС, с более низкими требованиями по безопасности — например отсутствием возможности предотвращения повышения привилегий.[48] Но ведь не их вина, что многие пользователи вынуждены пользоваться устройствами со старыми версиями Android. Это вина как раз корпораций, практикующих запланированное устаревание, разрабатывающих устройства, на которые невозможно поставить более новую версию. Свободные магазины приложений, наоборот, стараются заботится о таких пользователях. Могут возразить, что компании не виноваты, поскольку невозможно предугадать какой аппаратной архитектуры потребуют новые версии. Этот аргумент можно было бы принять, если бы не некоторые производители, которые все-таки каким-то образом выпускают устройства, позволяющие обновляться до более новых версий. Кроме того, не стоит забывать, что более новые устройства вовсе необязательно проектируются более безопасными, как уверяют апологеты проприетарного ПО. Свободная прошивка Android — Replicant — не разрабатывается для новых устройств. Одна из причин в том, что современные смартфоны повсеместно проектируются с прямым доступом радиомодуля к памяти, а также невозможностью извлечь аккумулятор.[49] Такая аппаратная реализация едва ли может считаться более безопасной, чем иная, которая имеет место в смартфонах, поддерживаемых свободной прошивкой. Ведь в этом случае радиомодуль может получать доступ к другим компонентам системы. Настоящая проблема как раз в том, что производители используют именно такую конфигурацию.

Теперь рассмотрим вопрос с реализаций уникальных методов безопасности. Иногда можно услышать заявление, что у свободного ПО с этим все довольно плохо. Например, указывают на то, что некоторые проприетарные системы, такие как MacOS, реализуют изоляцию приложений через песочницы. И если то или иное приложение в них будет скомпрометировано, оно не сможет навредить другим, поскольку они работают в отдельной среде.[50] Указывают, что у систем GNU/Linux такой реализации или вовсе нет, или она менее надежна. Большинство приложений в них проектируются с расчетом на полный доступ к системе.[51] Хотя такая реализация действительно имеет проблемы безопасности, и по-умолчанию во многих системах контроль прав доступа действительно реализован довольно слабо, у пользователя есть возможность его настроить и сделать гораздо более серьезным. При этом нет повода опасаться, что в системе есть некие функции, которые дают кому-то больше возможностей по такой настройке, чем самому пользователю. В проприетарных же системах, у разработчика есть привилегии по изменению системы. В свободных системах

помимо возможности пользователю контролировать реализацию изоляции процессов, есть также возможность в еще большей мере реализовать изоляцию, например, за счет виртуализации. Ее возможно использовать и в проприетарных системах. Но с учетом наличия иных проблем в проприетарном ПО, эта возможность не делает его более привлекательным.

Еще один пример — в системах GNU/Linux нет эффективных способов проверки подлинности загрузки, тогда как в MacOS, эти способы есть. Они предотвращают возможность подделки программного обеспечения на устройстве, а также установки вредоносного ПО в системный раздел.[52] Для того, чтобы осуществить такую атаку, необходимо иметь физический доступ к оборудованию. Нужно ли говорить, что опасаться подобного взлома имеет смысл очень немногим — обычные пользователи вряд ли имеют шанс стать его жертвами. При этом, как уже было показано выше, в проприетарных системах, присутствует большое количество вредоносного функционала, заложенного самими корпорациями. Много ли смысла опасаться физического взлома и не опасаться слежки через сеть? Таким образом, как преимущество, по крайней мере для большинства пользователей, этот функционал выглядит крайне сомнительно.

Существует, правда, возражение, что проверка подлинности загрузки, также предотвращает определенные вредоносные действия, которые могут быть осуществлены удаленно. Например, существует вредоносное ПО, которое может установить старую версию системы через механизм обновлений или иным путем, в которой еще не закрыты некоторые уязвимости, после чего сможет их эксплуатировать. Проверка подлинности загрузки же реализует, так называемую, защиту от отката, не позволяющую загружать старые версии системы. В результате этого, при попадании такого ПО, после перезагрузки устройства, оно будет удалено и система восстановлена.[53] На это следует заметить, что вредоносное ПО обычно залетает в систему либо с сомнительными приложениями, которые пользователь неосмотрительно себе устанавливает, либо через сайты, которые пользователь посещает. Вероятно, что после перезагрузи пользователь снова установит такое приложение или посетит такой сайт. К тому же, если говорить о смартфонах, то обычный пользователь крайне редко его перезагружает. В конце концов, взломщикам просто нерационально разрабатывать столь сложное программное обеспечение. При этом, если конкретный пользователь по каким-то причинам им все-таки сильно нужен, они могут заразить его устройство повторно. Потому польза от данной функции все еще остается незначительной. Конечно, лишней данная функция безопасности бы не была. Но с учетом того, что в проприетарном ПО есть

проблемный функционал, не свойственный свободному, его наличие в первом едва ли дает ему хоть какое-то преимущество.

Что касается вопроса упрощения для злоумышленника поиска уязвимостей в свободном ПО. Конечно, программное обеспечение не может делать разделения на предоставление возможностей для своей проверки в зависимости от намерений проверяющего. Лучше всего здесь будет опять сравнить данные по свободному ПО и проприетарному. Большинство вирусов ориентированы на систему Windows, являющуюся проприетарной.[54] Конечно, это является следствием ее популярности, а потому само по себе еще не говорит в пользу менее популярных свободных систем.[55] Но важно то, что использование проприетарной лицензии и недоступность ее исходного кода не спасло эту систему от создания для нее вирусов. Точно также как растет количество вредоносного ПО для GNU/Linux,[56] растет оно и для MacOS.[57] Потому аргумент про упрощение поиска дыр нельзя считать весомым. Точно также, как открытость кода позволяет искать дыры для их эксплуатации, она позволяет их выявлять и для закрытия. Что касается случая, о котором говорилось в начале — когда исследователи из Миннесотского университета пытались в рамках эксперимента внедрять в свободные системы бэкдоры — то нет никаких оснований полагать, что данная практика имеет широкое распространение. Тем более, что как показал этот случай, нередко такие проблемы достаточно быстро выявляются разработчиками, получающими патчи. Сами разработчики Linux их выявили независимо, еще до публикации исследования, и не внесли в ядро проблемный код.[58] Да, существуют отдельные свидетельства прецедентов подобного рода. Например в свободном приложении Webmin находили лазейку, которая, как оказалось, была внедрена взломщиками.[59] Но известны и случаи, когда злоумышленники внедряли лазейки и в проприетарное ПО.

