Максим Левин

PGP Кодирование и шифрование информации с открытым ключом

Москва

Литературное агентство «Бук-Пресс»
2006

ББК 32.973
УДК 681.3

Корректура и верстка
Ирина Царик

Максим Левин
Л80 PGP: Кодирование и шифрование информации с открытым ключом. — М.: Бук-пресс, 2006. — 166 с.

Руководство пользователя PGP. PGP использует шифрование с открытым ключом для защиты файлов данных и электронной почты. С ее помощью вы сможете надежно связаться с людьми, которых вы никогда не встречали, без использования специальных каналов связи для предварительного обмена ключами. Программа PGP обладает многими полезными качествами, работает быстро, позволяет осуществлять сложные манипуляции с ключами, реализует электронные подписи, позволяет сжимать данные и хорошо эргономически спроектирована.

ООО «Литературное агентство «Бук-Пресс».
127591, Москва, Керамический пр., д. 53. кор. 1.
http://www.book-press.ru

Криптографическая система

Pretty Good Privacy (PGP) выпущено фирмой Phil's Pretty Good Software и является криптографической системой с высокой степенью секретности для операционных систем MS-DOS, Unix, VAX/VMS и других. PGP позволяет пользователям обмениваться файлами или сообщениями с использованием функций секретности, установлением подлинности, и высокой степенью удобства. Секретность означает, что прочесть сообщение сможет только тот, кому оно адресовано. Установление подлинности позволяет установить, что сообщение, полученное от какого-либо человека было послано именно им. Нет необходимости использовать секретные каналы связи, что делает PGP простым в использовании программным обеспечением. Это связано с тем, что PGP базируется на мощной новой технологии, которая называется шифрованием с "открытым ключом".

PGP объединяет в себе удобство использования криптографической системы с открытым ключом Rivest-Shamir-Adleman (RSA) и скорость обычной криптографической системы, алгоритм "дайджеста сообщений" для реализации электронной подписи, упаковку данных перед шифрованием, хороший эргономический дизайн программы и развитую систему управления ключами. PGP выполняет функции открытого ключа быстрее, чем большинство других аналогичных реализаций этого алгоритма.

PGP — это криптографическая система с открытым ключом для всех.

PGP не имеет встроенной возможность работы с модемом. Для этого вы должны использовать отдельное программное обеспечение.

Зачем Вам нужна PGP?

Вы можете планировать политическую кампанию, обсуждать ваши налоги, или заниматься разными незаконными делами. Или вы можете делать что-либо, чувствуя, что это не должно быть запрещено, однако является таковым. Что бы это ни было, вы не желаете, чтобы ваше личное электронное сообщение (электронной почтой) или конфиденциальные документы были прочтены кем либо еще, кроме адресата. Нет ничего некорректного в том, что вы хотите сохранить в тайне свою информацию.

Возможно вы думаете что ваша Электронная Почта законна настолько, что шифрование совершенно не оправдано. Если вы действительно столь законопослушный гражданин, которому нечего скрывать, тогда почему вы не всегда посылаете ваши письма на открытках? Почему не проходите проверку на наркотики по первому требованию? Почему требуете ордер, если полиция делает обыск в вашей квартире? Не пытаетесь ли вы скрыть что-нибудь? Вы, очевидно, террорист или торговец наркотиками, раз вы скрываете ваше письмо внутри конверта. Или у вас мания преследования, не иначе. Необходимо ли законопослушному гражданину шифровать его электронную почту?

Что если бы каждый считал, что законопослушные граждане должны использовать открытки для их писем? Тогда те смелые натуры, которые попытались бы утвердить свои права на секретность с помощью конвертов, навели бы на себя подозрение. При этом, возможно, власти вскрывали бы их конверты, дабы увидеть, что они там скрывают. К счастью, мы не живем в таком мире, так как каждый защищает большинство своей корреспонденции конвертами, и поэтому никто никого не подозревает, если тот использует конверт. Аналогично, было бы хорошо, если бы все использовали шифрование для своей электронной почты, будь они виновны перед законом, или нет, так, чтобы никто не подозревал ни в чем тех, кто использует шифрование. Подумайте об этом, как о форме солидарности.

Сегодня, если правительство хочет нарушить право на секретность обычного гражданина, оно должно понести определенные затраты, затратить труд, чтобы пресечь, вскрыть и прочитать письменное сообщение, прослушать и, возможно, записать телефонный разговор. Такой вид контроля с большими трудозатратами неприменим в крупных масштабах, это выполнимо только в особых, важных случаях, когда такие трудозатраты оправданы.

Все больший и больший процент нашей частной связи распространяется через электронные каналы. Электронное сообщение будет постепенно заменять обычное бумажное сообщение. И все бы хорошо, но сообщения в электронной почте слишком хорошо доступны для просмотра всем на предмет поиска интересующих ключевых слов. Это может быть выполнено легко, автоматически, выполняться постоянно и быть трудно обнаруживаемым. Международные телеграммы

NSA уже просматривает таким образом в крупных масштабах.

Мы движемся к тому времени, когда страны будут пересечены волоконно-оптическими сетями передачи данных, связывающими вместе все наши все более и более вездесущие персональные компьютеры. Электронная почта будет обычным делом для каждого, а не новшеством, которым она является сегодня. Возможно правительство будет осуществлять защиту нашей электронной почты своими специально разработанными алгоритмами шифрования. Вероятно, большинство людей будут доверять этому. Но, возможно, кто-то будет предпочитать свои собственные защитные меры.

Билль Сената № 266, заключал в себе некоторые интересные мероприятия. Если бы эта резолюция стала реальным законом, это вынудило бы изготовителей оборудования для секретной связи вставлять специальные "люки" в их изделия, так что правительство могло бы читать шифрованные сообщения кого угодно. Читаем: "Конгресс считает, что поставщики электронных услуг связи и изготовители сервисного оборудования для электронной связи будут обеспечивать, чтобы системы связи разрешали правительству получать простое текстовое содержимое разговора, данных, и других видов связи, соответственно санкционированных законом." Эта мера была отвергнута после решительного гражданского протеста и протеста групп представителей промышленности. Но правительство тогда представило другое законодательство, чтобы достичь подобных целей.

Если секретность вне закона, то только люди вне закона будут ею обладать. Секретные агентства имеют

доступ к хорошей криптографической технологии. Такой же технологией пользуются перевозчики оружия и наркотиков. Так поступают нефтяные компании и другие корпорации-гиганты. Но обычные люди и массовые организации в своем большинстве не имели доступа к криптографическим технологиям "военного уровня" с использованием открытого ключа. До сих пор.

PGP позволяет людям взять секретность в собственные руки. Сейчас налицо возрастастающая социальная потребность для этого.

Как PGP работает

Это поможет вам, если вы уже были знакомы с концепцией криптографии вообще, и криптографии с открытым ключом в частности. Тем не менее, есть небольшое количество вводных замечаний относительно криптографии с открытым ключом.

Сначала немного терминологии. Предположим, что я хочу послать вам сообщение, но не хочу, чтобы кто-либо, кроме вас, имел возможность его прочитать. Я могу "шифровать" или "кодировать" сообщение, это означает, что я закодирую его весьма сложным образом, что обеспечит невозможность прочтения для кого угодно, кроме вас, адресата. Я беру криптографический "ключ" для шифровки сообщения, и вы тоже должны использовать тот же самый ключ, чтобы декодировать или "расшифровать" его. По крайней мере, так это работает в стандартной криптографической системе с одним ключом.

