Возможности применения ЭЦП в операционных системах из Реестра отечественного ПО

Дмитрий Державин, Базальт СПО, версия документа 71, .  , gost@basealt.ru.

Краткая аннотация

Вниманию читателей предлагается исследовательская работа, в ходе которой выявлен ряд проблем с применением средств ЭЦП, возникающих у пользователей Linux. В частности, выяснились следующие не вполне очевидные моменты:

- возможность получить и использовать персональный сертификат электронной цифровой подписи для доступа к государственным услугам на сегодняшний день предоставляется только за плату коммерческими организациями;
- носители данных электронной подписи, выданные конечным пользователям разными уполномоченными организациями, могут быть несовместимы как между собой, так и с порталами, предоставляющими доступ к услугам, в том числе — к государственным;
- уровень защищённости носителей данных электронной подписи, массово выдаваемых конечным пользователям, как правило, существенно снижен по отношению к доступному на сегодняшний день технологическому уровню;
- для большинства пользователей ОС из Реестра отечественного ПО механизмы ЭЦП на Едином портале государственных услуг недоступны из-за несовместимости программного обеспечения ЕПГУ и ПО уполномоченных организаций, выдающих персональные сертификаты электронной подписи;
- в некоторых случаях разработчики порталов, предоставляющих государственные услуги, рекомендуют использовать не входящие в Реестр операционные системы, а также программные средства и конфигурации, заведомо снижающие защищённость пользовательских данных.

Авторы работы рассчитывают, что полученные результаты будут полезны пользователям решений, задействующих механизмы ЭЦП, интеграторам, внедряющим соответствующие решения, а также будут приняты во внимание организациями, отвечающими за предоставление государственных информационных услуг и за реализацию конкретных механизмов инфраструктуры ЭЦП, а также разработчиками соответствующего программного и аппаратного обеспечения.

Оглавление

О чём это

Статья посвящена поддержке электронной цифровой подписи (ЭЦП) документов в ОС ALT, а также специфике применения ЭЦП в Российской федерации.

Основная задача — понять, что нужно сделать «обычному пользователю»™ — не важно, физическому или юридическому лицу, — действующему на общих основаниях, чтобы, работая в ОС из Реестра отечественного ПО, полноценно использовать возможности электронных торговых площадок и порталов государственных услуг. Под «полноценным использованием» подразумевается в первую очередь возможность аутентификации на соответствующем сайте по сертификату, размещённому на отдельном физическом носителе, и возможность электронной подписи документов, сформированных в интерфейсе сайта.

На эту тему уже проведён ряд исследовательских работ, результатом которых стало заключение о том, что, в принципе, всё работает. Здесь важно понимать, что большинство исследований совместимости с порталами государственных услуг РФ современных криптосредств, работающих под ОС Linux, проводилось в лабораторных условиях. Например, при наличии специальных договорённостей между исследователем и удостоверяющим центром, выдающим сертификат. К сожалению, по результатам таких работ о реальных возможностях лиц, действующих на общих основаниях, судить сложно.

Как всё это работает

Для того, чтобы зайти на сайт, не вводя при этом логин и пароль, а просто подсоединив к разъёму USB аппаратный токен, необходимо, чтобы в операционной системе корректно работал специализированный стек программных средств: обеспечивалась бы поддержка физических устройств, криптографических алгоритмов, программных интерфейсов, форматов и протоколов обмена данными. При этом совместимость криптографических алгоритмов, форматов и протоколов должна обеспечиваться также и за пределами области ответственности ОС — в удостоверяющем центре, выдающем сертификат, и на сайте, доступ к которому необходимо обеспечить.

А если речь идёт к тому же о сайтах государственных структур, необходимо также, чтобы используемые криптосредства были сертифицированы для соответствующего применения в Российской Федерации.

Основные механизмы

В основе технологии ЭЦП, официально применяемой в РФ, находится инфраструктура открытых ключей. Рассмотрим её работу на примере.

Для наглядности рассмотрим механизм действия универсальных алгоритмов, пригодных как для электронной подписи, так и для шифрования. К таким алгоритмам, в том числе, относятся принятый в качестве стандарта Интернет RSA и российский ГОСТ Р 34.10-94, действовавший в РФ в качестве стандарта электронной подписи до 2001 года. Более современные алгоритмы ЭЦП, к которым относятся, в том числе, действующие в настоящий момент ГОСТ Р 34.10-2001 и ГОСТ Р 34.10-2012, как правило, специализированы. Они предназначены исключительно для подписи документов, и непригодны для шифрования. Технически, отличие специализированных алгоритмов в том, что хэш в их случае не шифруется. Вместо шифрования над ним также с помощью закрытого ключа производятся другие вычисления, результат которых сохраняется в качестве подписи. При проверке подписи соответствующие комплиментарные вычисления производятся с помощью открытого ключа. Потеря универсальности в данном случае — плата за более высокую

криптостойкость. Приведённый ниже пример с универсальным алгоритмом, возможно, чуть менее актуален, но, наверняка более понятен неподготовленному читателю.

Подпись документа

Итак, для формирования электронной подписи в инфраструктуре открытых ключей применяется асимметричная схема шифрования, для которой характерно использование пары ключей. То, что зашифровано одним из этих ключей, может быть расшифровано только другим ключом пары. Один из ключей пары называется секретным, или закрытым, и хранится максимально скрытно, другой называется публичным, или открытым, и свободно распространяется — как правило, в составе сертификата. Кроме ключа в состав сертификата электронной подписи входит информация о владельце сертификата, а также подпись ключа совместно с информацией о владельце, сделанная некоей доверенной стороной. Таким образом, в составе сертификата уже находится электронная подпись, подтверждающая соответствие информации о владельце его паре ключей.

Организационно эта подпись выполняется удостоверяющим центром (УЦ) — юридическим лицом, которому делегированы полномочия устанавливать и подтверждать соответствие владельца и ключа. Устанавливается соответствие после предъявления бумажных документов, а подтверждается именно путём электронной подписи, которую удобно рассмотреть как раз на примере изготовления сертификата.

У удостоверяющего центра для подписи клиентских сертификатов тоже есть пара ключей. Подтверждённая информация о владельце сертификата в виде специальным образом оформленной таблицы объединяется в один документ с его публичным ключом. Этот документ затем проходит через два преобразования. Сначала с помощью функции хэширования документ превращается в уникальную последовательность символов фиксированной длины (хэш). Далее полученный хэш шифруется закрытым ключом удостоверяющего центра. Результат шифрования и есть собственно электронная подпись. Она прикрепляется к подписанному документу, в данном случае — информации о пользователе и его ключу, и распространяется вместе с ним. Всё это вместе — документ с информацией о пользователе и его публичным ключом, а также подпись этого документа публичным ключом УЦ, оформляется специальным образом и называется сертификатом пользователя.

Так же точно, как и в случае данных пользователя в составе сертификата, оформляется электронная подпись любого другого документа. Например, файла с заявлением на получение какой-нибудь услуги. Файл хэшируется, полученный хэш шифруется секретным ключом пользователя и прикрепляется к документу. В результате получается подписанный документ.

Проверка подписи

Как обычно происходит в случае асимметричного шифрования — то, что зашифровано одним ключом, может быть расшифровано только другим ключом пары. Так, в случае сертификата, зашифрованный хэш документа, содержащего публичный ключ пользователя и подтверждённую информацию о пользователе, может быть расшифрован с помощью открытого ключа удостоверяющего центра, который свободно распространяется в составе сертификата удостоверяющего центра. Таким образом, любой, кто получит сертификат УЦ, сможет получить из пользовательского сертификата расшифрованный хэш. Так как функция хэширования даёт уникальный результат, применив её к документу, содержащему публичный ключ пользователя и информацию о нём, можно проверить, будут ли соответствовать друг другу эти два хэша. Если будут, значит перед нами тот же самый документ, который был

подписан удостоверяющим центром, и информации, содержащейся в нём, можно верить. Если нет, значит подпись не соответствует документу, и перед нами подделка.