Например компания Juniper заявляла, что обнаружила уязвимости, внедренные в их ПО и эксплуатировавшиеся неизвестной стороной.[60] Поскольку их программы проприетарные, это вызвало негодование в сообществе — каким образом в такое ПО кто-то со стороны мог внедрить какой-то функционал, как это возможно без доступа к исходному коду и инфраструктуре распространения программ этой компании? Закономерно, что возникли подозрения о внедрении этих уязвимостей разработчиками намеренно, а после вскрытия факта их наличия, их закрыли и объявили о том, что внедрил их кто-то другой. Не удивительно, что после этого пошли шутки о том, что обновление, закрывающее уязвимости, заменяло старые лазейки на новые. Таким образом, либо в проприетарное ПО также можно ввести бэкдоры со стороны, либо его разработчики сами таковые закладывают. Ни тот ни другой вариант не делает такое ПО пригляднее свободного.

Иногда указывают, что свободное ПО менее функционально по сравнению с проприетарным. В отношении некоторых программ, особенно профессиональной направленности, это действительно так. Однако, если говорить о повседневном использовании, то функционал свободных программ вполне сопоставим с возможностями проприетарного ПО. Учитывая все проблемы последнего, больший функционал нельзя считать поводом предпочесть такое ПО, за исключением случаев, когда без такого функционала действительно обойтись не выходит. Несовершенство и угнетение — не одно и то же.[61] Проприетарное ПО не дает пользователю контроля, и в результате разработчики реализуют в нем множество вредоносных особенностей.[62] Свободное такой контроль дает.

Иногда апологеты проприетарного ПО, говорят, что те или иные преимущества такого ПО в плане безопасности перед свободным, это лишь факты. Выше уже было показано, что это весьма односторонний и предвзятый взгляд на вопрос. Наличие слежки и лазеек в нем, это тоже факты. Те, кто отмахивается от проблем слежки заявлением, что ее можно отключить, игнорируют факты ограниченности такого отключения в проприетарном ПО.

Само по себе наличие или отсутствие определенного функционала в тех или иных программах, это действительно лишь факты. И соответственно, описание этого функционала, это и впрямь лишь изложение фактов. Но когда речь идет о практике использования программ, речь уже не просто о фактах. Ведь разные практики имеют разные социальные последствия. Соответственно разные рекомендации по использованию программного обеспечения приводят к разным социальным результатам. Одни практики имеют последствия социально вредные, иные — благие.

В случае использования свободного ПО оказывается поддержка сообществу, практикам взаимопомощи и сотрудничества. В случае использования проприетарного — корпорациям, системе эксплуатации и угнетения.

Пагубность власти корпораций

Иногда можно услышать возражение, что критики корпораций несправедливы, поскольку те вкладывают миллионы в разработку открытого ПО, много таких программ было выпущено именно ими. Например Google разработали свободную систему Android, свободный браузер Chromium и многое другое. Но ведь в том же Android, огромное количество несвободных прикладных приложений, разработанных тем же Google, который не спешит делать их свободными. Библиотеки в нем также проприетарны. Как и все ПО их сервисов. Корпорация Microsoft выпускает под свободной лицензией такие

программы как Visual Studio Code и Windows Driver Frameworks. Но основной их инструмент — Windows — остается проприетарным, как и множество других программ. Корпорация Apple открыла ядра своих операционных систем, но оболочка — буквально все приложения, разработанные ими — остаются проприетарными. Так что их вклад в свободное ПО ничтожен по сравнению с вкладом в проприетарное. Здесь таже ситуация, что и с благотворительностью, которой занимаются миллиардеры — они вкладывают миллионы в борьбу с бедностью, которую сами же и создали, выкачав из угнетенных миллиарды.[63]

Могут возразить, но ведь они могли бы не вкладывать деньги в свободное ПО, а делать все его проприетарным. Могли бы, но использование в определенной мере практик открытости обеспечивает им конкурентное преимущество.[64] И используют они его ровно в той мере, в какой оно это преимущество дает.

Не лишним будет заметить, что говоря о важности открытого ПО корпорации делают акцент именно на открытости, а не на свободе. Разница между открытым ПО и свободным в том, что первое может не предоставлять тех или иных свобод пользователю.[65] Хоть у него и открыт исходный код, но иногда его не дозволяется изменять, или налагается ограничение на распространение измененных копий. Таким образом, такое ПО пользователь все равно полноценно не контролирует. Корпорации и те, кто поддерживают открытое ПО указывают на то, что это просто удобный способ разработки.[66] В такой интерпретации упускается важность свободы как таковой, что препятствует осознанию пагубности власти тех, кто разрабатывает программы и пытается с помощью них эксплуатировать пользователей. Таким образом, поддержка проприетарного ПО и даже открытого, при подчеркивании лишь преимуществ открытости исходного кода, оборачивается поддержкой множества других грязных практик.

Например поддержкой попустительства репрессивной форме авторского права. Ведь именно с помощью него проприетарное ПО получает власть над пользователем. Проприетарные лицензии налагают ограничения на возможности пользователя контролировать программу — изучать ее и изменять, что уже было показано выше. Они также налагают ограничения на распространение программ — не позволяют пользователям сотрудничать друг с другом, не позволяют копировать программы и делится ими с другими людьми. Такая практика оправдывается тем, что в ином случае страдает доход авторов. На самом деле, во-первых, владельцами прав не всегда являются сами авторы. Во-вторых, как неоднократно показывала практика, деньги вполне можно хорошо зарабатывать и без препятствия обмену среди людей.[67]

Авторское право является практикой применения к информации мерила материальных объектов. Когда материальный объект забирается у владельца, у него данного объекта больше нет. Но при копировании программы или файла, у владельца первоначальной копии ничего не отнимается.[68] Потому говорить о «воровстве» или «пиратстве» неправомерно. Тем не менее, такая риторика позволяет оправдать внесение в область информации практики, позволяющей одним людям эксплуатировать других. Практики, опирающейся на идею, отражающую стремление к угнетению — право собственности.

Часто под собственностью понимают любую вещь, которой человек обладает. Однако в контексте устройства общества такое понимание вносит путаницу. Владение тех или иных индивидов едой или одеждой не определяет социально-экономического устройства. Его определяет то, что является источником предметов потребления, то что позволяет их производить. В том случае, если одни члены общества такими средствами обладают, а другие нет, вторые вынуждены работать на первых. В этом случае собственники таких средств присваивают результаты их работы, некоторую долю которых отдают обратно им в виде платы. Таким образом, о праве собственности состоятельно говорить только в отношении средств производства, причем только тех, которые позволяют одним людям присваивать себе плоды труда других. Предметы потребления к этому не относятся. Однако распространение на них идеи собственности, позволяет оправдать владение средствами производства, мол, это такая же собственность, и покушение на нее это воровство, отъем средств к существованию. Хотя на деле, именно владение средствами производства является воровством, поскольку позволяет владельцу получать прибыль взиманием ренты с тех, кто вынужден работать на принадлежащих ему средствах. То есть присваивать результаты чужого труда. Не относится к этому и информация, поскольку возможности ее копирования, в отличии от материальных ресурсов, не ограниченны.