В стандартных криптографических системах, таких, как US Federal Data Encryption Standart (DES), один и тот

же ключ используется и для шифрования, и для расшифровки. Это значит что ключ должен первоначально быть передан через секретные каналы так, чтобы обе стороны могли иметь его до того, как шифрованные сообщения будут посылаться по обычным каналам. Это может быть неудобно. Если вы имеете секретный канал для обмена ключами, тогда вообще зачем вам нужна криптография?

В криптографической системе с открытым ключом каждый имеет два связанных взаимно однозначно ключа: публикуемый открытый ключ и секретный ключ. Каждый из них дешифрует код, сделанный с помощью другого. Знание открытого ключа не позволяет вам вычислить соответствующий секретный ключ. Открытый ключ может публиковаться и широко распространяться через коммуникационные сети. Такой протокол обеспечивает секретность без необходимости использовать специальные каналы связи, необходимые для стандартных криптографических систем.

Кто угодно может использовать открытый ключ получателя, чтобы зашифровать сообщение ему, а получатель использует его собственный соответствующий секретный ключ для расшифровки сообщения. Никто, кроме получателя, не может расшифровать его, потому что никто больше не имеет доступа к секретному ключу. Даже тот, кто шифровал сообщение, не будет иметь возможности расшифровать его.

Кроме того, обеспечивается также установление подлинности сообщения. Собственный секретный ключ отправителя может быть использован для шифровки сообщения, таким образом "подписывая" его. Так

создается электронная подпись сообщения, которую получатель (или кто-либо еще) может проверять, используя открытый ключ отправителя для расшифровки. Это доказывает, что отправителем был действительно создатель сообщения и что сообщение впоследствии не изменялось кем-либо, так как отправитель — единственный, кто обладает секретным ключом, с помощью которого была создана подпись. Подделка подписанного сообщения невозможна, и отправитель не может впоследствии изменить свою подпись.

Эти два процесса могут быть объединены для обеспечения и секретности, и установления подлинности: сначала подписывается сообщение вашим собственным секретным ключом, а потом шифруется уже подписанное сообщение открытым ключом получателя. Получатель делает наоборот: расшифровывает сообщение с помощью собственного секретного ключа, а затем проверяет подпись с помощью вашего открытого ключа. Эти шаги выполняются автоматически с помощью программного обеспечения получателя.

В связи с тем, что алгоритм шифрования с открытым ключом значительно медленнее, чем стандартное шифрование с одним ключом, шифрование сообщения лучше выполнять с использованием высококачественного быстрого стандартного алгоритма шифрования с одним ключом. Первоначальное незашифрованное сообщение называется "открытым текстом" (или просто текст). В процессе, невидимом для пользователя, временный произвольный ключ, созданный только для этого одного "сеанса", используется для традиционного шифрования файла открытого текста. Тогда открытый ключ получателя используется только для шифровки этого временного

произвольного стандартного ключа. Этот зашифрованный ключ "сеанса" посылается наряду с зашифрованным текстом (называемым "ciphertext" — "зашифрованный") получателю. Получатель использует свой собственный секретный ключ, чтобы восстановить этот временный ключ сеанса, и затем применяет его для выполнения быстрого стандартного алгоритма декодирования с одним ключом, чтобы декодировать все зашифрованное сообщение.

Открытые ключи хранятся в виде "сертификатов ключей", которые включают в себя идентификатор пользователя владельца ключа (обычно это имя пользователя), временную метку, которая указывает время генерации пары ключей, и собственно ключи. Сертификаты открытых ключей содержат открытые ключи, а сертификаты секретных ключей — секретные. Каждый секретный ключ также шифруется с отдельным паролем. Файл ключей, или каталог ключей ("кольцо с ключами" — "keyring") содержит один или несколько таких сертификатов. В каталогах открытых ключей хранятся сертификаты открытых ключей, а в каталогах секретных — сертификаты секретных ключей.

На ключи также внутренне ссылаются "идентификаторы ключей", которые являются "сокращением" открытого ключа (самые младшие 64 бита большого открытого ключа). Когда этот идентификатор ключа отображается, то показываются лишь младшие 24 бита для краткости. Если несколько ключей могут одновременно использовать один и тот же идентификатор пользователя, то никакие два ключа не могут использовать один и тот же идентификатор ключа.

PGP использует "дайджесты сообщений" для формирования подписи. Дайджест сообщения — это криптографически мощная 128-битная односторонняя хэш-функция от сообщения. Она несколько напоминает контрольную сумму, или CRC-код, она однозначно представляет сообщение и может использоваться для обнаружения изменений в сообщении. В отличие от CRC-кода (контроля циклическим избыточным кодом), дайджест не позволяет создать два сообщения с одинаковым дайджестом. Дайджест сообщения шифруется секретным ключом для создания электронной подписи сообщения.

Документы подписываются посредством добавления перед ними удостоверяющей подписи, которая содержит идентификатор ключа, использованного для подписи, подписанный секретным ключом дайджест сообщения и метку даты и времени, когда подпись была сгенерирована. Идентификатор ключа используется получателем сообщения, чтобы найти открытый ключ для проверки подписи. Программное обеспечение получателя автоматически ищет открытый ключ отправителя и идентификатор пользователя в каталоге открытых ключей получателя.

Шифрованным файлам предшествует идентификатор открытого ключа, который был использован для их шифрования. Получатель использует этот идентификатор для поиска секретного ключа, необходимого для расшифровки сообщения. Программное обеспечение получателя автоматически ищет требуемый для расшифровки секретный ключ в каталоге секретных ключей получателя.

Эти два типа каталогов ключей и есть главный метод сохранения и работы с открытыми и секретными ключами. Вместо того, чтобы хранить индивидуальные ключи в отдельных файлах ключей, они собираются в каталогах ключей для облегчения автоматического поиска ключей либо по идентификатору ключа, либо по идентификатору пользователя. Каждый пользователь хранит свою собственную пару каталогов ключей. Индивидуальный открытый ключ временно хранится в отдельном файле, достаточно большом для посылки его вашим друзьям, которые добавят его в свои каталоги ключей.

Установка PGP

Версия PGP 2.2 для MSDOS распространяется в виде архивного файла с именем **PGP22.ZIP** (каждая новая версия будет иметь имя вида **PGPxy.ZIP** для PGP с номером версии x.y). Этот архив нужно распаковать с помощью утилиты PKUNZIP для MSDOS (распространяемой как shareware) или утилиты для Unix **unzip**. Пакет с очередной версией PGP содержит файл README.DOC, который вам нужно прочитать перед установкой PGP. Этот файл содержит последние новости о том, что нового в данной версии PGP, а также о том, что находится в остальных файлах архива.

Если у вас уже есть версия 1.0 PGP для MS-DOS, вы можете удалить ее, так как никто более не использует ее. Если вы не хотите ее удалять, переименуйте старый исполняемый файл в pgp1.exe для предотвращения конфликта имен файлов.

Для установки PGP на вашей системе MSDOS, вы должны просто скопировать архив PGPxx.ZIP в подходящий каталог на вашем жестком диске (например, C:\PGP), и распаковать его с помощью PKUNZIP. Для более эффективной работы с PGP, вам также будет необходимо изменить ваш файл AUTOEXEC.BAT, как описано далее в этом руководстве, но вы можете сделать это позже, после того, как немного повозитесь с PGP и прочитаете это руководство. Если вы ранее не использовали PGP, первым шагом после установки (и прочтения настоящего руководства) будет запуск команды PGP для генерации ключей **pgp -kg**.

Установка в системах Unix и VAX/VMS в основном похожа на установку в MS-DOS, но для начала вам необходим исходный текст для компиляции. Для этого вариант с исходными текстами для Unix дополняется make-файлом.