Ситуация с проверкой сертификата УЦ ровно такая же — он тоже подписан каким-то ключом. В итоге цепочка подписанных сертификатов заканчивается «корневым» сертификатом, который подписан сам собой. Такой сертификат называется самоподписанным. Для официально аккредитованных удостоверяющих центров Российской Федерации корневым сертификатом является сертификат Головного удостоверяющего центра Минкомсвязи.

Кроме информации о пользователе и его публичного ключа, в состав сертификата входят некоторые дополнительные данные — в частности, срок действия сертификата. Если срок действия хотя бы одного сертификата в цепочке истёк, подпись считается недействительной.

Также подпись будет считаться недействительной, если клиентский сертификат отозван удостоверяющим центром. Возможность отозвать сертификат полезна, например, в ситуациях утечки секретного ключа. Уместна аналогия с обращением в банк в случае утери банковской карты.

Услуги удостоверяющих центров

Итак, удостоверяющий центр своей подписью подтверждает соответствие некоего публичного ключа некоему набору записей. Теоретически затраты УЦ на подпись одного сертификата близки к нулю: собственно подпись представляет собой хорошо автоматизируемую, не очень затратную на современном оборудовании вычислительную операцию. При этом услуги УЦ платны. Но, в отличие от УЦ, выдающих сертификаты для сайтов, здесь деньги делаются не совсем «из воздуха».

Первое слагаемое стоимости сертификата — накладные расходы на обслуживание аккредитованного УЦ. Сюда относятся: стоимость сертификата УЦ, который тоже платный, стоимость лицензии на сертифицированное ПО, затраты на обеспечение организационных мер по защите персональных данных и так далее. Так определяется стоимость, например, персонального сертификата физического лица. Который даёт возможность подписывать локальные файлы, документы на сайтах государственных услуг и почтовые сообщения.

Второе слагаемое стоимости сертификата появляется, когда пользователь хочет с помощью своего сертификата, например, работать на электронных торговых площадках. Согласно действующему регламенту, сайт электронной торговой площадки аутентифицирует клиента при соблюдении уже двух условий: во-первых, естественно, если срок действия сертификата не истёк, сертификат не отозван и подпись его валидна. А во-вторых, если в сертификате явно указано, что он предназначен для работы на конкретной площадке. Запись об этом выглядит как обычная запись в той же таблице, в которой хранятся остальные данные сертификата. Вот за каждую такую запись в каждом пользовательском сертификате удостоверяющий центр отдаёт определённую сумму. Которую и стремится компенсировать за счёт владельца сертификата. Поэтому сертификаты, дающие возможность входа на электронные торговые площадки, сто́ят дороже.

Угрозы и атаки

В отличие от шифрования, в случае электронной подписи основная задача атакующего сводится не к получению расшифрованного текста, а к возможности подделать или произвести электронную подпись произвольного документа. То есть, условно говоря, не к расшифрованию, а к зашифрованию. Условно — потому что современные специализированные алгоритмы ЭЦП, как уже говорилось выше, не шифруют хэш документа, а выполняют над ним близкие по смыслу, но отличающиеся технически

математические операции.

По стандартам, принятым в Российской Федерации, при электронной подписи документов применяются криптографические алгоритмы, соответствующие Государственному стандарту РФ (ГОСТ Р). На момент написания данного текста (вторая половина 2016 года) ни для одного из алгоритмов, имеющих отношение к ЭЦП и имеющих в РФ статус стандарта, неизвестны способы атак, существенно отличающиеся по трудоёмкости от простого перебора. На практике это означает, что для атакующего, чей уровень доступа к вычислительным ресурсам ниже, чем у крупного государства, проще попытаться украсть ключ, чем взламывать алгоритм и подделывать подпись.

Таким образом, в случае электронной подписи основные векторы атаки направлены на получение атакующим секретного ключа. Если ключ хранится в файле на диске, украсть его можно в любой момент, получив соответствующий доступ на чтение. Например, с помощью вируса. Если ключ хранится в зашифрованном виде, получить его можно в момент применения — например, когда программа, выполняющая электронную подпись, уже расшифровала ключ и обрабатывает им данные. В данном случае задача существенно усложняется — нужно найти уязвимость в программе, или ключ придётся расшифровывать самостоятельно. Несмотря на существенное усложнение задачи, она по-прежнему остаётся вполне реальной. Для специалистов в соответствующей области она относится к категории рутинных.

Ещё более существенно усложнить задачу получения злоумышленником секретного ключа можно, разместив секретный ключ на отдельном аппаратном носителе таким образом, чтобы ключ никогда не покидал пределов этого физического устройства. В таком случае получить доступ к ключу злоумышленник сможет только похитив физическое устройство. Пропажа устройства будет сигналом для владельца о том, что ключ похищен. В любом другом случае — и это важнейший нюанс — похищение ключа может остаться незамеченным, и владелец ключа не узнает вовремя, что необходимо срочно обратиться в УЦ и отозвать действие своего сертификата.

Здесь опять таки уместна аналогия с банковской картой, которая является средством аутентификации для доступа к банковскому счёту. Пока она у владельца, он спокоен. Если карта теряется — нужно срочно её заблокировать. Задача злоумышленника при этом — не украсть карту, а незаметно сделать её копию, чтобы владелец не заблокировал доступ. Для современных аппаратных токенов способы клонирования на текущий момент неизвестны.

Токены

В данный момент общепринятый термин для обозначения отдельных физических устройств, используемых для хранения ключей электронной подписи — токен. Токены могут подключаться к компьютеру через интерфейсы USB, Bluetooth, или через специальные устройства-считыватели. Большинство современных токенов используют интерфейс USB. Но главное различие между типами токенов не в интерфейсе подключения. Токены можно разделить на два типа — те, которые используются фактически только как хранилище ключей, и те, которые «умеют» выполнять криптографические операции собственными средствами.

Токены первого типа отличаются от обычных flash-накопителей по сути только тем, что считать с них данные можно только с помощью специального программного обеспечения. В остальном — это обычное внешнее хранилище данных, и, если мы храним в нём секретный ключ, то украсть его может любой, кто получит права на доступ к устройству и будет знать, как считать с него ключ. Такие токены называются программными и практически не дают преимуществ по сравнению с хранением ключа на диске компьютера — владелец ключа так

же не может быть уверен, что ему известны все места, где хранится его секретный ключ.

Токены второго типа называются аппаратными, и основное их отличие в том, что секретный ключ является неизвлекаемым — он никогда не покидает пределов токена. Для этого на токене размещается специальный набор программного обеспечения, который активируется в момент подключения токена к компьютеру. По сути такой токен представляет собой самостоятельное вычислительное устройство со своим процессором, памятью и приложениями, которое обменивается данными с компьютером.

Секретный ключ никогда не покидает пределов аппаратного токена, потому что он генерируется прямо на самом токене. Для подписи документа в токен загружается хэш документа, прямо «на борту» токена производятся вычисления с помощью хранящегося там же секретного ключа, и готовая подпись выгружается обратно. Таким образом, зная местонахождение токена, мы всегда знаем местонахождение секретного ключа.

Одной из основных характеристик аппаратного токена является набор поддерживаемых криптографических алгоритмов. Например, если мы хотим использовать аппаратный токен для аутентификации на своём домашнем компьютере, подойдёт любой современный токен. А если мы хотим аутентифицироваться на портале Госуслуг, то необходим токен, поддерживающий сертифицированные в России криптографические алгоритмы.