Иногда заявляют, что автор произведения или программы провел работу, создал нечто нужное, а потому ему логично требовать за это плату. Это так, но одно дело плата за работу, и совсем другое неограниченное взимание ренты за каждое ее использование. Если произведение позволяет автору получать доход в сотни раз превосходящий доход тех, кто использует его произведение, это уже не плата за труд — это эксплуатация. Каким бы важным не было произведение, оно не может быть оправданием для извлечения из пользователей такого количества средств в пользу автора, что он может обеспечить себе беззаботную жизнь, а они остаются прозябать в бедности. Таким образом, говорить об отъеме средств к существованию, в случае копирования и распространения информации, также не правомерно. Однако, говоря такое, также становится

возможным оправдать практику эксплуатации. Ведь людей вынуждают отдавать свои средства — произведенную ими стоимость — за пользование информацией, которая не ограниченна. То есть опять же имеет место присвоение результатов чужого труда.

Применение права собственности к информации, ровно как и попытка распространить его на предметы потребления, необходимо системе угнетения, а потому эксплуататоры такие идеи раскручивают. Они смешивают обладание теми материальными предметами, которые потребляются, и теми, которые производят новые, с информацией, которая не потребляется и использование которой для производства также ее не уменьшает. Это ключевое отличие информации от материальных предметов. Предметы потребления в процессе потребления исчезают. Средства производства в процессе производства также уменьшаются. Но информация не убывает сколько бы копий ее не было сделано, и сколько бы раз не обращались к ним.[69] Однако собственники упорно используют риторику «потребления» информации, когда речь идет, например, о просмотре фильма или прослушивании музыки. Это выгодно, поскольку позволяет создать впечатление, что такое «потребление» ничем не отличается от материального, что оно также отнимает ресурсы. А значит за каждую копию и даже за каждый акт использования нужно взимать плату. Для того, чтобы было возможным реализовывать это и применяется авторское право. А чтобы серьезно затруднить возможность смутьянам использовать свойства информации передавать ее, не теряя самому, применяются различные технические наработки, такие как DRM и проектирование несовместимых форматов.[70]

Проприетарное ПО реализует представление об информации, именно как предмете потребления, в контексте, где его значение объявляется равнозначным средству производства. Практика использования проприетарного ПО, вместо свободного, подпитывает именно такое представление. Таким образом подпитывается миф о естественности и необходимости права собственности, которое и является основным источником пороков человеческой цивилизации. Ведь именно право собственности позволяет все тем же корпорациям выкачивать огромные средства из работников, в первую очередь в странах «третьего мира», куда вынесено производство корпораций из стран «первого мира», оставляя их в нищете.[71] Именно право собственности является причиной чудовищного социального неравенства.[72] Следствием именно этого является голод миллиардов людей на всей планете.[73] Распределение производимого богатства никак нельзя считать справедливым. Зарплата тех, кто занимается дизайном и маркетингом, сидя в головных офисах превосходит зарплату тех, кто собирает микросхемы, стоя у конвейера в цеху в десятки, а порой и в сотни

раз. При этом их доходы могут уступать доходам собственников бизнеса в тысячи раз.[74] В результате горстка богачей получает абсолютно все, а большинство не имеет возможности удовлетворить даже самые основные потребности.[75] Погружение большей части человечества в страдания позволяет собственникам капитала обеспечить себе райскую жизнь. А для того, чтобы сохранить свое положение, собственники капитала вынуждены прибегать и к другим методам наращивания прибылей, — экономить не только на рабочей силе, но и на технике производства.[76]

Это выливается, например, в отказ от экологически чистых практик, следствием чего является повальное загрязнение окружающей среды.[77] Оно в свою очередь влечет весьма мрачные последствия в виде глобального потепления,[78] загрязнения воздуха,[79] сокращения биоразнообразия,[80] и еще много много другого. Оно же потворствует чрезмерной растрате ресурсов,[81] не только ввиду неэффективных из-за дешевизны методик производства, но и за счет запланированного устаревания, т.е. практики производства товаров с низким сроком эксплуатации.[82] Многие отрицают ее наличие, говоря, что никакого запланированного устаревания нет, а есть только стремление отдельных производителей удешевить товар, за счет его качества. Но от того, что вместо слов «запланированное устаревание» используется фраза «стремление производителей удешевить товар» суть не меняется. Вот чему способствуют те, кто рекомендует регулярно менять устройства, чтобы якобы, достичь наибольшей безопасности.

Таким образом, использование проприетарного ПО попустительствует грязным практикам собственников капитала и, в конце концов, увеличению их власти, и усугублению положения угнетенных.

Слежка и контроль, внедренные в проприетарные программы, позволяют серьезнейшим образом увеличить возможности экономического и социального угнетения. Это позволяет собственникам капитала и власти эффективнее выявлять и устранять несогласных. Спецслужбы активно пользуются возможностями слежки.[83] А корпорации им потворствуют.[84] Свою власть они могут использовать не только для подавления противников системы, но и в личных интересах, например для отслеживания привлекательных женщин.[85]

Возможность противодействия угнетению

Использование проприетарного ПО негативно сказывается на обществе. Функции безопасности, реализованные в нем, никак не способны защитить от собственников капитала, которые сохраняют над ним полный контроль.

Выше было приведено множество сведений пагубных моментов проприетарного ПО. Все эти гнусности и изъяны говорят об одном,

безопасность бессмысленно искать там, где нет приватности, а приватности нет без свободы.

Использование свободных программ препятствует сбору данных корпорациями. Оно позволяет пользователям контролировать свои вычисления. Способствует распространению практики взаимопомощи и сотрудничества. Его использование, это поддержка благих начинаний. Вклад в общество равенства. Вклад в социальное освобождение. Именно поэтому необходимо настаивать на использовании свободных программ, способствовать его распространению. Для этого необходимо объяснять пагубность практики использования проприетарного ПО. Раскрывать его гнусные особенности, вредоносные функции. Разъяснять взаимосвязь такого ПО и социального неравенства, усугубления экологической обстановки, отчуждения. Составлять подробные и ясные инструкции по использованию свободных программ. Не только отдельных прикладных приложений, но и свободных операционных систем, свободных драйверов и прошивок. По возможности стараться идти дальше и развивать свободное оборудование — проектируемое со свободными чертежами и спецификациями.

Безусловно не стоит питать иллюзий и полагать, что само по себе использование свободного ПО изменит существующую социально-экономическую систему. Для ее ниспровержения понадобятся иные методы, которые еще не найдены. Но свободное ПО, предотвращая усиление власти собственников капитала, способно посодействовать этому. Использование же проприетарного, пусть и с реализацией самых современных методов безопасности, нет. Потому не нужно делать вывод, что переход на свободное ПО бессмысленный и лучше остаться на проприетарном. Тем более не стоит так полагать на том основании, что слежка и контроль останется через другие каналы. Закрытие хотя бы некоторых, это уже снижение контроля.