Как использовать PGP

Вывод краткой справки

Для получения краткой справки об использовании команды PGP введите:

pgp -h

Шифрование Сообщения

Для шифровки текстового файла открытым ключом получателя, введите:

pgp -e textfile her_userid

В результате будет получен зашифрованный файл textfile.pgp.

Типичный пример:

pgp -e letter.txt Alice

или:

pgp -e letter.txt "Alice S"

В первом примере PGP будет искать в каталоге ваших открытых ключей "pubring.pgp" сертификаты открытых ключей, которые содержат строку "Alice" в поле идентификатора пользователя. Во втором примере PGP будет искать идентификаторы пользователя, которые содержат строку "Alice S". Нельзя использовать пробелы в командной строке, если вы не заключаете целую строку в кавычки. Поиск ведется без учета регистра. Если соответствующий открытый ключ найден, то он используется для шифровки текстового файла "letter.txt", получается зашифрованный файл "letter.pgp".

PGP будет пытаться упаковывать текст перед шифрованием, таким образом значительно усиливая его сопротивляемость криптоанализу. Это означает, что зашифрованный файл скорее всего будет меньше, чем исходный текстовый файл.

Если вы хотите послать полученное шифрованное сообщение через каналы электронной почты, преобразуйте его в печатаемый ASCII-формат Radix-64 с помощью добавления опции -a, как описано ниже.

Шифрование сообщения для нескольких адресатов

Если вы хотите послать одно и то же сообщение более чем одному человеку, вы можете задать шифрование

для нескольких адресатов, любой из которых может его расшифровать. Для задания нескольких адресатов просто добавьте несколько идентификаторов пользователей в командную строку, примерно так:

pgp -e letter.txt Alice Bob Carol

В результате будет создан зашифрованный файл letter.pgp, который может быть прочитан любым адресатом: Alice, Bob или Carol. Можно задать любое количество адресатов.

Подписание сообщения

Для подписания текстового файла вашим секретным ключом, наберите:

pgp -s textfile [-u your_userid]

Обратите внимание, что скобки [] просто обозначают необязательное поле, не вводите сами скобки.

В результате выполнения этой команды получится подписанный файл textfile.pgp. Типичный пример:

pgp -s letter.txt -u Bob

По этой команде PGP будет искать в файле каталога секретных ключей "secring.pgp" сертификаты секретного ключа, в которых содержится строка "Bob" в поле идентификатора пользователя. Поиск ведется без учета регистра. Если соответствующий секретный ключ будет найден, он будет использован для подписания текстового файла "letter.txt", в результате будет получен подписанный файл "letter.pgp".

Если вы не указываете поле идентификатора пользователя, то в качестве ключа про умолчанию будет

использован первый секретный ключ из каталога секретных ключей.

Подписание и шифрование

Для подписания текстового файла вашим секретным ключом и последующей его зашифровки открытым ключом адресата, наберите:

pgp -es textfile her_userid [-u your_userid]

Обратите внимание, что скобки [] просто обозначают необязательное поле, не вводите сами скобки.

В результате выполнения данного примера будет получен "вложенный" зашифрованный файл textfile.pgp. Ваш секретный ключ для создания подписи автоматически ищется в вашем каталоге секретных ключей по your_userid. Открытый ключ адресата для шифрования автоматически ищется в вашем каталоге открытых ключей по her_userid. Если вы опускаете этот параметр в командной строке, PGP запросит его у вас.

Если вы опускаете второй параметр, то для подписания по умолчанию будет использован первый ключ из вашего каталога секретных ключей.

Обратите внимание, что PGP будет пытаться упаковывать текст перед шифрованием.

Если вы хотите послать полученное шифрованное сообщение через каналы электронной почты, преобразуйте его в печатаемый ASCII-формат Radix-64 с помощью добавления опции -a, как описано ниже.

Можно определить несколько адресатов, задавая в командной строке несколько идентификаторов пользователя.

Использование стандартного шифрования

Иногда вам необходимо зашифровать файл традиционным способом, с помощью шифрования с одним ключом. Это может быть полезно для защиты файлов в архиве, которые будут сохраняться но не будут посылаться кому — нибудь. Так как расшифровывать файл будет тот же самый человек, который и шифровал его, шифрование с открытым ключом действительно в этом случае не является необходимым.

Чтобы зашифровать текстовый файл традиционным способом, наберите:

pgp -c textfile

В этом случае PGP зашифрует текстовый файл и получит выходной файл textfile.pgp без использования метода открытого ключа, каталогов ключей, идентификаторов и т.д. PGP запросит у вас фразу пароля для шифрования файла. Эта фраза не должна быть (действительно НЕ ДОЛЖНА БЫТЬ) одинаковой с фразой пароля, которую вы используете для защиты вашего секретного ключа. PGP будет пытаться упаковать текст перед шифрованием.

PGP не будет шифровать один и тот же файл тем же самый способом дважды, даже если вы используете ту же самую фразу пароля.

Дешифровка и проверка подписей

Для дешифровки зашифрованного файла или проверки целостности подписи подписанного файла используется команда:

pgp ciphertextfile [-o plaintextfile]

Обратите внимание, что скобки [] просто обозначают необязательное поле, не вводите сами скобки.

По умолчанию для шифрованного файла принимается расширение ".pgp". Необязательное имя выходного текстового файла определяет, где размещать обработанный текстовый файл. Если никакое имя не задается, то используется имя шифрованного файла без расширения. Если подпись находится внутри шифрованного файла, то производится дешифровка и проверка целостности подписи. На экран будет выведен полный идентификатор пользователя, подписавшего текст.

Обратите внимание, что расшифровка файла ведется полностью автоматически, независимо от того, только подписан ли он, только зашифрован, или и то и другое. PGP использует префикс идентификатора ключа из зашифрованного файла для автоматического поиска соответствующего секретного ключа для расшифровки в каталоге секретных ключей. Если в файле имеется вложенная подпись, PGP будет использовать затем префикс идентификатора из вложенной подписи для автоматического нахождения соответствующего общего ключа в вашем каталоге открытых ключей, чтобы проверить подпись. Если в ваших каталогах ключей уже имеются все верные ключи, то вмешательство пользователя не требуется, за исключением того, что PGP

запросит у вас пароль для вашего секретного ключа, если это необходимо. Если файл шифровался традиционным способом без использования общего ключа, PGP определит это и запросит у вас фразу пароля для дешифровки.

Работа с ключами

Со времен Юлия Цезаря работа с ключами всегда была наиболее трудной частью криптографии. Одной из принципиально выдающихся особенностей PGP является сложная работа с ключами.

Генерация ключа RSA

Для генерации вашей собственной уникальной пары открытый/секретный ключ заданного размера, наберите:

pgp -kg

PGP покажет вам меню рекомендуемых размеров ключа (простой уровень, коммерческий уровень или военный уровень) и запросит требуемый размер ключа (около тысячи бит). Чем длиннее ключ, тем выше степень секретности, но платить за это придется скоростью.

PGP также запросит идентификатор пользователя, что означает ваше имя. Хорошая мысль — использовать ваше имя как идентификатор пользователя, так как впоследствии меньше риск того, что другой человек использует неверный открытый ключ для шифровки сообщения, адресованного вам. В идентификаторе пользователя допускаются пробелы и знаки пунктуации.

Это поможет вам в том случае, если вы помещаете ваш адрес в электронной почте в <угловые скобки> после вашего имени, например:

Robert M. Smith <rms@xyzcorp.com>

Если вы не имеете адреса электронной почты, используйте ваш номер телефона или любую другую уникальную информацию, которая поможет гарантировать, что ваш идентификатор пользователя уникален.