Ниже приведены списки поддерживаемых криптографических механизмов для токенов eToken и JaCarta ГОСТ. Для запроса списка механизмов применена открытая утилита pkcs11-tool с параметром «-M» (от слова «mechanism»), которая может выступать в роли клиентского приложения для любой библиотеки, реализующей в её сторону интерфейс PKCS#11. В качестве библиотек PKCS#11 применены libeToken.so и libjcPKCS11.so.1 для eToken и JaCarta соответственно. Библиотека для eToken распространяется в составе ПО «SafeNet», библиотека для JaCarta доступна для загрузки с сайта компании «Аладдин Р.Д.»

```
$ pkcs11-tool --module /usr/local/lib64/libeToken.so.9.1.7 -M
Supported mechanisms:
  DES-MAC, keySize={8,8}, sign, verify
  DES-MAC-GENERAL, keySize={8,8}, sign, verify
  DES3-MAC, keySize={24,24}, sign, verify
  DES3-MAC-GENERAL, keySize={24,24}, sign, verify
  AES-MAC, keySize={16,32}, sign, verify
  AES-MAC-GENERAL, keySize={16,32}, sign, verify
  RC4, keySize={8,2048}, encrypt, decrypt
  DES-ECB, keySize={8,8}, encrypt, decrypt, wrap, unwrap
  DES-CBC, keySize={8,8}, encrypt, decrypt, wrap, unwrap
  DES-CBC-PAD, keySize={8,8}, encrypt, decrypt, wrap, unwrap
  DES3-ECB, keySize={24,24}, hw, encrypt, decrypt, wrap, unwrap
  DES3-CBC, keySize={24,24}, hw, encrypt, decrypt, wrap, unwrap
  DES3-CBC-PAD, keySize={24,24}, hw, encrypt, decrypt, wrap, unwrap
  AES-ECB, keySize={16,32}, encrypt, decrypt, wrap, unwrap
  AES-CBC, keySize={16,32}, encrypt, decrypt, wrap, unwrap
  AES-CBC-PAD, keySize={16,32}, encrypt, decrypt, wrap, unwrap
  mechtype-0x1086, keySize={16,32}, encrypt, decrypt, wrap, unwrap
  mechtype-0x1088, keySize={16,32}, encrypt, decrypt, wrap, unwrap
  RSA-PKCS-KEY-PAIR-GEN, keySize={1024,2048}, hw, generate_key_pair
  RSA-PKCS, keySize={1024,2048}, hw, encrypt, decrypt, sign, sign_recover,
verify, verify_recover, wrap, unwrap
  RSA-PKCS-OAEP, keySize={1024,2048}, hw, encrypt, decrypt, wrap, unwrap
  RSA-PKCS-PSS, keySize={1024,2048}, hw, sign, verify
  SHA1-RSA-PKCS-PSS, keySize={1024,2048}, hw, sign, verify
  mechtype-0x43, keySize={1024,2048}, hw, sign, verify
  mechtype-0x44, keySize={1024,2048}, hw, sign, verify
  mechtype-0x45, keySize={1024,2048}, hw, sign, verify
  RSA-X-509, keySize={1024,2048}, hw, encrypt, decrypt, sign, sign_recover,
```

```
verify, verify_recover, wrap, unwrap
  MD5-RSA-PKCS, keySize={1024,2048}, hw, sign, verify
  SHA1-RSA-PKCS, keySize={1024,2048}, hw, sign, verify
  SHA256-RSA-PKCS, keySize={1024,2048}, hw, sign, verify
  SHA384-RSA-PKCS, keySize={1024,2048}, hw, sign, verify
  SHA512-RSA-PKCS, keySize={1024,2048}, hw, sign, verify
  RC4-KEY-GEN, keySize={8,2048}, generate
  DES-KEY-GEN, keySize={8,8}, generate
  DES2-KEY-GEN, keySize={16,16}, generate
  DES3-KEY-GEN, keySize={24,24}, generate
  AES-KEY-GEN, keySize={16,32}, generate
  PBE-SHA1-RC4-128, keySize={128,128}, generate
  PBE-SHA1-RC4-40, keySize={40,40}, generate
  PBE-SHA1-DES3-EDE-CBC, keySize={24,24}, generate
  PBE-SHA1-DES2-EDE-CBC, keySize={16,16}, generate
  GENERIC-SECRET-KEY-GEN, keySize={8,2048}, hw, generate
  PBA-SHA1-WITH-SHA1-HMAC, keySize={160,160}, hw, generate
  PBE-MD5-DES-CBC, keySize={8,8}, generate
  PKCS5-PBKD2, generate
  MD5-HMAC-GENERAL, keySize={8,2048}, sign, verify
  MD5-HMAC, keySize={8,2048}, sign, verify
  SHA-1-HMAC-GENERAL, keySize={8,2048}, sign, verify
  SHA-1-HMAC, keySize={8,2048}, sign, verify
  mechtype-0x252, keySize={8,2048}, sign, verify
  mechtype-0x251, keySize={8,2048}, sign, verify
  mechtype-0x262, keySize={8,2048}, sign, verify
  mechtype-0x261, keySize={8,2048}, sign, verify
  mechtype-0x272, keySize={8,2048}, sign, verify
  mechtype-0x271, keySize={8,2048}, sign, verify
  MD5, digest
  SHA-1, digest
  SHA256, digest
  SHA384, digest
  SHA512, digest
  mechtype-0x80006001, keySize={24,24}, generate
$ pkcs11-tool --module /usr/local/lib64/libjcPKCS11.so.1 -M
Supported mechanisms:
  GOSTR3410-KEY-PAIR-GEN, hw, generate_key_pair
  GOSTR3410, hw, sign, verify
  GOSTR3410-WITH-GOSTR3411, hw, sign, verify
  mechtype-0x1204, hw, derive
  GOSTR3411, hw, digest
  mechtype-0x1220, generate
  mechtype-0xC4321101
  mechtype-0xC4321102
  mechtype-0xC4321103
  mechtype-0xC4321104
  mechtype-0xC4900001
```

Видно, что список поддерживаемых механизмов для eToken очень длинный, но не включает алгоритмы ГОСТ. Список поддерживаемых механизмов JaCarta включает только алгоритмы ГОСТ, но зато в объёме, необходимом для реализации функциональности ЭЦП на аппаратном токене.

Важно понимать, что современные аппаратные токены, как правило, можно использовать также и в качестве программных. То есть, на них обычно есть небольшой участок памяти, доступный извне, который при желании можно использовать для записи и хранения сгенерированного снаружи секретного ключа. Технологически в этом нет никакого смысла, но фактически такой метод используется, к сожалению, довольно широко. К сожалению, потому что зачастую владелец токена не знает, что его современный аппаратный токен, честно купленный не за символическую плату, используется как программный.

В качестве примеров исключительно программных токенов можно привести «Рутокен S» и «Рутокен Lite». В качестве примеров аппаратных токенов, не поддерживающих сертифицированные в России криптографические алгоритмы, — устройства «eToken»; в качестве поддерживающих российскую криптографию — «Рутокен ЭЦП», «JaCarta ГОСТ».

Криптопровайдеры

Программное обеспечение, предоставляющее оператору доступ к криптографическим функциям — электронной подписи, зашифрованию, расшифрованию, хэшированию — называется криптопровайдером, провайдером криптографических функций. В случае аппаратного токена криптопровайдер реализован непосредственно на токене, в случае программного токена или в случае хранения ключей на диске компьютера — в виде обычного пользовательского приложения.

С точки зрения защищённости пользовательских данных, на криптопровайдер направлен один из основных векторов атаки — именно в памяти криптопровайдера содержится в расшифрованном виде секретный ключ. В случае успешной атаки злоумышленник сможет подменить код приложения своим и снять копию секретного ключа, даже если ключ в обычном состоянии хранится в зашифрованном виде. Поэтому в случае применения криптографии для выполнения электронной подписи, имеющей юридическую силу, государство стремится оградить граждан от возможной утечки секретных ключей. Выражается это в том, что для работы с квалифицированной электронной подписью официально разрешено использовать только криптопровайдеры, имеющие соответствующий сертификат и прошедшие, следовательно, соответствующие проверки.