Иногда говорят, что тем, кто стремиться к свободе нужно не только переходить на свободное ПО, но и жить вдали от городов с камерами слежения, самому выращивать хлеб, шить одежду, изготавливать бытовую технику. Это невозможно, ибо сложно согласится с приемлемостью затворничества для большинства людей. Подобное доведение до абсурда не состоятельно. В том-то и дело, что программное обеспечение, это с одной стороны, та область, в которой простой человек без непомерных затрат может снизить контроль над собой со стороны системы. Это обусловлено ключевым отличием информации от материальных объектов — возможностью ее воспроизводить, копировать и распространять, не тратя самому. В областях касающихся материальных предметов потребления дело обстоит иначе. Здесь остается уповать только на нахождение эффективных методик преобразования общества. И с другой

стороны, учитывая особое место программного обеспечения в современной цивилизации — с ним связаны буквально все сферы человеческой деятельности — достижение в нем свободы, способно внести свой вклад в такое преобразование.

Конечно не стоит впадать и в другую крайность и говорить, что бессмысленно использовать свободную систему с несвободным драйвером, и что обязательно нужно искать свободное оборудование. Такого оборудования сейчас, к сожалению, нет. Существуют некоторые проекты, например свободного процессора[86] или Wi-Fi-адаптера,[87] но они не имеют распространения, и такое оборудование практически невозможно достать. Иногда, правда, под свободным оборудованием понимается не то, которое проектируется по свободным чертежам, или в котором используются свободные прошивки, а то, которое просто хорошо работает со свободными системами и не требует для этого установки несвободных драйверов. Как правило, это устройства, в которые прошивка уже внесена. И она всегда проприетарна. Просто те системы, из которых удален функционал для подтягивания несвободных прошивок смогут работать с ней. А с тем оборудованием, которому нужна ее отдельная загрузка, нет. У такого подхода есть преимущество — разработчик не может в любой момент времени внедрить в прошивку новый вредоносный функционал через обновления. В случае с зашитой прошивкой он может внедрить ее только в процессе производства, а не в процессе эксплуатации пользователем. Недостаток в том, что и исправлять уязвимости он не может. Точно также и пользователь не может такую прошивку изучать, например с помощью обратной разработки, и заменить на другую. Таким образом, подход с настаиванием на полностью свободных системах, т.е. тех, из которых удален функционал для взаимодействия с подтягиваемыми прошивками имеет заметный изъян, хотя имеет и преимущество. Не говоря уже о том, что он заметно снижает охват оборудования, которое возможно использовать. И как следствие снижает количество пользователей свободных систем. Учитывая ситуацию с оборудованием и прошивками, необходимо составлять соответствующие рекомендации.

Если возможно использовать полностью свободное оборудование и свободное программное обеспечение, то стоит использовать именно их. Если свободное оборудование достать проблематично, это требует особых поисков и времени, а также больших финансовых затрат, но имеющееся позволяет использовать свободные прошивки, то стоит придерживаться этого варианта, при этом также используя только свободное ПО. Если оборудование просто хорошо работает со свободными системами, не требует установки несвободных драйверов и подтягивания прошивок, то стоит предпочесть полностью

свободные операционные системы и свободные прикладные программы. Если оборудование отказывается корректно взаимодействовать с полностью свободными системами, то можно использовать те, которые допускают подтягивание прошивок, если менять оборудование не является беспроблемной задачей. Если же финансовые и временные траты на замену оборудования неприемлемы, а аппаратура не работает без несвободных драйверов, то может считаться приемлемым установить проприетарный драйвер. Не стоит отмахиваться от возможности перейти на свободное ПО, на основании плохого взаимодействия оборудования, если возможно обеспечить хорошее. Свободные операционные системы вполне доступны для освоения.

Свободное ПО дает возможность благоприятного взаимодействия благодаря распространению добрых практик.

1 О том, что такое свободные программы говорится в статье «Что такое свободная программа?» https://www.gnu.org/philosophy/free-sw.html

2 *Арс Либрев* «Свобода в мире программного обеспечения» https://mega.nz/file/Ph4TTAoD#Yq6R9mvSzSnx940CCBuKY9fXWOh-cnQx4_UtOaqnBO0

3 Об этом сказано, например, в статье «Эксперты обнаружили критическую уязвимость в sudo» https://habr.com/ru/news/539526/. Также в статье «В ядре Linux обнаружили уязвимость, позволяющую получать права суперпользователя» https://habr.com/ru/articles/275543/

4 *Qiushi Wu, Kangjie Lu* «On the Feasibility of Stealthily Introducing Vulnerabilities in Open-Source Software via Hypocrite Commits» https://qiushiwu.github.io/papers/OpenSourceInsecurity.pdf. *Chris Mason, Steven Rostedt, Christian Brauner, Dan Williams, Greg Kroah-Hartman, Jonathan Corbet, Kees Cook, Laura Abbott, Sasha Levin, Ted Ts'o* «Report on University of Minnesota Breach-of-Trust Incident» https://lkml.org/lkml/2021/5/5/1244. Также об этом сказано в статье «Linux забанил коммиты Миннисотского университета за эксперименты с намеренными некачественными патчами» https://habr.com/ru/news/553712/

5 *Robert Freeman* «IBM X-Force Researcher Finds Significant Vulnerability in Microsoft Windows» https://securityintelligence.com/ibm-x-force-researcher-finds-significant-vulnerability-in-microsoft-windows/. *Tavis Ormandy* «Down the Rabbit-Hole» https://googleprojectzero.blogspot.com/2019/08/down-rabbit-hole.html. А также статья «Обнаружена опаснейшая уязвимость Windows DNS Server» https://habr.com/ru/articles/511002/. Также «Apple закрыла дыру MacOS с повышением полномочий sudo» https://habr.com/ru/news/541706/. *Наталия Ришко* «Apple десятилетиями скрывала опасную уязвимость MacOS» https://midgardinfo.com/news/apple_desjatiletijami_skryvala_opasnuju_ujazvimost_macos/2019-06-05-2978

6 *Micah Lee* «Recently Bought a Windows Computer? Microsoft Probably Has Your Encryption Key» https://theintercept.com/2015/12/28/recently-bought-a-windows-computer-microsoft-probably-has-your-encryption-key/. Также об этом сказано в статье «Microsoft’s new small print — how your personal data is (ab)used» https://edri.org/our-work/microsofts-new-small-print-how-your-personal-data-abused/

7 *Chris Welch* «Google remotely changed the settings on a bunch of phones running Android 9 Pie» https://www.theverge.com/2018/9/14/17861150/google-battery-saver-android-9-pie-remote-settings-change