PGP также запросит "фразу пароля" для защиты вашего секретного ключа на случай, если он попадет в чужие руки. Никто не сможет использовать ваш файл секретного ключа без этой фразы пароля. Фраза пароля — это обычный пароль, за исключением того, что это может быть целая фраза или предложение с большим количеством слов, пробелов, знаков пунктуации, или любых других символов. Не потеряйте эту фразу пароля, так как нет никакого способа восстановить ее при утрате. Эта фраза пароля будет необходима каждый раз при использовании вашего секретного ключа. Фраза учитывает регистр, и не должна быть слишком короткой или простой настолько, что ее можно было бы предположить. Она никогда не отображается на экране. Не оставляйте ее в записанном виде нигде, где кто-либо может ее увидеть и не храните ее на вашем компьютере. Если вы не хотите использовать фразу пароля (и тогда вы просто дурак!), просто нажмите ENTER в ответ на запрос PGP.

Пара открытый/секретный ключ — это производная от множества действительно случайных чисел, полученных путем измерения интервалов времени между вашими нажатиями клавиш быстрым таймером.

Имейте ввиду, что генерация ключей RSA — ОЧЕНЬ длительный процесс. На это может потребоваться от нескольких секунд для маленького ключа на быстром процессоре, до нескольких минут для большого ключа на старой IBM PC/XT.

Сгенерированная пара ключей будет помещена в ваши каталоги открытых и секретных ключей. Вы можете позже использовать опцию -kx для извлечения (копирования) вашего нового открытого ключа из вашего каталога открытых ключей и помещать его в отдельный файл открытого ключа, который уже будет пригоден для распространения среди ваших друзей. Файл открытого ключа может посылаться вашим друзьям для включения в их каталоги открытых ключей. Конечно, вы храните ваш файл секретного ключа у себя, и вы должны включать его в ваш каталог секретных ключей. Каждый секретный ключ в каталоге защищен своей собственной фразой пароля.

Никогда не передавайте ваш секретный ключ кому-либо другому. По той же причине, не делайте пары ключей для своих друзей. Каждый должен делать их собственноручно. Всегда сохраняйте физический контроль за вашим секретным ключом и не рискуйте "засветить" его, храня на удаленном компьютере, храните его только на вашем персональном компьютере.

Добавление ключа в ваш каталог ключей

Для добавления содержимого файла открытого или секретного ключа в ваш каталог открытых или секретных ключей наберите (помните, что [скобки] обозначают необязательный параметр):

Pgp -ka keyfile [keyring]

Для keyfile по умолчанию берется расширение ".pgp". Необязательные имена файлов каталогов keyring по умолчанию имеют значения "pubring.pgp" или "secring.pgp" в зависимости от того, содержит ли файл открытый или секретный ключ. Вы можете задавать и другие имена файлов с расширением по умолчанию ".pgp".

Если ключ уже есть в вашем каталоге, PGP не будет добавлять его снова. Все ключи из keyfile будут добавлены в каталог, кроме дубликатов. Если добавляемый ключ имеет прикрепленную подпись, удостоверяющую его, подпись будет добавлена в каталог вместе с ключом. Если ключ уже находится в вашем каталоге, PGP будет добавлять только новые удостоверяющие подписи для ключа в вашем каталоге, если он их еще не имеет.

Удаление ключа из вашего каталога ключей

Для удаления ключа из вашего каталога открытых ключей наберите:

pgp -kr userid [keyring]

PGP будет искать заданный идентификатор пользователя в вашем каталоге открытых ключей и, при нахождении оного, удалит его. Не забудьте, что любого фрагмента идентификатора пользователя будет вполне достаточно для установления соответствия. В качестве необязательного имени файла каталога ключей принимается по умолчанию "pubring.pgp". Оно может быть опущено, либо вы можете задать имя "secring.pgp" для удаления секретного ключа. Вы можете определять

различные имена файлов каталогов ключей. По умолчанию для имени файла принимается расширение ".pgp".

Если для заданного ключа существует больше одного идентификатора пользователя, будет задан вопрос о необходимости удаления только заданного вами идентификатора, при этом ключ и остальные идентификаторы будут сохранены неизменными.

Извлечение (копирование) ключа из вашего каталога ключей

Для извлечения (копирования) ключа из вашего каталога открытых или секретных ключей, наберите:

Pgp -kx userid keyfile [keyring]

PGP просто скопирует ключ, заданный идентификатором пользователя, из вашего каталога открытых или секретных ключей в заданный файл ключа. Эту возможность можно использовать, если вы хотите передать копию вашего открытого ключа кому-либо.

Если ключ имеет любые удостоверяющие подписи, присоединенные к нему в вашем каталоге ключей, они будут скопированы наряду с ключом.

Если вы хотите получить извлеченный ключ в виде печатаемых символов ASCII для пересылки по E-mail, используйте опции -kxa.

Просмотр содержания вашего каталога ключей

Для просмотра содержания вашего каталога открытых ключей наберите:

pgp -kv[v] [userid] [keyring]

По этой команде на экран будет выводиться список всех ключей из каталога, у которых хотя бы частично совпадает идентификатор пользователя с параметром userid в командной строке. Если этот параметр опущен, выводятся все ключи из каталога. В качестве имени файла каталога ключей keyring по умолчанию принимается "pubring.pgp". Оно может быть опущено, либо вы можете задать "secring.pgp" для просмотра оглавления каталога секретных ключей. Кроме того, вы можете определить любое другое имя файла каталога. Расширение имени файла по умолчанию ".pgp".

Для того, чтобы увидеть все удостоверяющие подписи, связанные с каждым ключом, используйте опцию -kvv:

pgp -kvv [userid] [keyring]

Если вы хотите задать отдельное имя файла каталога ключей, но хотите увидеть все ключи из него, попробуйте использовать такой альтернативный способ:

pgp keyfile

Если не заданы опции командной строки, PGP выведет список всех ключей в keyfile.pgp, и также будет пытаться добавлять их в ваш каталог ключей, если их там еще нет.

Как защищать открытые ключи от подделки

В криптографической системе с открытым ключом вы не должны защищать открытые ключи от взлома. Фактически, лучше, чтобы они широко распространялись. Но важно зашищать их от подделки, то есть, вы должны быть уверены, что если вам кажется, что данный ключ принадлежит определенному человеку, то он именно ему и принадлежит. Это является наиболее уязвимым местом криптографии с открытым ключом. Давайте сначала рассмотрим потенциальное несчастье, а затем, как избежать этого с помощью PGP.

Предположим, что вы хотите послать частное сообщение Алисе. Вы списываете открытый удостоверенный ключ Алисы с BBS ("электронная доска объявлений"). Вы шифруете ваше письмо к Алисе с помощью этого открытого ключа и посылаете его к ней через электронную почту BBS.

К сожалению, без вашего и Алисы ведома, другой пользователь с именем Чарли "отфильтровал" эту BBS и сгенерировал свой собственный открытый ключ с присоединенным идентификатором пользователя Алисы. Он скрытно подставляет свой поддельный ключ вместо настоящего открытого ключа Алисы. Вы, ничего не подозревая, используете этот поддельный ключ, принадлежащий Чарли, вместо ключа Алисы. Все проходит нормально, так как этот поддельный ключ имеет идентификатор пользователя Алисы. Теперь Чарли может декодировать сообщение, предназначенное Алисе, потому что он имеет соответствующий секретный ключ. Он может даже снова зашифровать декодированное сообщение с

настоящим открытым ключом Алисы и послать это ей, так что никто и не заподозрит никакого подвоха. Кроме того, он может даже делать вполне достоверные подписи Алисы с этим секретным ключом, потому что все будут использовать поддельный открытый ключ для проверки подписи Алисы.