К сертифицированным на территории Российской Федерации для ЭЦП криптопровайдерам относятся, в частности: «Рутокен ЭЦП», «JaCarta ГОСТ», «КриптоПро CSP», «ЛИССИ-CSP», «VipNet CSP». Первые два реализованы непосредственно на аппаратных токенах, остальные — в виде пользовательских приложений. Важно понимать, что при приобретении сертифицированного в РФ аппаратного токена мы уже приобретаем сертифицированный в РФ криптопровайдер, и необходимости — технологической и юридической — в покупке ещё одного криптопровайдера нет.

Кроме набора поддерживаемых алгоритмов, криптопровайдеры различаются также набором криптографических функций — зашифрованию и расшифрованию документов, подписанию и проверке подписи, наличию графического интерфейса пользователя и так далее. Причём, из всего этого набора функций сертифицированный криптопровайдер должен выполнять только те, которые относятся непосредственно к реализации криптографических алгоритмов. Всё остальное может быть выполнено сторонним приложением. Именно так и работают криптопровайдеры на аппаратных токенах: пользовательский интерфейс реализуется сторонним приложением, не подлежащим обязательной сертификации. Приложение, реализующее пользовательский интерфейс общается с криптопровайдером на токене через другой стандартный интерфейс — PKCS#11. При этом с точки зрения пользователя работа с ключами происходит, например, непосредственно из html-браузера Firefox. На самом же деле браузер через интерфейс PKCS#11 задействует специальную программную прослойку, в которой реализованы механизмы доступа к конкретному аппаратному токену.

Кроме термина «криптопровайдер» существует ещё один близкий по смыслу термин — «средство криптографической защиты информации» (СКЗИ). Чёткого различия между этими двумя понятиями нет. Первый термин менее официальный, второй чаще используется применительно к сертифицированным техническим решениям. Так как в данном документе, в основном, описываются технологические, а не формальные аспекты, мы чаще будем пользоваться термином «криптопровайдер».

Реализация механизмов

Алгоритмы электронной подписи

В данный момент в Российской Федерации для квалифицированной электронной подписи официально разрешено использовать только два алгоритма подписи и два алгоритма хэширования. До конца 2017 года разрешено использовать ГОСТ Р 34.10-2001 совместно с алгоритмом хэширования ГОСТ Р 34.11-94. С начала 2018 года разрешено использовать только ГОСТ Р 34.10-2012 и хэши по ГОСТ Р 34.11-2012. В ситуациях, когда не требуется обязательное использование алгоритмов ГОСТ, можно пользоваться любыми доступными алгоритмами.

Например, сейчас большинство веб-сайтов, доступных по протоколу HTTP, не умеет использовать алгоритмы ГОСТ для взаимной аутентификации клиента и сервера. В случае взаимодействия с одним из таких сайтов, клиентской стороне также придётся применить алгоритмы «иностранного производства». Но, например, если хочется использовать аппаратный токен в качестве хранилища ключей для аутентификации на сайте Госуслуг, придётся выбирать токен с поддержкой ЭЦП по ГОСТ.

Необходимость применения российских алгоритмов при взаимодействии с государственными структурами обусловлена вовсе не желанием ограничить граждан в выборе. Причина в том, что государство, вложившись в разработку этих алгоритмов, отвечает за их криптостойкость и отсутствие в них недекларированных возможностей. Все стандартизированные в РФ криптографические алгоритмы неоднократно проходили независимый аудит и убедительно доказали свою состоятельность. То же самое можно сказать и по поводу многих других распространённых криптографических алгоритмов. Но, к сожалению, отсутствие в них недекларированных возможностей с такой же лёгкостью доказать нельзя. Вполне понятно, что с точки зрения государства, неразумно использовать недоверенные средства, например, для обработки персональных данных граждан.

Интерфейсы, форматы и протоколы

Для обеспечения совместимости при работе с электронной подписью выработан ряд международных стандартов, касающихся хранения данных и предоставления к ним доступа. К основным стандартам относятся:

- PC/SC — низкоуровневый интерфейс доступа к криптографическим устройствам, в том числе — программным и аппаратным токенам;
- PKCS#11 — высокоуровневый интерфейс для взаимодействия с аппаратными криптографическими модулями, можно рассматривать как унифицированный интерфейс доступа к криптопровайдерам;
- PKCS#15 — формат контейнера с ключами электронной подписи, предназначенный для хранения на физическом устройстве;
- PKCS#12, PEM — форматы контейнеров с ключами электронной подписи предназначенные для хранения в файлах, бинарных и текстовых соответственно;
- PKCS#10 — формат документа — запроса на подпись, отправляемого клиентом в УЦ для получения подписанного сертификата.

Важно понимать, что такие стандарты, как PKCS#11 или PKCS#15 первоначально разрабатывались без учёта специфики сертифицированных в Российской федерации криптосредств. Поэтому для реализации полноценной поддержки отечественной криптографии стандарты пришлось, и приходится, дорабатывать. Процесс принятия поправок к стандартам долгий, поэтому, пока доработанные стандарты не были окончательно приняты, появились их реализации, несовместимые между собой. В том числе, это касается сертифицированных в РФ криптопровайдеров. Так, сейчас все сертифицированные

криптопровайдеры имеют несовместимые между собой реализации контейнера для хранения ключей на токене. Что касается стандартов обмена данными — PKCS#10, PKCS#12, PEM — их реализации, к счастью, обычно между собой совместимы. Также, обычно не возникает разночтений в трактовке стандарта PC/SC.

Вопросами доработки стандартов, выработки рекомендаций, обеспечения совместимости в области ЭЦП в Российской Федерации в данный момент занимается специальная организация — Технический комитет по стандартизации «Криптографическая защита информации» (ТК 26), в которую входят эксперты, представители государственных структур, разработчики криптопровайдеров и другие заинтересованные лица. Об эффективности работы комитета можно спорить, но даже сам факт существования такой площадки крайне важен.

Программная часть

Программный стек для работы с электронной подписью состоит из следующих компонентов:

- реализации интерфейса для низкоуровневого доступа к хранилищам контейнеров с ключами — например, интерфейса PC/SC для доступа к физическим устройствам — токенам;
- модуля, реализующего интерфейс PKCS#11 для взаимодействия с криптопровайдером — например, выполненном на аппаратном токене;
- криптопровайдера, реализующего соответствующие криптографические алгоритмы и выполняющего действия с ними — например, электронную подпись или шифрование данных;
- пользовательского приложения, взаимодействующего с другой — по отношению к криптопровайдеру — стороны с модулем PKCS#11, и выполняющего от имени пользователя такие операции, как электронная подпись или аутентификация.

Пример стека:

- открытое программное обеспечение pcsc-lite реализует интерфейс PC/SC для взаимодействия с аппаратным токеном «Рутокен ЭЦП»;
- библиотека libcppksc11.so из состава ПО «КриптоПро CSP» обеспечивает взаимодействие с размещённым на токене контейнером, в котором хранится закрытый ключ и сертификат пользователя;
- приложение командной строки cryptcp из состава ПО «КриптоПро CSP» взаимодействует с библиотекой libcppksc11.so через интерфейс PKCS#11 и осуществляет возможность подписания документов секретным ключом пользователя; при этом функции криптопровайдера полностью реализованы в компонентах ПО «КриптоПро CSP», криптопровайдер аппаратного токена не задействуется.

Другой пример:

- открытое программное обеспечение pcsc-lite реализует интерфейс PC/SC для взаимодействия с аппаратным токеном «Рутокен ЭЦП»;
- библиотека libcppksc11.so из состава ПО «КриптоПро CSP» обеспечивает взаимодействие с размещённым на токене контейнером, в котором хранится закрытый ключ и сертификат пользователя;
- приложение «КриптоПро ЭЦП Browser plugin», работающее в составе html-браузера Firefox взаимодействует с библиотекой libcppksc11.so через интерфейс PKCS#11 и осуществляет возможность подписания документов в интерфейсе браузера на сайтах электронных торговых площадок; при этом функции криптопровайдера полностью реализованы в компонентах ПО «КриптоПро CSP», криптопровайдер аппаратного токена не задействуется.