8 *Nicholas Deleon* «Apple can remotely remove applications from your iPhone» https://techcrunch.com/2008/08/07/apple-can-remotely-remove-applications-from-your-iphone/

9 *David Pogue* «Some E-Books Are More Equal Than Others» https://archive.nytimes.com/pogue.blogs.nytimes.com/2009/07/17/some-e-books-are-more-equal-than-others/. *Cory Doctorow* «Kindle user claims Amazon deleted whole library without explanation» https://boingboing.net/2012/10/22/kindle-user-claims-amazon-dele.html. *Matthew Humphries* «Samsung Can Remotely Disable Any of Its TVs Worldwide» https://www.pcmag.com/news/samsung-can-remotely-disable-any-of-its-tvs-worldwide. Также «Samsung может удаленно отключать умные телевизоры по всему миру с помощью функции TV Block» https://habr.com/ru/news/574508/. *Jane McEntegart* «Google Remotely Deletes Apps on Android Phones» https://www.tomsguide.com/us/Android-Google-Applications-Android-Apps,news-7216.html. *Ryne Hager* «Google remotely deletes The Great Suspender Chrome extension after malware accusations» https://www.androidpolice.com/2021/02/04/google-remotely-disables-the-great-suspender-chrome-extension-after-malware-accusations/. *Gregg Keizer* «Google throws kill switch on Android phones» https://www.computerworld.com/article/2506557/google-throw—kill-switch--on-android-phones.html. *Gregg Keizer* «Microsoft: We can remotely delete Windows 8 apps» https://www.computerworld.com/article/2732767/microsof--we-can-remotely-delete-windows-8-apps.html. *Natasha Lomas* «Encrypted messaging platform WhatsApp denies “backdoor” claim» https://techcrunch.com/2017/01/13/encrypted-messaging-platform-whatsapp-denies-backdoor-claim/. *Anna Tims* «How can HP block me from using a cheaper printer cartridge?» https://www.theguardian.com/money/2023/may/10/how-can-hp-block-me-from-using-a-cheaper-printer-cartridge. *Matt Burgess* «Locked out of God mode, runners are hacking their treadmills» https://arstechnica.com/information-technology/2021/11/locked-out-of-god-mode-runners-are-hacking-their-treadmills/

10 Сводку статей, в которых раскрываются различные гнусные особенности несвободного ПО можно посмотреть на этой странице https://www.gnu.org/proprietary/proprietary.html

11 Заявление конфиденциальности Microsoft https://privacy.microsoft.com/ru-ru/privacystatement. Политика конфиденциальности Apple https://www.apple.com/ru/legal/privacy/ru/. Политика конфиденциальности Google https://policies.google.com/privacy?hl=ru. Политика использования

данных Facebook*

12 *Sofia Elizabella Wyciślik-Wilson* «Microsoft is using the KB5021751 update to see if you have an unsupported version of Office installed» https://betanews.com/2023/01/19/microsoft-is-using-the-kb5021751-update-to-see-if-you-have-an-unsupported-version-of-office-installed/. Похожим было и и другое исправление Windows, о котором говорится в статье «Windows 10: Update KB4023057 re-released» https://borncity.com/win/2019/01/17/windows-10-update-kb4023057-re-released-1-16-2019/. То, что его функционал включал в себя копание в настройках пользователя и, соответственно, позволял изменить то, что тот мог намеренно отключить, показано на этой странице https://support.microsoft.com/en-us/topic/kb4023057-update-health-tools-windows-update-service-components-fccad0ca-dc10-2e46-9ed1-7e392450fb3a. *Andrew Orlowski* «Sneaky Microsoft renamed its data slurper before sticking it back in Windows 10» https://www.theregister.com/2015/11/26/microsoft_renamed_data_slurper_reinserted_windows_10/

13 На это указано, например, в статье *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#dynamic-analysis

14 *Geoffrey A. Fowler* «Google Chrome has become surveillance software. It’s time to switch» https://www.mercurynews.com/2019/06/21/google-chrome-has-become-surveillance-software-its-time-to-switch/. Также сведения о слежке в браузере Google Chrome приведены на данной странице https://spyware.neocities.org/articles/chrome

15 Сведения о слежке в браузере Opera приведены в статье «Слежка в Opera и как ее отключить» https://spy-soft.net/slezhka-v-opera/. Сведения о вредоносных функциях Opera https://spyware.neocities.org/articles/opera

16 Сведения о слежке в Яндекс.Браузер приведены в статье «Слежка в Яндекс Браузер» http://www.spy-soft.net/slezhka-yandex-browser/. Также слежка через данный браузер показана на этой странице https://reports.exodus-privacy.eu.org/en/reports/85066/

17 Это показано на данной странице https://spyware.neocities.org/articles/firefox

18 Это отражено на этой странице https://spyware.neocities.org/guides/firefox

19 Об этом говорится на этой странице https://spyware.neocities.org/articles/librewolf

20 Сведения о различных браузерах приведены на этой странице https://spyware.neocities.org/articles/. Сведения, например, по свободному браузеру Lynx даны на этой странице https://spyware.neocities.org/articles/lynx

21 См. ссылку в сноске 10

22 *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#dynamic-analysis

23 *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#special-builds

24 *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#fuzzing

25 *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#good-counter-arguments

26 *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#binary-analysis

27 *Rohan Kumar* «FLOSS Security» https://seirdy.one/posts/2022/02/02/floss-security/#good-counter-arguments

28 Об этом говорится в статье «Утечка данных: в чем опасность и как с этим бороться?» https://habr.com/ru/companies/astralinux/articles/707258/

29 *Arwa Mahdawi* «There's a dark side to women's health apps: Menstrual surveillance» https://www.theguardian.com/world/2019/apr/13/theres-a-dark-side-to-womens-health-apps-menstrual-surveillance

30 *Thomas Le Bonniec* «On the matter of Apple'smassive collection of recordings and data» https://www.politico.eu/wp-content/uploads/2020/05/Public-Statement-Siri-recordings-TLB.pdf

31 *Joseph Cox* «Microsoft Contractors Listened to Xbox Owners in Their Homes» https://www.vice.com/en/article/43kv4q/microsoft-human-contractors-listened-to-xbox-owners-homes-kinect-cortana

32 *Мария Нефедова* «У Microsoft произошла утечка данных, коснувшаяся 65000 организаций по всему миру» https://xakep.ru/2022/10/20/microsoft-leak-3/. Также о разных утечках из корпорации Microsoft сказано на этой странице https://clickfraud.ru/istoriya-utechek-dannyh-microsoft-i-polnaya-hronologiya-do-2023-goda/. Также см. статью «GitHub стал источником утечки для Microsoft: кто допустил ошибку?» https://www.securitylab.ru/news/541954.php. Об утечках Apple сказано на вот этой странице https://clickfraud.ru/istoriya-narushenij-dannyh-apple-i-polnaya-hronologiya-do-2023-goda/. Еще см. статью «Applebee's Hit With POS Data Breach» https://www.pymnts.com/news/security-and-risk/2018/applebees-malware-card-theft/