Единственый способ предотвращать это бедствие заключается в том, чтобы предотвратить возможность подделки кем-либо открытого ключа. Если вы получили открытый ключ Алисы прямо от нее, то нет проблем. Но это может быть затруднительно, если Алиса находится за тысячи миль или вообще сейчас недостижима.

Возможно, вы смогли бы получить открытый ключ Алисы от вашего общего друга Давида, который знает, что у него есть достоверная копия открытого ключа Алисы, и которому вы оба доверяете. Давид может подписать открытый ключ Алисы, удостоверяя его целостность. Давид может создать эту подпись с помощью своего собственного секретного ключа.

Таким образом будет создано подписанное удостоверение открытого ключа, и будет показано, что ключ Алисы не был подделан. Но для этого требуется, чтобы вы имели заведомо верную копию открытого ключа Давида для проверки его подписи. При этом возможно, чтобы Давид обеспечил также и Алису подписанной копией вашего открытого ключа. Таким образом, Давид служит как бы "поручителем" между вами и Алисой.

Это подписанное удостоверение открытого ключа для Алисы может быть передано Давидом или Алисой на BBS, и вы можете списать его позже. Тогда вы сможете проверять подпись с помощью открытого ключа Давида и,

таким образом, быть уверенным, что это действительно открытый ключ Алисы. Никакой мошенник не сможет одурачить вас так, чтобы вы приняли его собственный поддельный ключ за ключ Алисы, потому что никто другой не может подделывать подписи, сделанные Давидом.

Пользующийся доверием человек может даже специализироваться в области "рекомендации" пользователей друг другу посредством удостоверения их открытых ключей своей подписью. Этот доверенный человек мог бы расцениваться как "сервер" ("ключник") или как "Удостоверяющий Авторство". Любым удостоверениям открытого ключа, обладающим подписью такого сервера можно вполне доверять, как действительно принадлежащим тому, кто в них указан. Все пользователи, кто хочет участвовать в этом, будут нуждаться в заведомо верной копии только открытого ключа сервера, чтобы его подписи могли бы быть проверены.

Доверенный централизованный сервер ключей или Удостоверяющий Авторство наиболее подходит для большой безликой центрально управляемой корпорации или правительственных учреждений.

Для более децентрализованных массовых "партизанских" использований, разрешение всем пользователям действовать как доверенные поручители для их друзей, вероятно, сработает лучше, чем центральный сервер ключей. PGP настроена, чтобы подчеркивать этот органически децентрализованный подход.

Он лучше отражает естественный способ взаимодействия людей на персональном социальном

уровне, и позволяет людям лучше выбирать, кому они могут доверять управление ключами.

Все это дело защиты открытых ключей от подделки — наиболее трудная проблема в программных средствах, использующих принцип открытого ключа. Это ахиллесова пята криптографии с открытым ключом, и некоторая сложность программ связана с решением именно этой проблемы.

Вы должны использовать открытый ключ только после того, как вы уверены, что он вполне достоверен, не был подделан и действительно принадлежит тому человеку, который на это претендует. Вы можете быть в этом уверены, если вы получили открытый ключ непосредственно от его владельца, либо если этот ключ имеет подпись кого-либо из тех, кому вы доверяете и от которого вы уже получили достоверный открытый ключ. Кроме того, идентификатор пользователя должен содержать полное имя и фамилию владельца, а не только имя.

Неважно, как вас соблазнят — а вас будут соблазнять — никогда , НИКОГДА не признавайте целесообразность и не доверяйте открытому ключу, который вы переписали с BBS, если он не подписан кем-либо, кому вы доверяете. Такой никем не удостоверенный ключ вполне мог быть поделан кем-то, возможно даже администратором BBS.

Если вас попросят подписать чей-либо сертификат открытого ключа, убедитесь, что этот ключ действительно принадлежит человеку, чье имя указано в идентификаторе пользователя удостоверения этого ключа. Это необходимо потому, что ваша подпись на этом удостоверении — ваше

личное утверждение принадлежности этого ключа. Другие люди, доверяющие вам, будут принимать этот ключ как достоверный, потому что он подписан вами. Плохой совет — полагаться на слухи; не подписывайте открытый ключ, пока не получите независимую информацию из первых рук о принадлежности этого ключа. Предпочтительно подписывать только после получения ключа прямо от автора.

Если вы удостоверяете открытый ключ, вы должны быть уверены в принадлежности этого ключа в гораздо большей степени, чем при простом шифровании сообщения этим ключом. Чтобы быть убежденным в достоверности ключа настолько, чтобы использовать его, ключ должен иметь удостоверяющие подписи от надежных поручителей. Однако, для того, чтобы самому удостоверить этот ключ, вы должны знать о действительном владельце этого ключа из первых рук. К примеру, вы могли бы позвонить ему по телефону и прочитать ему файл ключа для окончательного подтверждения того факта, что этот ключ — его, при этом будучи уверенным, что разговариваете с тем, кем надо.

Имейте в виду, что ваша подпись на удостоверении открытого ключа не подтверждает честность человека, а только утверждает целостность (монопольное использование) открытого ключа этого человека. Вы не рискуете вашим авторитетом при подписании открытого ключа человека с социально-патологическими изменениями, если вы были полностью уверены, что ключ действительно принадлежал ему. Другие люди будут верить, что ключ принадлежит ему, потому что вы подписали его (если они доверяют вам), но они не будут доверять владельцу ключа.

Достоверный ключ не то же самое, что доверие к его владельцу.

Доверие не обязательно передается; у меня есть друг, я ему доверяю и считаю, что он говорит правду. Он — доверчивый человек, который доверяет Президенту и считает, что тот говорит правду. Это отнюдь не значит, что я доверяю Президенту и считаю, что он говорит правду. Это просто здравый смысл. Если я доверяю подписи Алисы на ключе, а она, в свою очередь, доверяет подписи Чарли на ключе, это не означает, что я должен доверять подписи Чарли на ключе.

Хорошая идея — хранить ваш собственный открытый ключ у себя вместе с коллекцией удостоверяющих подписей, которыми ваш ключ подписали "поручители" в надежде, что большинство людей будут доверять по крайней мере одному из поручителей, которые удостоверяют правильность вашего открытого ключа. Вы могли бы посылать ваш ключ с коллекцией удостоверяющих подписей на различные BBS. Если вы подписываете кому-либо его открытый ключ, возвращайте ему его со своей подписью, он затем добавит ее к своей собственной коллекции "верительных грамот" для своего открытого ключа.

PGP следит, какие ключи в вашем каталоге открытых ключей правильно удостоверяются подписями поручителей, которым вы доверяете. Все, что вы должны делать — сообщать PGP, каким людям вы доверяете, как поручителям, и удостоверять их ключи у себя с помощью вашей собственной абсолютно достоверной подписи. PGP может брать ее отсюда, автоматически при утверждении любых других ключей, которые были подписаны вашими

указанными поручителями. И, конечно, вы можете прямо подписывать большинство ключей самостоятельно. Подробнее об этом ниже.