Третий пример:

- открытое программное обеспечение pcsc-lite реализует интерфейс PC/SC для

взаимодействия с аппаратным токеном «Рутокен ЭЦП»;
- библиотека librtpksc11.so производства компании «Актив» обеспечивает взаимодействие с криптопровайдером токена;
- криптопровайдер токена осуществляет функции подписи секретным ключом передаваемых ему данных; секретный ключ при этом не покидает пределов токена;
- приложение «Рутокен Плагин», работающее в составе html-браузера Firefox взаимодействует с библиотекой libcppksc11.so через интерфейс PKCS#11 и осуществляет возможность подписания документов в интерфейсе браузера на совместимых сайтах; при этом функции криптопровайдера полностью реализованы на аппаратном токене.

Выбор конкретной реализации стека в данный момент определяется, в первую очередь, тремя факторами — предполагаемой областью применения ЭЦП, уровнем технической грамотности пользователя и готовностью удостоверяющего центра работать с запросом на подпись сертификата.

Кроме перечисленного выше набора программного обеспечения, работающего на стороне пользователя, существует ещё два приложения, особенности которых необходимо учитывать. Первое — ПО, обслуживающее удостоверяющий центр. Второе — ПО, с которым мы хотим быть совместимыми. Например, сайт Госуслуг. Или ПО для подписи электронных документов.

Если удостоверяющий центр, в котором мы планируем получить сертификат, не готов работать с запросами на подпись в формате PKCS#10 и умеет работать только с программными токенами (то есть, воспринимает любой токен как программный), у нас не остаётся выбора. Как правило, в этом случае ПО УЦ сгенерирует для нас пару ключей, запишет её на токен, тут же сгенерирует на базе открытого ключа и наших персональных данных запрос на подпись, подпишет его, и сохранит сертификат на токене. Сертификат и ключ при этом будут находиться в контейнере закрытого формата, известного только разработчику ПО УЦ. Соответственно, для доступа к контейнеру с ключами придётся покупать криптопровайдер того же разработчика. И область применения ЭЦП будет ограничена возможностями ПО одного конкретного разработчика. Покупать в этом случае аппаратный токен смысла нет — можно обойтись программным. Токен в этом случае обязательно придётся нести в УЦ.

Если удостоверяющий центр, в котором мы планируем получить сертификат, готов работать с запросами на подпись в формате PKCS#10, то нам не важно, какое именно ПО используется в этом УЦ. Мы сможем использовать у себя тот криптопровайдер, который совместим с нашими целевыми приложениями. Например, с электронными торговыми площадками или сайтом Госуслуг. Нести токен в УЦ в этом случае не нужно, достаточно сгенерировать запрос на подпись и предъявить его там вместе со своими бумажными документами; получить сертификат, и самостоятельно сохранить его на токене с помощью выбранного криптопровайдера.

К сожалению, очень немногие УЦ в данный момент (конец 2016 года) готовы работать с запросом на подпись. Отчасти такая ситуация обусловлена недостаточным уровнем технической подготовки пользователей, которые не в состоянии проследить за тем, чтобы запрос на подпись был оформлен надлежащим образом — с указанием всех необходимых атрибутов и их значений. На решение этой проблемы направлено, в том числе, данное руководство.

Аппаратная часть

Среди представленных на российском рынке токенов можно отметить следующие:

Программные токены: Рутокен Lite, Рутокен S.

Аппаратные носители с поддержкой российской криптографии: Рутокен ЭЦП, JaCarta ГОСТ, MS_KEY K.

Рассмотрим в качестве примера списки поддерживаемых криптографических механизмов аппаратного Рутокен ЭЦП и программного Рутокен_Lite:

```
$ pkcs11-tool --module /usr/local/lib64/librtpkcs11ecp.so -M
Supported mechanisms:
  RSA-PKCS-KEY-PAIR-GEN, keySize={512,2048}, hw, generate_key_pair
  RSA-PKCS, keySize={512,2048}, hw, encrypt, decrypt, sign, verify
  RSA-PKCS-OAEP, keySize={512,2048}, hw, encrypt, decrypt
  MD5, digest
  SHA-1, digest
  GOSTR3410-KEY-PAIR-GEN, hw, generate_key_pair
  GOSTR3410, hw, sign, verify
  mechtype-0x1204, hw, derive
  GOSTR3411, hw, digest
  GOSTR3410-WITH-GOSTR3411, hw, digest, sign
  mechtype-0x1224, hw, wrap, unwrap
  mechtype-0x1221, hw, encrypt, decrypt
  mechtype-0x1222, hw, encrypt, decrypt
  mechtype-0x1220, hw, generate
  mechtype-0x1223, hw, sign, verify

$ pkcs11-tool --module /usr/local/lib64/librtpkcs11ecp.so -M
Supported mechanisms:
```

Как видим, списки поддерживаемых криптографических механизмов Рутокен Lite пуст, в отличие от Рутокен ЭЦП, который включает алгоритмы ГОСТ и RSA.

К аппаратным токенам без поддержки российской криптографии, как уже упоминалось выше, относятся, в частности, JaCarta PKI и eToken.

Учитывая относительно невысокие цены на аппаратные токены с поддержкой российской криптографии, можно с уверенностью рекомендовать использовать именно их. Кроме явного преимущества в виде неизвлекаемого секретного ключа, аппаратный токен ещё и включает в себя сертифицированный криптопровайдер. То есть, существует возможность организовать работу с токеном таким образом, что не потребуется дополнительно покупать дорогостоящее ПО.

Из недавних разработок хотелось бы отметить Рутокен ЭЦП 2.0 с поддержкой стандарта ГОСТ Р 34.10-2012.

Применение

Инструменты пользователя

Открытые инструменты

Открытое программное обеспечение на текущий момент позволяет реализовать практически полный программный стек для работы с ЭЦП.

PCSC lite

Реализация PC/SC, разработанная в рамках проекта PCSC lite, является эталонной для ОС семейства Linux. Она входит в состав любого из рассматриваемых в настоящем документе

вариантов программного стека для работы с ЭЦП. В дальнейшем, если конкретный вариант реализации не указан, будем считать её задействованной по умолчанию.

OpenSC

Библиотека, реализующая интерфейс PKCS#11, а также набор прикладных инструментов, задействующих её функциональность, была разработана в рамках проекта OpenSC. Инструментарием поддерживается множество аппаратных токенов, среди которых Рутокен ЭЦП, поддержка которого была добавлена специалистами компании «Актив», разрабатывающей токены семейства Рутокен.

С помощью утилит OpenSC можно выполнить на аппаратном токене, в частности, следующие действия:

- инициализировать токен — сбросить настройки к первоначальному состоянию;
- установить PIN-коды администратора и пользователя (если поддерживаются);
- сгенерировать пару ключей (если поддерживается библиотекой PKCS#11);
- импортировать на токен сертификат, подписанный удостоверяющим центром.

Библиотека PKCS#11 из комплекта OpenSC позволяет выполнять на поддерживаемых токенах полный набор операций с электронной подписью. К поддерживаемым токенам относится также Рутокен ЭЦП.

Здесь важно понимать, что для Рутокен ЭЦП существует два разных варианта поддержки программным обеспечением, не совместимых между собой по формату хранения ключей на токене. При использовании библиотеки PKCS#11 из состава OpenSC мы можем пользоваться открытым стеком ПО, а при использовании бесплатно распространяемой закрытой библиотеки производства компании «Актив», — закрытым стеком «Актива».

OpenSSL

Чтобы полноценно использовать возможности ЭЦП, кроме собственно библиотеки PKCS#11 должны быть реализованы пользовательские приложения, предоставляющие оператору доступ к функциям библиотеки и возможностям токена. Ярким примером такой реализации является открытое ПО из проекта OpenSSL. В нём поддерживаются, в частности, следующие функции:

- зашифрование данных;
- расшифрование данных;
- подпись документа;
- проверка подписи;
- формирование запроса на подпись сертификата;
- импорт сертификата;
- экспорт сертификата.

