Широта проблематики рекомендаций по безопасности

Арс Либрев 11.12.2023

Тем, кто возмущается тому, что проект использует Internxt, поскольку он какой-то нехороший из-за CloudFlare, хочется задать вопрос, а вас не смущает, что проект также имеет и канал YouTube. Для вас CloudFlare плохо, а Google нормально? А то что вы сами пользуйтесь Telegram вас не смущает? Или Telegram с проприетарным ПО на серверах и привязкой к номеру телефона нормально? Я использую те возможности, которые у меня есть. Чтобы донести информацию до людей приходится это делать на тех площадках, которыми пользуются. А если отказываться от любого сервиса у которого есть недостатки, то в итоге пользоваться не будешь вообще ничем. Тут открывается очень важный момент — насколько вообще стоит заострять внимание на нюансах безопасности и свободы операционных систем, прошивок, программ, сервисов и т.п.

Возьмем к примеру systemd. Я сам много говорил о том, что это инструмент проблемный, он увеличивает поверхность атаки, и его стоит избегать. Но во-первых, по настоящему простых и удобных систем без него нет. Если взять Devuan, а это пожалуй наиболее удобная система из всех, в которых нет systemd, то она уступает по удобству даже Debian, не говоря о каком-нибудь Linux Mint. Во-вторых, системы без systemd имеют свои изъяны в отношении безопасности. Сравним например Devuan и Fedora. В Fedora есть systemd — поверхность атаки увеличена. Но зато в ней хорошо настроена система прав доступа, чем не может похвастаться Devuan — поверхность атаки снижена. Другой пример Whonix. В нем также есть systemd — поверхность атаки увеличена. Я предлагаю настраивать аналогичную систему на основе Devuan, где нет systemd, поверхность атаки снижена. Но в Whonix при этом реализовано множество дополнительных настроек безопасности, которые я не могу предложить для Devuan. Таким образом в Whonix поверхность атаки снижена. И простых способов реализовать эти функции в Devuan нет, их просто так туда из Whonix не перенесешь, поскольку многие из них завязаны как раз на элементах systemd.

Еще один пример. Возьмем, скажем, графическую оболочку XFCE. Она на данный момент не использует графический сервер Wayland. Разработчики двигаются в направлении его использования, но полный переход произойдет очень не скоро. Данный графический сервис, в отличии от другого более

популярного, используемого в XFCE, обеспечивает изоляцию процессов, благодаря чему вредоносный процесс не сможет получить доступ к остальным процессам, что снижает поверхность атаки. С этой точки зрения, XFCE менее безопасна, чем те оболочки, которые уже используют Wayland. С другой стороны, такие оболочки, как правило, требуют графического ускорения и имеют изобилие функционала и элементов, что увеличивает поверхность атаки. Таким образом снова все неоднозначно.

И таких вот моментов очень много. Вшитые прошивки не позволяют разработчику внедрить новый вредоносный функционал в процессе их эксплуатации пользователем. Соответственно системы с ядром Linux-libre, из которых удалены функции по подтягиванию прошивок, имеют преимущество в отношении безопасности. С другой стороны, разработчик в них также не может и закрывать уязвимости, что делает системы с ними, напротив, менее безопасными. При возможности разработчику изменять прошивку — в случае если она подтягиваемая — он имеет над пользователем больше власти. Но также пользователю проще изучать и заменить прошивку, если она подтягиваемая, чем если вшитая, таким образом у него больше свободы. Можно вспомнить историю с ноутбуком Novena, в котором изначально были несвободные прошивки. Его разработчики могли бы получить одобрение от GNU, если бы просто сделали их вшитыми. Они решили этого не делать. В итоге появились свободные прошивки, и разработчики смогли заменить старые проприетарные у пользователей через простое обновление. Если бы прошивки были вшиты, как хотело GNU, это было бы невозможно.

Еще пример все с тем же Whonix и Qubes. В версии Whonix, предполагающей использовании в обычных системах, реализованы многие функции безопасности, отсутствующие в версии для Qubes, в которой их просто не успевают своевременно реализовывать. Это снижает надежность Qubes. В то же время, в Qubes реализованы некоторые моменты безопасности, которых нет в обычных системах. Это повышает ее надежность. Таким образом снова невозможно однозначно сказать какой вариант лучше.

Все это показывает насколько все неоднозначно. Я сейчас нахожу наиболее рациональный подход в том, чтобы не гнаться за соблюдением всяких нюансов, а просто использовать наиболее удобную систему, конкретно для вас. Поскольку потребности у всех разные, рекомендации также могут быть разными. И тут мы снова приходим к тому, что есть множество нюансов. Если вам нужно это — используйте то, если это, то настройте то. Снова количество рекомендаций и работы становится чрезмерным. Вероятно лучше всего сосредоточится на наиболее простом варианте. А кто имеет возможность обеспечить что-то более серьезное разберется сам.

Что касается браузеров, то те кто критикуют рекомендации проекта по Firefox, делают вид, что его я рекомендовал без всяких оговорок и настроек. Но ведь это не так, я показывал, что его необходимо перед использованием серьезнейшим образом настроить, чтобы устранить все недостатки. Другое дело, что эти рекомендации были совершенно недостаточны. И опять же вместо того, чтобы горбатится с ними, проще просто использовать то, что не имеет таких изъянов.

Таким образом рекомендации проекта необходимо сократить по объему во много раз и полностью поменять.