Удостоверьтесь, что никто другой не может подделать ваш собственный каталог открытых ключей. Проверка нового подписанного удостоверения открытого ключа должна в конечном счете зависеть от целостности достоверных открытых ключей, которые уже имеются в вашем собственном каталоге открытых ключей. Поддерживайте физический контроль за вашим каталогом открытых ключей, предпочтительно на вашем собственном персональном компьютере, нежели на удаленной системе с разделением времени, так же, как вы будете делать это для вашего секретного ключа. Это должно защищать его от подделки, но не от раскрытия. Храните достоверную резервную копию вашего каталога открытых ключей и вашего каталога секретных ключей на защищенных от записи носителях.

Так как ваш собственный достоверный открытый ключ используется как последняя инстанция для прямого или косвенного удостоверения всех остальных ключей в каталоге, этот ключ является наиболее важным для защиты от подделки. Для обнаружения любой подделки вашего собственного абсолютно достоверного открытого ключа PGP может быть сконфигурирована таким образом, чтобы автоматически сравнивать ваш открытый ключ с резервной копией на защищенном от записи носителе. Для получения более подробной информации см. описание команды "-kc" в части "Специальные Разделы".

PGP, в принципе, считает, что вы будете поддерживать физическую защиту вашей системы и ваших

каталогов ключей, столь же хорошо, сколь и собственно вашу копию PGP. Если злоумышленник имеет доступ к вашему диску, тогда, теоретически, он может изменить собственно PGP.

Один несколько сложный способ защиты вашего собственного каталога открытых ключей от вмешательства заключается в подписывании всего каталога вашим секретным ключом. Вы могли бы сделать это посредством создания отдельного удостоверения подписи каталога открытых ключей, посредством подписания каталога с помощью опции "-sb".

К сожалению, вы будете должны еще хранить отдельную достоверную копию вашего собственного открытого ключа для проверки сделанной подписи. Вы не можете полагаться на ваш собственный открытый ключ, хранящийся в каталоге, так как он является частью того, что вы пытаетесь проверить.

Как PGP следит за корректностью ключей

Прежде, чем вы начнете читать этот раздел , убедитесь что вы прочитали предыдущий раздел "Как защитить открытые ключи от подделки".

PGP следит, какие ключи в вашем каталоге открытых ключей правильно удостоверяются подписями поручителей, которым вы доверяете. Все, что вы должны делать — сообщать PGP, каким людям вы доверяете, как поручителям, и удостоверять их ключи у себя с помощью вашей собственной абсолютно достоверной подписи. PGP может брать ее отсюда, автоматически при утверждении любых других ключей, которые были подписаны вашими

указанными поручителями. И, конечно, вы можете прямо подписывать большинство ключей самостоятельно.

Имеются два совершенно разных критерия, которые PGP использует, чтобы судить о полезности открытого ключа:

1) Действительно ли ключ принадлежит упомянутому в идентификаторе человеку? Иными словами, был ли ключ удостоверен с помощью достоверной подписи?

2) Принадлежит ли он кому-либо из тех, кому мы можем доверять для удостоверения других ключей?

PGP может вычислять ответ на первый вопрос. Для ответа на второй вопрос вам, пользователю, необходимо явно пообщаться с PGP. Когда вы вводите ответ на вопрос 2, PGP может вычислить ответ на вопрос 1 для других ключей, подписанных тем поручителем, которого вы указываете как достоверного.

Ключи, удостоверенные поручителем, которому вы доверяете, PGP считает истинными. Ключи, принадлежащие доверенным поручителям, должны быть удостоверены либо вами, либо другими доверенными поручителями.

PGP также позволяет вам иметь несколько степеней доверия для людей, чтобы действовать как поручитель. Ваше доверие к владельцу ключа не просто отражает вашу оценку их персональной честности — это также должно отражать то, насколько компетентны по вашему мнению в понимании управления ключами и в принятии верного решения при подписании ключа. Вы можете обозначить человека как неизвестного для PGP, или как которому нет доверия, или как обладающего частичным доверием, или, наконец, обладающего полным вашим доверием при

удостоверении других открытых ключей. Эта информация сохраняется в вашем каталоге вместе с их ключами, но когда вы даете команду PGP извлечь ключ из вашего каталога, PGP не копирует информацию о степени доверия наряду с ключом, так как ваши частные соображения относительно степени доверия расцениваются как конфиденциальные.

Когда PGP рассчитывает достоверность открытого ключа, она исследует степень доверия для всех присоединенных удостоверяющих подписей. Она вычисляет взвешенную величину достоверности -- два удостоверения с частичной степенью доверия значат то же, что и одно с полной степенью. Скептицизм PGP's можно регулировать — например, вы можете настроить PGP, чтобы она требовала два удостоверения с полной степенью доверия или три с частичной степенью для оценки ключа как допустимого к использованию.

Ваш собственный ключ "абсолютно" достоверен для PGP, он не нуждается ни в какой подписи никакого поручителя для установления истинности. PGP определяет, какие из ключей являются вашими при поиске соответствующих секретных ключей в каталоге секретных ключей. PGP также считает, что вы абсолютно доверяете себе при удостоверении других ключей.

Поскольку время идет, вы будете накапливать ключи от других людей, которых вы можете захотеть обозначить как доверенные поручители. Кто-то еще будет выбирать своих доверенных поручителей. И каждый будет постепенно накапливать и распространять коллекцию удостоверенных подписей других людей, надеясь, что кто-нибудь из получивших ее будет доверять хотя бы одной

или двум подписям. Это может вызвать появление децентрализованной отказоустойчивой сети доверия для всех открытых ключей.

Этот уникальный массовый подход резко контрастирует с правительственной стандартной схемой управления открытыми ключами, такой, как Internet Privacy Enhanced Mail (PEM), которая базируется на централизованном управлении и обязательном централизованном доверии. Стандартные схемы основываются на иерархии Удостоверения Авторства, когда диктуется, кому вы должны доверять. Децентрализованный вероятностный метод PGP для определения законности общего ключа — центральная часть архитектуры управления ключами. PGP позволяет вам быть единственным, кто выбирает, кому вам доверять, помещая вас в верхней части вашей собственной индивидуальной пирамиды достоверности. PGP — для тех людей, которые предпочитают сами упаковывать их собственный парашют.

Как защитить секретные ключи от раскрытия

Тщательно защищайте ваш собственный секретный ключ и фразу пароля. По-настоящему тщательно. Если случилось так, что ваш секретный ключ скомпрометирован, срочно известите об этом все заинтересованные стороны, до того, как ваш ключ будет использован кем-либо для генерации подписи с вашим именем. Например, ваш ключ может быть использован для подписи поддельных удостоверений открытых ключей, что может создать определенные проблемы для широкого

круга людей, особенно если вашей подписи широко доверяют. И, разумеется, компроментация вашего секретного ключа может дать возможность просмотра всех сообщений, адресованных вам.

Для начала всегда сохраняйте физический контроль над вашим секретным ключом. Если вы храните его на вашем личном домашнем или переносном компьютере, то это вполне нормально. Если вы должны использовать компьютер на работе, над которым вы не всегда имеете физический контроль, то тогда храните ваши ключи на защищенной от записи дискете и не оставляйте ее, уходя с работы. Отнюдь не будет способствовать сохранности секретного ключа его хранение на удаленном компьютере, который работает в режиме разделения времени, например, в Unix. Кто-нибудь может подключиться и прослушивать вашу модемную линию, узнать таким образом фразу пароля и получить затем ваш секретный ключ с удаленной системы. Вы должны использовать ваш секретный ключ только на той машине, над которой у вас есть физический контроль.