Кроме того, с помощью OpenSSL можно реализовать функциональность полноценного удостоверяющего центра, в том числе:

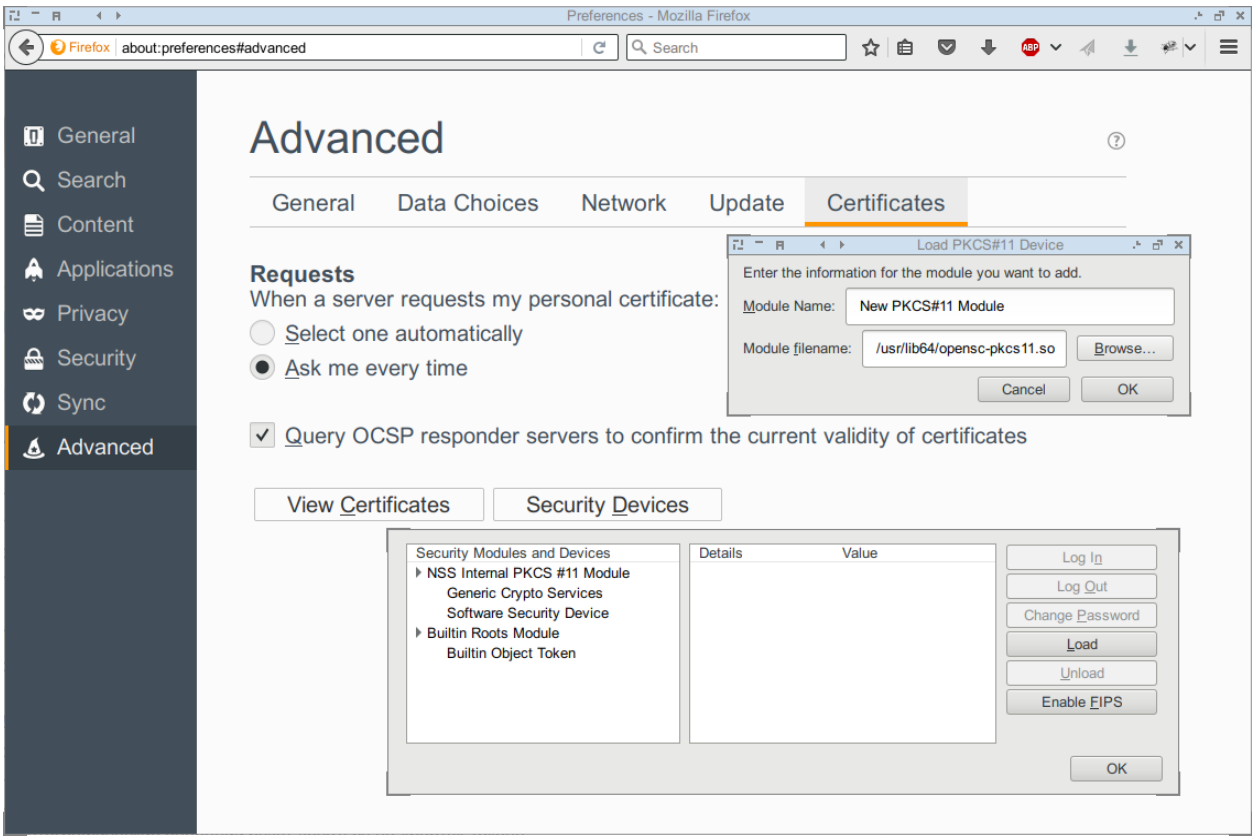
- выдачу клиентских сертификатов путём подписания запросов на подпись в формате PKCS#10;
- отзыв клиентских сертификатов;
- учёт выданных и отозванных сертификатов.

Единственная на текущий момент недоработка OpenSSL, не позволяющая пока реализовать полнофункциональный вариант программного стека ЭЦП на базе открытого ПО — отсутствие открытого модуля для взаимодействия с библиотеками PKCS#11 с поддержкой алгоритмов ГОСТ. Существует закрытая реализация такого модуля, выполненная в компании «Актив», но она не входит в базовую поставку OpenSSL, и поэтому с выходом новых версий OpenSSL совместимость периодически нарушается. Открытая реализация этого модуля пока

не поддерживает алгоритмы ГОСТ.

Firefox

Кроме OpenSSL взаимодействовать с библиотеками PKCS#11 умеет также всем известный html-браузер Firefox. Для подключения библиотеки PKCS#11 нужно зайти в меню управления настройками «Preferences», далее — в вертикальном списке слева выбрать «Advanced», в горизонтальном списке выбрать «Certificates», нажать кнопку «Security Devices», в появившемся диалоговом окне нажать кнопку «Load». Появится ещё одно окно с возможностью выбора пути к файлу с библиотекой PKCS#11 и возможностью ввода локального имени для этой конкретной библиотеки. Можно таким образом загрузить несколько разных модулей PKCS#11 для разных типов физических и виртуальных устройств.



К сожалению, функциональности Firefox пока недостаточно для подписи документов в интерфейсе веб-сайта. Поэтому для полноценного открытого программного стека ЭЦП с поддержкой ГОСТ не хватает ещё подключаемого модуля (плагина), позволяющего обращаться к объектам на токене из ПО сайта. Надеемся, что в ближайшее время такой плагин будет написан. Или открыт.

КриптоПро

Компания «КриптоПро» в настоящий момент является крупнейшим игроком на рынке ПО для ЭЦП в России. Среди её разработок отметим криптопровайдер «КриптоПро CSP», ПО для обеспечения работы УЦ «КриптоПро УЦ» и браузерный плагин «Cades плагин». Популярность решений КриптоПро обусловлена поддержкой со стороны удостоверяющих центров и сайтов электронных торговых площадок.

КриптоПро CSP

Самый популярный сертифицированный криптопровайдер. Поддерживает программные (или аппаратные в режиме программных) токены, инициализированные с помощью него же или с помощью ПО КриптоПро УЦ. Библиотека PKCS#11 из состава КриптоПро CSP поддерживается ПО «Cades плагин» — популярным браузерным плагином. Три перечисленных программных компонента вместе формируют стек, поддерживающий полный набор функций ЭЦП. Закрытое проприетарное ПО, платная лицензия.

Поддерживает практически все продававшиеся и продающиеся на территории РФ программные и аппаратные токены. Правда, в режиме программных — с потенциально извлекаемым закрытым ключом. Потенциально — потому что сообщения об успешных атаках с целью получения закрытого ключа на актуальную версию КриптоПро CSP не публиковались.

Версия под Linux имеет интерфейс командной строки с нестандартным (несовместимым с POSIX.2) синтаксисом. Поддерживается большинство современных дистрибутивов, в том числе ALT.

В версиях КриптоПро CSP до 4.0 включительно применяется ручное управление локальными кэшами для хранения пользовательских сертификатов и сертификатов УЦ. Для полноценной работы с пользовательским сертификатом необходимо, чтобы его копия лежала в одном локальном кэше, а полная цепочка сертификатов УЦ до корневого включительно — в другом. Технологически эта особенность КриптоПро, строго говоря, не вполне обоснована: цепочку имеет смысл проверять при аутентификации; собственно факт валидности сертификата на возможность подписи не влияет. Тем более, если это наш собственный сертификат и мы знаем, откуда он взялся. В свежих на момент написания статьи бета-версиях КриптоПро CSP, по словам разработчиков, реализована автоматическая загрузка цепочки сертификатов УЦ. Но за тем, чтобы в локальном кэше пользовательских сертификатов находился только тот, который в данный момент используется, похоже, всё ещё приходится следить вручную.

Сторонние разработчики предпринимают попытки написания графических интерфейсов к КриптоПро CSP, облегчающие проведение рутинных пользовательских операций. Примером такой утилиты может служить rosa-crypto-tool, автоматизирующий подписание и зашифрование документов. Пакет доступен в дистрибутивах ALT.