33 Об этом сказано в статье «Сообщается, что Apple и Facebook* предоставили личные данные пользователей хакерам, выдававшим себя за правительственные органы» https://applepro.news/soobshhaetsya-chto-apple-i-facebook-predostavili-lichnye-dannye-polzovatelej-hakeram-vydavavshim-sebya-za-pravoohranitelnye-organy/

34 *Kit Walsh* «Nest Reminds Customers That Ownership Isn't What It Used to Be» https://www.eff.org/deeplinks/2016/04/nest-reminds-customers-ownership-isnt-what-it-used-be. *Karl Bode* «Printer Makers Are Crippling Cheap Ink Cartridges Via Bogus Security Updates» https://www.vice.com/en/article/pa98ab/printer-makers-are-crippling-cheap-ink-cartridges-via-bogus-security-updates

35 *Matt Burgess* «Locked out of “God mode,” runners are hacking their treadmills» https://arstechnica.com/information-technology/2021/11/locked-out-of-god-mode-runners-are-hacking-their-treadmills/. *Samuel Gibbs* «Apple deleted music from users’ iPods purchased from rivals, court told» https://www.theguardian.com/technology/2014/dec/04/apple-deleted-music-ipods-rivals-steve-jobs

36 *Ars Staff* «Windows Update drivers bricking USB serial chips beloved of hardware hackers» https://arstechnica.com/information-technology/2014/10/windows-update-drivers-bricking-usb-serial-chips-beloved-of-hardware-hackers/. *Tim Cushing* «Light Bulb DRM: Philips Locks Purchasers Out Of Third-Party Bulbs With Firmware Update» https://www.techdirt.com/2015/12/14/lightbulb-drm-philips-locks-purchasers-out-third-party-bulbs-with-firmware-update/

37 См., например, статью «Choosing Your Android-Based Operating System» https://privsec.dev/posts/android/choosing-your-android-based-operating-system/#firmware-updates

38 См., например, статью «Choosing Your Desktop Linux Distribution» https://privsec.dev/posts/android/choosing-your-android-based-operating-system/#patch-levels

39 О таком случае говорится в статье «Починка мелкой уязвимости в важной библиотеке Linux вызвала к жизни жуткую дыру» https://forum.sources.ru/index.php?showtopic=422477

40 О таком изменении говорится в статье «Windows 10: Update KB4023057 re-released» https://borncity.com/win/2019/01/17/windows-10-update-kb4023057-re-released-1-16-2019/. То, что его функционал включал в себя копание в настройках пользователя и, соответственно, позволял изменять их показано на этой странице https://support.microsoft.com/en-us/topic/kb4023057-update-health-tools-windows-update-service-components-fccad0ca-dc10-2e46-9ed1-7e392450fb3a. *Andrew Orlowski* «Sneaky Microsoft renamed its data slurper before sticking it back in Windows 10» https://www.theregister.com/2015/11/26/microsoft_renamed_data_slurper_reinserted_windows_10/

41 *Sofia Elizabella Wyciślik-Wilson* «Microsoft is using the KB5021751 update to see if you have an unsupported version of Office installed»

https://betanews.com/2023/01/19/microsoft-is-using-the-kb5021751-update-to-see-if-you-have-an-unsupported-version-of-office-installed/. *Sergiu Gatlan* «Microsoft pushes KB5021751 to check for outdated Office installs» https://www.bleepingcomputer.com/news/microsoft/microsoft-pushes-kb5021751-to-check-for-outdated-office-installs/

42 *Karl Bode* «Printer Makers Are Crippling Cheap Ink Cartridges Via Bogus Security Updates» https://www.vice.com/en/article/pa98ab/printer-makers-are-crippling-cheap-ink-cartridges-via-bogus-security-updates/ *Ars Staff* «Windows Update drivers bricking USB serial chips beloved of hardware hackers» https://arstechnica.com/information-technology/2014/10/windows-update-drivers-bricking-usb-serial-chips-beloved-of-hardware-hackers/. *Tim Cushing* «Light Bulb DRM: Philips Locks Purchasers Out Of Third-Party Bulbs With Firmware Update» https://www.techdirt.com/2015/12/14/lightbulb-drm-philips-locks-purchasers-out-third-party-bulbs-with-firmware-update/

43 Такие данные приведены в исследовании «Vulnerability Alerts» https://thebestvpn.com/vulnerability-alerts/. Также об этом говорится в статье «Linux — самая уязвимая операционная система» https://www.linuxadictos.com/ru/Linux---самая-уязвимая-операционная-система%2C-но-не-на-что-квалифицировать.html. А также в статье «Самая уязвимая ОС — не Windows 10» https://devby.io/news/windows-10-ne-samaya-uyazvimaya-os

44 *Александр Панасенко* «OS X наиболее уязвимая система в мире» https://www.anti-malware.ru/news/2012-04-26/8987. Cristian Florian «Most vulnerable operating systems and applications in 2014» https://web.archive.org/web/20230327110650/https://techtalk.gfi.com/most-vulnerable-operating-systems-and-applications-in-2014/. *Анатолий Ализар* «OS X: самая уязвимая операционная система?» https://xakep.ru/2015/02/24/os-x-bugs/

45 На это указано в статье «Уязвимости операционных систем» https://habr.com/ru/companies/ua-hosting/articles/407979/. *Мария Нефедова* «Эксперты перечислили 15 самых атакуемых уязвимостей в Linux» https://xakep.ru/2021/08/24/linux-threats/

46 Такую статистику см. https://www.cvedetails.com/top-50-products.php?year=2016

47 *Ryan Schoen* «A walk through Project Zero metrics» https://googleprojectzero.blogspot.com/2022/02/a-walk-through-project-zero-metrics.html. Также об этом сказано в статье «Разработчики Linux быстрее всего исправляют ошибки в ПО» https://apptractor.ru/info/analytics/razrabotchiki-linux-bystree-vsego-

ispravlyayut-oshibki-v-po.html

48 См, например, статью «F-Droid Security Issues» https://privsec.dev/posts/android/f-droid-security-issues/

49 О важности изоляции радиомодуля сказано на этой странице https://replicant.us/freedom-privacy-security-issues.php. А также на этой https://redmine.replicant.us/projects/replicant/wiki/ModemIsolationResearch. Об этом упоминается на данной странице https://www.replicant.us/about.php#faq. Также об этом говорится здесь https://redmine.replicant.us/projects/replicant/wiki/WhatCanIDoIfMyDeviceIsNotSupported. А также здесь https://redmine.replicant.us/projects/replicant/wiki/TargetsEvaluation#Minimal-requirements. Об отсутствии изоляции радиомодуля на некоторых моделях современных смартфонов говорится на этой странице https://qna.habr.com/q/1058382