Не храните вашу фразу пароля на том же компьютере, на котором хранится ваш секретный ключ. Хранение секретного ключа и фразы пароля для него на одном и том же компьютере столь же опасно, как хранение вашего кода в одном бумажнике с кредитной карточкой. Вы же не хотите, чтобы кто-нибудь получил доступ к этому компьютеру и получил одновременно и ключ и пароль. Наиболее безопасным будет вообще нигде не записывать фразу пароля, а просто запомнить ее и не хранить ее нигде, кроме вашей головы.

Если же вы чувствуете, что ее необходимо записать, сделайте это, но как следует защитив ее, даже лучше, чем файл с секретным ключом.

Храните резервные копии вашего каталога секретных ключей — помните, что если у вас есть единственная копия секретного ключа, то его потеря сделает бесполезными все копии вашего открытого ключа, которые вы уже распространили по всему миру.

Децентрализованный подход, который PGP использует для управления открытыми ключами, имеет свои преимущества, но, к сожалению, в то же время это означает, что мы не можем полагаться на один центральный список скомпрометированных ключей. Этот факт делает несколько более трудным распространение информации о компрометации секретных ключей. Вы должны просто распространять словесную информацию и надеяться, что все ее услышат.

Если все-таки случится самое плохое — ваш секретный ключ и фраза пароля все же будут скомпрометированы (надеемся, однако, что вам удастся этого избежать) — вы должны будете выпустить удостоверение "компрометации ключа". Этот тип удостоверения используется для предупреждения других людей о прекращении использования вашего открытого ключа. Для создания такого удостоверения используется команда PGP "-kd". Затем вы должны любым возможным способом постать это удостоверение отмены ключа каждому на планете, или, по крайней мере, всем вашим друзьям, и их друзьям и так далее. Их программа PGP будет устанавливать это удостоверение отмены в их каталогах открытых ключей и будет автоматически

предотвращать использование скомпрометированных ключей в дальнейшем. после этого вы можете сгенерировать новую пару секретный/открытый ключ и опубликовать новый открытый ключ. Вы можете послать один пакет, содержащий и удостоверение отмены и новый открытый ключ.

Отмена открытого ключа

Предположим, что ваш секретный ключ и фраза пароля каким-то образом были скомпрометированы. Вы должны сообщить об этом миру, чтобы никто более не использовал ваш открытый ключ. Для этого вы должны выпустить удостоверение отмены ключа для отмены вашего открытого ключа.

Чтобы сгенерировать удостоверение отмены ключа, используйте команду -kd:

pgp -kd your_userid

Вы должны широко распространить это удостоверение отмены ключа, и как можно скорее. Все, кто получат его, могут добавить его к своему каталогу открытых ключей, и программа PGP будет их автоматически предохранять от использования вашего старого отмененного открытого ключа. После этого вы можете сгенерировать новую пару секретный/открытый ключи и опубликовать новый открытый ключ.

Вы можете использовать отмену открытого ключа не только при его компрометации, но и по каким-то иным соображениям. Механизм отмены остается прежним.

Что, если вы потеряете ваш секретный ключ?

Обычно, если вы хотите отменить ваш собственный секретный ключ, вы можете использовать команду "-kd" для выпуска удостоверения отмены, подписанного вашим собственный секретным ключом.

Но что вам делать, если вы потеряли ваш секретный ключ, либо он был разрушен? Вы не можете сами отменить его, так как нужно использовать сам секретный ключ для отмены, а у вас его больше нет. Будущая версия PGP предложит более надежный способ отмены ключа в такой ситуации, позволяя доверенным поручителям сертифицировать отмену открытого ключа. А пока в этом случае следует всем пользователям передавать информацию о недействительности вашего открытого ключа, дабы они скорректировали свои каталоги.

Другие пользователи могут отменить ваш открытый ключ с помощью команды "-kd". Если заданный идентификатор пользователя не соответствует секретному ключу в каталоге секретных ключей, то эта команда будет искать такой идентификатор в каталоге открытых ключей и помечать соответствующий открытый ключ как недействительный. Недействительный ключ не может быть использован для шифрации сообщений и не может быть извлечен из каталога с помощью команды "-kx". Он по-прежнему может использоваться для проверки подписей, но с выдачей предупреждения. Если пользователь попытается добавить такой же ключ в каталог ключей, этого не произойдет, потому что недействительный ключ уже присутствует в каталоге.

Эти скомбинированные возможности помогут предотвратить дальнейшее распространение недействительного ключа.

Если указанный открытый ключ уже помечен как недействительный, команда -kd выдаст запрос о необходимости восстановления ключа.

Посылка зашифрованного текста через каналы электронной почты: формат Radix-64

Многие системы электронной почты поддерживают только сообщения в виде ASCII-текста, а не в виде двоичных 8-битных данных, из которых состоят зашифрованные тексты. Чтобы обойти эту проблему, PGP позволяет получить формат ASCII Radix-64, подобный формату Internet Privacy-Enhanced Mail (PEM). Этот специальный формат представляет двоичные данные, используя только печатаемые символы ASCII, это полезно для передачи двоичных зашифрованных данных через 7-битовые каналы или для посылки двоичных зашифрованных данных как обычный текст электронной почты. Этот формат действует как "транспортная оболочка", защищая данные от повреждения при передаче их через межсистемные межсетевые шлюзы в Internet. Он также включает в себя CRC-код для определения ошибок при передаче.

Формат Radix-64 преобразует обычный текст при помощи расширения групп из 3-х двоичных 8-битовых байтов в 4 печатаемых символа ASCII, так что файл увеличивается приблизительно на 33%. Но это увеличение

не так уж плохо, если вы учтете, что файл, вероятно, был сжат на большую величину перед кодированием его с помощью PGP.

Для получения зашифрованного файла в формате Radix-64 просто добавьте опцию "a" при шифровании или подписании сообщения, например, так:

pgp -esa message.txt her_userid

В результате будет получен шифрованный файл с именем "message.asc", который содержит данные в PEM-подобном формате Radix-64. Этот файл может быть легко загружен в текстовый редактор через 7-битовые каналы для передачи как нормальная электронная почта в Internet или другой сети.

Дешифровка сообщения в такой транспортной оболочке ничем не отличается от обычного. Например:

pgp message

PGP будет автоматически искать ASCII-файл "message.asc" прежде, чем двоичный файл "message.pgp". Она распознает, что файл находится в формате Radix-64 и преобразует его обратно в двоичный перед обычной обработкой, создавая, как побочный продукт, шифрованный файл ".pgp" в двоичной форме. Окончательный выходной файл будет иметь вид обычного текста, как это было в первоначальном файле "message.txt".

Большинство средств электронной почты Internet запрещают пересылку сообщений, размер которых больше 50000 байт. Более длинные сообщения должны быть разделены на более маленькие, которые пересылаются отдельно. Если ваше зашифрованное сообщение весьма велико, и вы дали команду преобразования в Radix-

формат, то PGP автоматически разделит его на отдельные части, каждая из которых является достаточно малой для посылки через систему электронной почты. Эти части будут помещаться в файлы с расширениями имен ".asc", ".as2", ".as3", и т.д. Получатель должен соединить эти отдельные файлы снова вместе в один большой файл перед тем, как приступать к дешифровке. При дешифровке PGP будет игнорировать любой посторонний текст в заголовках сообщений, который не включен в блоки Radix-64.

Если вы хотите послать открытый ключ кому-либо в формате Radix-64, вам только надо добавить опцию "-a" при извлечении ключа из вашего каталога ключей.

Если вы забыли использовать опцию "-a" при создании зашифрованного файла или извлечении ключа, вы просто можете преобразовать двоичный файл в формат radix-64, используя одну опцию "-a", без задания любого шифрования. PGP преобразует его в файл ".asc".