КриптоПро УЦ

Самое популярное в России ПО для удостоверяющих центров. Для хранения ключей создаёт на токене контейнер собственного закрытого формата. Для работы с контейнером необходимо ПО КриптоПро CSP. Таким образом, после инициализации с помощью КриптоПро УЦ любой современный токен превращается в программный, поддерживаемый единственным совместимым криптопровайдером. Закрытым, требующим покупки лицензии.

Cades плагин

Единственный браузерный плагин для работы с ЭЦП в Linux, поддерживаемый электронными торговыми площадками. Использует для доступа к криптопровайдеру библиотеку PKCS#11 из состава КриптоПро CSP. Соответственно, поддерживает только токены, инициализированные КриптоПро CSP или КриптоПро УЦ.

VipNet

Другой популярный набор из совместимых друг с другом сертифицированного криптопровайдера и ПО УЦ. Умеет работать с аппаратными токенами, но не даёт

преимуществ в части поддержки электронных торговых площадок и сайтов государственных услуг. Версия под Linux пока находится на стадии бета-тестирования и не является сертифицированным средством криптографической защиты. Что, кстати, в случае применения аппаратных токенов несущественно.

Инструментарий производства компании «Актив»

Производитель токенов «Рутокен», компания «Актив», помимо большого вклада в открытое программное обеспечение, также разработала собственный полнофункциональный стек программных средств для поддержки ЭЦП, включающий в себя:

- библиотеку PKCS#11, предоставляющую доступ ко всем возможностям токена;
- уже упоминавшийся модуль реализации интерфейса PKCS#11 для OpenSSL с поддержкой ГОСТ;
- «Рутокен Плагин» для html-браузера, в полном объёме поддерживающий функциональность, предоставляемую библиотекой PKCS#11 собственной разработки;
- «Центр регистрации» — набор скриптов, выполняющихся на стороне браузера и работающих через «Рутокен Плагин», позволяющих генерировать ключевые па́ры на аппаратных токенах семейства «Рутокен», а также генерировать запрос на подпись на основе публичного ключа;
- библиотеку PKCS#11, распространяющуюся в составе браузерного плагина «IFCPlugin», предоставляющего возможность в том числе использовать токены семейства «Рутокен» для аутентификации и подписи документов в интерфейсе портала государственных услуг РФ.

Таким образом, компания «Актив» бесплатно предоставляет владельцам токенов «Рутокен» полный набор ПО для работы с электронной подписью.

Кроме того, «Актив» совместно с компанией «КриптоПро» выпускает отдельную версию ПО «КриптоПро CSP», совместимую с аппаратными возможностями токенов «Рутокен». В описании этой версии декларируется полная совместимость с другими продуктами КриптоПро. На практике это означает возможность использовать аппаратные возможности Рутокенов на электронных торговых площадках, поддерживающих Cades плагин.

Инструментарий производства компании «Аладдин Р.Д.»

Компания «Аладдин Р.Д.» — второй производитель сертифицированных в России аппаратных токенов, поддерживающих алгоритмы ГОСТ. Возможности собственного программного обеспечения под Linux ограничиваются графической утилитой на базе ПО SafeNet, позволяющей инициализировать токен и поменять PIN-коды. Библиотека PKCS#11 собственной разработки предоставляет доступ ко всем возможностям аппаратного токена, но, к сожалению через утилиту pkcs11-tool из состава ПО OpenSC бо́льшая часть этих возможностей недоступна. Например, чтобы создать на аппаратном токене ключевую пару, сейчас необходимо пользоваться сторонним ПО типа VipNet CSP или найти в официальном комплекте разработчика (SDK) примеры на языке C, адаптировать их, откомпилировать и запустить. Так же, изучая примеры из SDK, можно сгенерировать запрос на подпись в формате PKCS#10. Таким образом, теоретическая возможность обойтись без покупки дополнительного сертифицированного криптопровайдера можно. На практике эту возможность ещё предстоит проверить.

Инструментарий производства компании «Лисси СОФТ»

Компания «Лисси СОФТ» — ещё один производитель ПО, предлагающий полный набор инструментов для работы с электронной подписью. Кроме ПО для УЦ и традиционных программных криптопровайдеров в арсенале программных средств «Лисси СОФТ» есть ряд

уникальных технических решений. Например, криптопровайдеры для хранения программных токенов в облаке.

Для ПО «Лисси СОФТ» характерно строгое соблюдение стандартов и технических спецификаций. Криптопровайдеры, в том числе уникальные, оформлены в виде библиотек PKCS#11. По заявлениям разработчиков, они хорошо стыкуются с открытым программным обеспечением, также ориентированным на поддержку стандартов. Например, с html-браузерами и утилитами из комплекта OpenSSL.

Применительно к организации доступа к сайтам государственных услуг продукты «Лисси СОФТ» также представляют интерес. В первую очередь — совместимостью с «IFCPlugin» — браузерным плагином, реализующим механизмы ЭЦП на портале Госуслуг. Во-вторых — возможностью ПО удостоверяющих центров работать с запросами на подпись сертификата в формате PKCS#10.

Браузеры

Иногда, для доступа к сайтам, на которых мы планируем применять механизмы электронной подписи, приходится задействовать другие технологии, использующие российскую криптографию. Например, алгоритмы ГОСТ могут использоваться для организации шифрованного канала передачи данных. В этом случае необходима поддержка соответствующих алгоритмов в браузере. К сожалению, официальные сборки Firefox и Chromium пока не поддерживают российскую криптографию в полной мере. Поэтому приходится пользоваться альтернативными сборками. Такие сборки есть, например, в арсенале криптосредств компаний «КриптоПро» и «Лисси СОФТ», а также, конечно, в дистрибутивах Альт.

Сайты

Среди сайтов, для работы с которыми необходимо применение технологий электронной подписи, в первую очередь нас интересуют сайты, предоставляющие государственные услуги, а так же электронные торговые площадки (ЭТП). К сожалению, некоторые ЭТП не поддерживают пока работу с ОС из Реестра отечественного ПО. Но ситуация постепенно меняется в лучшую сторону.

Применение ЭЦП для веб сайтов обычно сводится к аутентификации и подписи документов, сформированных в интерфейсе сайта. Принципиально аутентификация выглядит так же, как и подпись любого другого документа: сайт, на котором клиент хочет аутентифицироваться, генерирует последовательность символов, которую отправляет клиенту. Клиент отправляет обратно выполненную с помощью его секретного ключа подпись этой последовательности и свой сертификат (сертификат — чуть раньше). Сайт берёт из клиентского сертификата публичный ключ и проверяет подпись оригинальной последовательности. То же происходит и при подписи документов. Здесь вместо произвольной последовательности выступает собственно документ.

Госуслуги

Основная функциональность портала государственных услуг, доступная с помощью механизмов ЭЦП — возможность подписи документов, сформированных в интерфейсе сайта. Так, например, можно через веб-форму заполнить заявление, тут же его подписать и передать в работу. Кроме того, ЭЦП позволяет аутентифицироваться на сайте без предъявления логина и пароля, только по сертификату на токене.

За взаимодействие с программными и аппаратными токенами со стороны портала отвечает разработанное Ростелекомом ПО «IFCPlugin», имеющее в своём составе ряд библиотек

PKCS#11 для доступа к заранее определённым типам токенов, и позволяющее подключать сторонние библиотеки PKCS#11 для доступа новым неизвестным устройствам. Некоторые библиотеки — например, Лисси — подключить удаётся. Некоторые — например, КриптоПро — нет.