50 См., например, статью «Linux Insecurities» https://privsec.dev/posts/linux/linux-insecurities/#lack-of-application-sandboxing

51 Там же

52 См., например, статью «Linux Insecurities» https://privsec.dev/posts/linux/linux-insecurities/#lack-of-verified-boot. Также см. статью «Desktop Linux Hardening» https://privsec.dev/posts/linux/desktop-linux-hardening/#secure-boot. Об этом же говорится в статье «Как root-права и альтернативные прошивки делают ваш Android смартфон уязвимым» https://habr.com/ru/articles/541190/

53 См., например, статью «Choosing Your Android-Based Operating System» https://privsec.dev/posts/android/choosing-your-android-based-operating-system/#verified-boot

54 Это показано в данном исследовании https://atlasvpn.com/blog/linux-malware-on-a-rise-reaching-all-time-high-in-h1-2022

55 См., например, статью «Linux Insecurities» https://privsec.dev/posts/linux/linux-insecurities/#but-there-is-less-malware-on-linux

56 Это показано в данном исследовании https://atlasvpn.com/blog/linux-malware-on-a-rise-reaching-all-time-high-in-h1-2022

57 Это показано в этом исследовании https://atlasvpn.com/blog/macos-malware-development-surged-by-over-1-000-in-2020

58 Это показано в статье «Ни один из вредоносных патчей Миннесотского университета не попал в ядро Linux» https://habr.com/ru/news/556106/

59 *Мария Нефедова* «В коде Webmin более года скрывался бэкдор» https://xakep.ru/2019/08/20/webmin-backdoor/

60 Об этом говорится в статье «В межсетевых экранах Juniper обнаружен бэкдор» https://www.securitylab.ru/news/477785.php. А также в статье «Откуда взялся бэкдор в ScreenOS от Juniper Networks» https://www.securitylab.ru/news/477847.php

61 *Ричард Столмен* «Несовершенство и угнетение — не одно и то же» https://www.gnu.org/philosophy/imperfection-isnt-oppression.html. *Ричард Столмен* «Преимущества свободных программ» https://www.gnu.org/philosophy/practical.html

62 *Ричард Столмен* «Проблема — в программах, контролируемых разработчиком» https://www.gnu.org/philosophy/the-root-of-this-problem.html

63 Обоснование этого можно найти в статье «Богатые нас не спасут» https://www.pf.team/articles/bogatye-nas-ne-spasut__bUqvbxiu. А также в статье «Власть и работа» https://www.pf.team/articles/vlast%2527-i-rabota__bqjDcHiq

64 *Пол Мейсон* Посткапитализм: путеводитель по нашему будущему. — М.: Ад Маргинем Пресс, 2016. — 416 с. (с 178–180)

65 *Ричард Столмен* «Почему открытый исходный текст не передает понятия свободная программа» https://www.gnu.org/philosophy/open-source-misses-the-point.html

66 Там же

67 См., например, статью «За два дня Radiohead заработали на бесплатном альбоме пять миллионов фунтов» https://gorod.lv/novosti/60138-za_dva_dnya_radiohead_zarabotali_na_besplatnom_albome_pyat_millionov_funtov. *Cory Doctorow* «Nine Inch Nails made at least $750k from CC release in two days» https://boingboing.net/2008/03/05/nine-inch-nails-made.html. *Matt Linderman* «Jane Siberry's "you decide what feels right" pricing» https://signalvnoise.com/posts/419-jane-siberrys-you-decide-what-feels-right-pricing. *Cory Doctorow* «Monty Pyton’s free weeb video increased DVD sales by 23,000 percent» https://boingboing.net/2009/01/23/monty-pythons-free-w.html. *Mike Masnick* «The Future Of Music Business Models (And Those Who Are Already There)» https://www.techdirt.com/2010/01/25/future-music-business-models-those-who-are-already-there/

68 *Ричард Столмен* «Свобода или авторское право?» https://www.gnu.org/philosophy/freedom-or-copyright.ru.html. *Ричард Столмен* «Неверное толкование авторского права: ряд ошибок» https://www.gnu.org/philosophy/misinterpreting-copyright.ru.html

69 *Андрей Колганов* Что такое социализм? Марксистская версия. Изд. 2-е, стериотип. М.: ЛЕНАНД, 2021. — 600 с. (с. 315–317)

70 *Пол Мейсон* Указ. соч. см. в сноске 64 (с. 170–175)

71 *Джон Смит* «Империализм в XXI веке», пер. с англ. Игоря Кончаковского под ред. Дмитрия Субботина https://scepsis.net/library/id_3796.html. *Зак Коуп, Токин Лауэсен* «Империализм и трансформация стоимости в цену», пер. с англ. Павла Чикарова https://scepsis.net/library/id_3829.html. *Джеймс Хикел* «Насколько велико мировое неравенство, если по правде», пер. с англ. Дмитрия Пономаренко https://scepsis.net/library/id_3913.html. *Джеймс Хикел* «Как не следует измерять уровень неравенства», пер. с англ. Дмитрия Субботина под ред. Дмитрия Пономаренко https://scepsis.net/library/id_3914.html. *Джеймс Хикел* «Бедных в мире становится все меньше? Письмо Стивену Пинкеру (и заодно Биллу Гейтсу) о бедности в мире», пер. с англ. Дмитрия Пономаренко https://scepsis.net/library/id_3878.html. *Борис Кагарлицкий* Марксизм: Введение в социальную и политическую теорию. Изд. 3-е, стериотип. М.: ЛЕНАНД, 2021. — 320 с. (с. 148–154). *Александр Бузгалин, Андрей Колганов* Глобальный капитал. Т. 2: Теория: Глобальная гегемония капитала и ее пределы («Капитал» re-loaded). Изд. 5-е. — М.: ЛЕНАНД, 2019. — 888 с. *Андрей Колганов* «Что такое социализм?...», см. ссылку в сноске 69 (с. 300–310). В дополнение см.: *Наоми Кляйн* Доктрина шока. Расцвет капитализма катастроф. — М.: Добрая книга, 2009. — 656 с.