Достоверно известно, что работу с порталом госуслуг под ОС семейства Linux можно организовать сейчас одним из двух способов. Один способ описан разработчиками «Лисси СОФТ», другой — разработчиками компании «Актив». Оба способа предполагают наличие специальных договорённостей с удостоверяющим центром. В первом случае придётся ехать за токеном в УЦ Лисси в город Юбилейный Московской области. Во втором случае — найти УЦ, который выдаст клиентский сертификат на основе запроса на подпись в формате PKCS#10. Первый вариант может оказаться проще. И — да — УЦ Лисси СОФТ не работает с запросами на подпись удалённо.

Вообще, если удастся найти удостоверяющий центр, готовый работать с запросами на подпись, можно попробовать вместо Рутокен использовать, например, JaCarta ГОСТ. Или использовать Рутокен, но не с ПО компании Актив, а с открытым ПО. Но поддержка таких решений со стороны IFCPlugin весьма сомнительна. Работа с Рутокенами поддерживается в IFCPlugin нативно, поддержка реализована разработчиками компании «Актив». Поддержка токенов Лисси реализована путём подключения сторонних библиотек PKCS#11 через конфигурационный файл IFCPlugin.

Что касается поддержки самого распространённого варианта токенов — программных (или аппаратных в режиме программных) токенов, обслуживаемых ПО КриптоПро, она в настоящий момент в IFCPlugin под Linux не работает. В полученном 19 декабря 2016 года официальном ответе Ростелекома, в частности, указано: «Просим рассмотреть возможность работы Плагина и ЭЦП с порталом через MSIE 10 или 11 на виртуализированной или хостовой ОС Windows XP/7/8/10, при обязательной установке сертификата в личное хранилище сертификатов в системе». То есть, если прийти в первый попавшийся УЦ со своим токеном с просьбой оформить на нём ключ и сертификат электронной подписи, почти наверняка зайти с таким токеном на сайт Госуслуг под Linux не получится.

Торговые площадки

Так же, как и в случае портала государственных услуг, для электронных торговых площадок механизмы электронной цифровой подписи дают возможность аутентифицироваться на сайте по сертификату на токене и — главное — подписывать документы, сформированные в интерфейсе сайта. То есть, например, размещать заявки и участвовать в аукционах. Для большинства ЭТП эта функциональность является основной. То есть, фактически, без исправно функционирующих на стороне клиента механизмов ЭЦП на электронной торговой площадке делать нечего.

В конце 2016 года компанией «Базальт СПО», г. Москва, при участии специалистов компаний «Бизнес-ИНФО», г. Санкт-Петербург, и «НТЦ Галэкс», г. Барнаул, было проведено исследование пяти федеральных электронных торговых площадок на предмет совместимости с ОС из Реестра российских программ.

В результате исследования выяснилось, что по состоянию на 14 декабря 2016 года большинство федеральных торговых площадок — «Единая электронная торговая площадка», «РТС-тендер» и «Сбербанк-АСТ» готовы к использованию на рабочих местах под управлением ОС семейства Linux. Оставшиеся две площадки пока не обеспечивают даже возможность входа по клиентскому сертификату. Планируются ли их разработчиками какие-то меры по обеспечению совместимости, к сожалению, пока выяснить не удалось.

При этом в официальных инструкциях всех площадок кроме «Единой электронной торговой площадки» в качестве единственной поддерживаемой указана операционная система Windows. Официальные ответы служб технической поддержки подтверждают эту информацию. Инструкция на сайте «Единой электронной торговой площадки» описывает настройку браузера без указания конкретной операционной системы.

В обобщённом виде результаты исследования приведены в таблице:

Электронная торговая площадка	**Возможность входа по сертификату пользователя**	**Возможность подписи документа в интерфейсе площадки**	**Документация по работе с площадкой под ОС из Реестра**
Сбербанк — Автоматизированная система торгов	Да	Да	Нет
Единая электронная торговая площадка	Да	Да	Да
Агентство по государственному заказу, инвестиционной деятельности и межрегиональным связям Республики Татарстан	Нет	Не проверялась	Нет
РТС-тендер	Да	Да	Нет
ММВБ — Информационные технологии	Нет	Не проверялась	Нет

Для ЭТП, обеспечивающих совместимость с Linux, единственным поддерживаемым криптосредством в данный момент является Cades plugin, который умеет использовать в качестве криптопровайдера только КриптоПро CSP. Таким образом, хорошая новость заключается в том, что получить токен для доступа к электронным торговым площадкам очень просто — большинство УЦ выдают именно их. Плохие новости — токен будет программным и не будет совместим с сайтом Госуслуг.

Для остальных торговых площадок пока единственным средством доступа к функциональности ЭЦП является компонент ОС Windows под названием CAPICOM. Специалистами компании «Этерсофт» проведена исследовательская работа, в результате которой выяснена теоретическая возможность запуска CAPICOM в среде Wine.

Прочие сайты

Помимо перечисленных сайтов, использующих ЭЦП непосредственно — для входа на сайт и подписи документов, сформированных в интерфейсе сайта, существует ряд площадок, предоставляющих возможность загрузки документов, предварительно подписанных квалифицированной электронной подписью. В качестве примера можно привести сайт Главного радиочастотного центра. Доступ на сайт осуществляется по логину и паролю, а документы готовятся и подписываются заранее — в интерфейсе пользовательской ОС. Таким образом, для работы с такими сайтами нужна только функциональность локальной подписи документов. То есть, ограничений по выбору криптопровайдера в данном случае практически нет.

Пример готового решения

К сожалению, на текущий момент авторам данного документа неизвестен способ применения одного и того же токена одновременно на сайте Госуслуг и на электронных торговых площадках. Поэтому придётся сделать два токена с двумя парами ключей и двумя сертификатами соответственно. Законодательно это не запрещено, технически — осуществимо. Финансово тоже вполне реально: одним токеном будет, например, аппаратный Рутокен ЭЦП, другим — какая-нибудь старая модель eToken, которую сейчас вполне можно найти за символическую плату.

Доступ к сайту Госуслуг

Для доступа к Госуслугам возьмём Рутокен ЭЦП и выполним следующие действия:

1. загрузим ПО «Рутокен плагин» со страницы по ссылке;
2. установим Рутокен плагин — скопируем файлы плагина (npCryptoPlugin.so и librtpkcs11ecp.so) в ~/.mozilla/plugins/;
3. зайдём на сайт с ПО «Центр регистрации» и по инструкции выполним следующие действия — проинициализируем токен, сгенерируем пару ключей, сгенерируем и сохраним локально файл с запросом на подпись в формате PKCS#10;
4. обратимся в УЦ, готовый выдать нам сертификат по запросу на подпись, получим сертификат в виде файла;
5. в интерфейсе «Центра регистрации» сохраним сертификат из полученного файла на токене;
6. по ссылке загрузим пакет формата «deb» — файл IFCPlugin-x86_64.deb;
7. с помощью ПО «Midnight Commander» (команда mc) «зайдём» в файл пакета как в каталог;
8. скопируем содержимое каталога CONTENTS/usr/lib/mozilla/plugins в локальный каталог ~/.mozilla/plugins;
9. в браузере Firefox на сайте Госуслуг пройдём последовательно по ссылкам «Войти» и «Войти с помощью электронных средств».

Основная проблема в осуществлении этой инструкции — найти удостоверяющий центр, готовый работать с запросом на подпись.

Доступ к электронным торговым площадкам

Для доступа к сайтам электронных торговых площадок, поддерживающих работу в Linux, выполним следующие действия:

1. с любым официально продававшимся в РФ токеном обратимся в любой удостоверяющий центр, использующий по «КриптоПро УЦ», и получим сохранённые на токене пользовательский сертификат и секретный ключ в контейнере формата, поддерживаемого ПО КриптоПро;
2. в соответствии с инструкцией установим ПО «CryptoPro CSP» и «Cades plugin» для браузера Chromium;
3. с помощью браузера Chromium зайдём на сайты электронных торговых площадок и начнём работу с ними согласно официальным инструкциям.

Возможность подписи документов в виде файлов будет доступна через консольные утилиты КриптоПро и через сторонние приложения-обёртки типа уже упоминавшегося rosa-crypto-tool.