72 Это показано, например, в статье «Миллиардеры богатеют, остальные беднеют» https://www.pf.team/articles/milliardery-bogateiut%252c-ostal%2527nye-bedneiut__bakHtzaS. Уровень бедности показан в докладе Pew Research Center https://www.pewresearch.org/global/2015/07/08/a-global-middle-class-is-more-promise-than-reality/. На это же указывают данные Всемирного банка https://www.vsemirnyjbank.org/ru/understanding-poverty. Также статью «Мировая несправедливость и борьба с нею» https://www.pf.team/articles/mirovaia-nespravedlivost%2527-i-bor%2527ba-s-neiu__bRyeiucY. А еще статью «Вернуть и объединить» https://www.pf.team/articles/vernut%2527-i-ob%2527edinit%2527__bELlxlDo. Также «Кому выгодно?» https://www.pf.team/articles/komu-vygodno%253f__btNprDCE. Еще «Власть и работа», см. ссылку в сноске 63

73 Об этом сказано в статье «Согласно приведенным в докладе ООН данным, рост числа голодающих и сохранение проблемы неполноценного питания могут поставить под вопрос возможность ликвидации голода к 2030 году» https://www.who.int/ru/news/item/13-07-2020-as-more-go-hungry-and-malnutrition-persists-achieving-zero-hunger-by-2030-in-doubt-un-report-warns

74 Об этом говорится, например, в статье «Миллиардеры богатеют, остальные беднеют», см. ссылку в сноске 72

75 Об этом говорится в статье «Миллиардеры богатеют, остальные беднеют», см. ссылку в сноске 72. Также об этом сказано в статье «Мировая несправедливость и борьба с нею», см. ссылку в сноске 72. Еще статья «Вернуть и объединить», см. ссылку в сноске 72. Также статья «Кому выгодно?», см. ссылку в сноске 72. Еще «Власть и работа», см. ссылку в сноске 63

76 Если они не будут прибегать к методам дальнейшего наращивания прибылей, рискуют проиграть своим более беспринципным конкурентам. Это разъяснено в статье «Власть и работа», см. ссылку в сноске 63

77 Пагубное влияние существующей экономической системы на экологию показано в статье «Капитализм и разрушение планеты» https://www.pf.team/articles/kapitalizm-i-razrushenie-planety__bcEGrroT. А также в статье «Второе предупреждение ученых человечеству» https://www.pf.team/articles/vtoroe-preduprezhdenie-uchenykh-chelovechestvu__baarHlLT

78 Об этом говорят доклады МГЭИК, опирающиеся на многочисленные научные исследования https://www.ipcc.ch/languages-2/russian/publications-russian/. Вероятность того, что глобальное потепление вызвано деятельностью человека в данный момент установлена твердо https://nplus1.ru/news/2019/02/27/five-sigmas-climate-change. *Джонатан Нил* Глобальное потепление: Как остановить катастрофу? Пер. с англ. И. А. Рисмухамедова / Под ред. А. П. Белицкой. Изд. Стериотип. М.: УРСС: Книжный дом «ЛИБРОКОМ», 2019. — 288 с.

79 ВОЗ «7 миллионов смертей ежегодно связанны с загрязнением воздуха» https://www.who.int/ru/news/item/25-03-2014-7-million-premature-deaths-annually-linked-to-air-pollution

80 Об этом говорится в статье «Снижение биоразнообразия — угроза продовольствию» https://www.pf.team/articles/snizhenie-bioraznoobraziia---ugroza-prodovol%2527stviiu__bASUhxSf

81 Об этом говорится в статье «Исчерпание ресурсов и истощение почв» https://www.pf.team/articles/ischerpanie-resursov-i-istoshchenie-pochv__bflcnlul

82 *Аврутская С.Г.* «Запланированное устаревание, инновации и устойчивое развитие» // Компетентность / Competency (Russia). — 2019 г. — №7 (с. 8–16) https://web.archive.org/web/20200922223124/https://cyberleninka.ru/article/n/zaplanirovannoe-ustarevanie-innovatsii-i-ustoychivoe-razvitie/pdf

83 *Ричард Столмен* «Сколько слежки может выдержать демократия?» https://www.gnu.org/philosophy/surveillance-vs-democracy.html. Также об этом говорится в статье «Противники анонимности и их методы» https://www.pf.team/articles/protivniki-anonimnosti-i-ikh-metody__bwgaomcc. *Richard Bilton* «Camera grid to log number plates» http://news.bbc.co.uk/2/hi/programmes/whos_watching_you/8064333.stm. *Jeff Larson* «Spy Agencies Probe Angry Birds and Other Apps for Personal Data» https://www.propublica.org/article/spy-agencies-probe-angry-birds-and-other-apps-for-personal-data. *Peter Taylor* «Edward Snowden interview: 'Smartphones can be taken over'» https://www.bbc.com/news/uk-34444233. Также статья «NSA Can Spy on Smart Phone Data» https://web.archive.org/web/20180816030205/http://www.spiegel.de/international/world/privacy-scandal-nsa-can-spy-on-smart-phone-data-a-920971.html. *James Glanz, Jeff Larson, Andrew W. Lehren* «Spy Agencies Tap Data Streaming From Phone Apps» https://www.nytimes.com/2014/01/28/world/spy-agencies-scour-phone-apps-for-personal-data.html

84 Об этом говорится в статье «Microsoft помогал АНБ и ФБР шпионить за пользователями Hotmail, Skype и Outlook» https://habr.com/ru/articles/186460/. *Glenn Greenwald, Ewen MacAskill* «NSA Prism program taps in to user data of Apple, Google and others» https://www.theguardian.com/world/2013/jun/06/us-tech-giants-nsa-data. Также это показано в статье «NSA slides explain the PRISM data-collection program» https://www.washingtonpost.com/wp-srv/special/politics/prism-collection-documents/. *Jamie Hinks* «Microsoft openly offered cloud data to support NSA PRISM programme» https://web.archive.org/web/20190421070310/https://www.itproportal.com/2014/05/14/microsoft-openly-offered-cloud-data-fbi-and-nsa/. Также об этом сказано в статье «NSA Built Back Door In All Microsoft Windows Software Since 1999» https://www.marketoracle.co.uk/Article40836.html. *Glyn Moody* «How Can Any Company Ever Trust Microsoft Again?» https://web.archive.org/web/20130622044225/http://blogs.computerworlduk.com/open-enterprise/2013/06/how-can-any-company-ever-trust-microsoft-again/index.htm. *Andrew Cunningham* «New guidelines outline what iPhone data Apple can give to police» https://arstechnica.com/gadgets/2014/05/new-guidelines-outline-what-iphone-data-apple-can-give-to-police/

85 Об этом говорится в статье «NSA analysts 'wilfully violated' surveillance systems, agency admits» https://www.theguardian.com/world/2013/aug/24/nsa-analysts-abused-surveillance-systems. *Michael L. Elrick* «Misuse among police frequent, say some, but punishments rare»

https://web.archive.org/web/20160401102120/http://www.sweetliberty.org/issues/privacy/lein1.htm#.V_mKlYbb69I

86 Проект свободного процессора https://riscv.org/

87 Проект свободного Wi-Fi-адаптера https://github.com/open-sdr/openwifi

*— Запрещен на территории РФ