Если вы хотите послать по каналам электронной почты обычный текстовый файл, который подписан, но не зашифрован, PGP просто конвертирует его в формат radix-64, делая его нечитабельным для обычного наблюдателя. Если исходный файл является просто текстом (не двоичным файлом), то существует способ, с помощью которого можно оставить сам текст в его исходном виде, преобразовав в ASCII-оболочку только электронную подпись. Это делает возможным для получателя прочитать текст сообщения просто глазами, без применения PGP. Естественно, PGP остается необходимой для проверки подписи.

Системная переменная для задания имени пути

PGP использует несколько специальных файлов для своих целей, таких, как ваши стандартные каталоги ключей "pubring.pgp" и "secring.pgp", файл начального числа для генерации случайных чисел "randseed.bin", файл конфигурации PGP "config.txt" и файл перевода сообщений на другие языки "language.txt". Эти специальные файлы могут храниться в любом каталоге, только требуется занести в системную переменную "PGP-PATH" требуемый каталог. Например, для MSDOS, это будет выглядеть так:

SET PGPPATH=C:\PGP

В результате PGP будет знать, что полное имя файла вашего каталога открытых ключей будет иметь вид "C:\PGP\pubring.pgp". Естественно, если этот каталог существует. Используйте ваш любимый текстовый редактор чтобы изменить ваш файл AUTOEXEC.BAT в MSDOS для автоматической установки этой переменной при начальной загрузке системы. Если PGPPATH остается неопределенным, считается, что эти специальные файлы находятся в текущем каталоге.

Установка параметров конфигурации: файл CONFIG.TXT

PGP имеет ряд параметров, устанавливаемых пользователем, которые могут быть определены в специальном текстовом файле конфигурации с именем "config.txt", в каталоге, на который указывает системная

переменная PGPPATH. Наличие файла конфигурации дает возможность пользователю определять в нем различные флажки и параметры для PGP, исключая необходимость каждый раз определять эти параметры в командной строке.

С помощью этих параметров конфигурации, например, вы можете определить, где PGP сохраняет временные рабочие файлы, или вы можете выбирать иностранный язык, который PGP будет использовать для отображения диагностических сообщений и подсказок пользователю, либо можете регулировать уровень скептицизма PGP при определении истинности ключей, который основывается на числе удостоверяющих подписей.

Уязвимость

Никакая система защиты данных не является несокрушимой. PGP может быть обойдена целым рядом способов. Это может быть компроментация вашего секретного ключа и фразы пароля, подделка общего ключа, файлы, которые вы удалили, но они остались физически на диске, вирусы и троянские кони, бреши в вашей физической защите, электромагнитная эмиссия, дефект в многопользовательских системах, анализ траффика, и, возможно, даже прямой криптоанализ.

Доверие к змеиному маслу

При исследовании пакета криптографического программного обеспечения всегда остается вопрос: почему вы должны доверять этой программе? Даже если вы

самостоятельно исследовали исходный текст, но ведь не каждый имеет опыт в криптографии, чтобы судить о степени защиты. Даже если вы опытный шифровальщик, вы можете пропустить небольшие слабости в алгоритмах.

Когда я учился в колледже в начале семидесятых, я верил, что изобрел прекрасную схему шифрования. Простой псевдослучайный поток чисел добавлялся к потоку текста, чтобы создть зашифрованный текст. В результате, по-видимому, частотный анализ зашифрованного текста станет невозможным, и такой текст будет недоступен для расшифровки даже наиболее мощным правительственным службам. Я был настолько уверен в этом достижении. Излишне самоуверен.

Спустя несколько лет я обнаружил эту самую схему в нескольких статьях и учебниках по криптографии. Другой специалист размышлял о той же самой схеме. К сожалению, она была представлена как одна из простейших в качестве примера об использовании элементарной криптографической технологии для ее вскрытия. Столь много для моей прекрасной схемы.

Из этого скромного опыта я понял, как легко ошибиться в оценке качества защиты при изобретении алгоритма шифрования. Большинство людей не понимают, как дьявольски трудно изобрести алгоритм шифрования, который мог бы долго противостоять и отражать атаки сильного противника. Множество инженеров-программистов разработали одинаково наивные схемы шифрования (часто одинаковые), и некоторые из них были включены в коммерческие криптографические программные пакеты и продавались за хорошие деньги тысячам доверчивых пользователей.

Это похоже на продажу самофиксирующихся ремней безопасности, которые смотрятся хорошо и удобны, но фиксируются в открытом состоянии даже при самой медленной проверке. Зависимость от таких ремней может быть хуже, чем даже не использовать ремни совсем. Никто не предполагает, что они они действительно плохи вплоть до настоящей аварии. Зависимость от слабого криптографического программного обеспечения может привести к необходимости размещения секретной информации с долей риска. Ведь вы не смогли бы иначе выполнить это, если бы у вас не было никакого криптографического программного обеспечения вообще. Возможно, вы даже никогда и не обнаружите, что ваши данные были скомпрометированы.

Иногда коммерческие пакеты используют Федеральный Стандарт Шифрования Данных (DES), хороший стандартный алгоритм рекомендуемый правительством для коммерческого применения (но не для секретной информации, что в значительной степени странно — хм...). Существует несколько "режимов операции", которые DES может использовать, некоторые из них лучше, другие хуже. Правительство специально не рекомендует использовать самый слабый и простейший режим для сообщений, режим Electronic Codebook (ECB). Оно рекомендуют более сильные и более сложные режимы Cipher Feedback (CFB) или Cipher Block Chaining (CBC).

К сожалению, большинство рассмотренных мною коммерческих пакетов для шифрования применяют режим ECB. Когда я разговаривал с авторами ряда подобных систем, они говорили мне, что никогда не слышали о режимах CBC и CFB, и ничего не знали о слабостях режима ECB. Тот факт, что они даже не изучили

криптографию в объеме, достаточном для знания элементарных принципов, не убеждает. Те же самые программные пакеты часто включают в себя второй более быстрый алгоритм шифрования который может использоваться вместо медленного DES. Автор такого пакета нередко полагает, что его собственный более быстрый алгоритм является столь же надежным, как и DES, но после подробных расспросов я обычно выясняю, что это просто-напросто разновидность моей знаменитой схемы времен колледжа. А иногда он может даже не рассказать, как работает его схема, но будет уверять меня, что это прекрасная схема и я должен ему верить. Несомненно, он полагает, что его алгоритм великолепен, но как я могу верить этому, не видя самого алгоритма?

Со всей честностью я должен подчеркнуть, что в большинстве случаев такие программы выпускаются компаниями, которые не специализируются в криптографической технологии.

Существует компания, которая называется AccessData (87 East 600 South, Orem, Utah 84058, телефон 1-800-658-5199), которая продает за 185 долларов пакет, который взламывает встроенные схемы шифрования, используемые WordPerfect, Lotus 1-2-3, MS Excel, Symphony, Quattro Pro, Paradox и MS Word 2.0. Она не просто разгадывает пароли, она осуществляет настоящий криптоанализ. Некоторые люди покупают ее, если они забыли пароли своих файлов. Административные власти также покупают ее, таким образом они могут читать захваченные файлы. Я разговаривал с автором, Eric Thompson, он сказал что его программа вскрывает защиту за долю секунды, но он добавил в программу несколько циклов задержки, чтобы замедлить этот процесс, дабы это

не казалось слишком простым для заказчика. Он также сообщил нам, что функция шифрования с паролем архиватора PKZIP может быть легко взломана, и его официальные заказчики уже обладают таким средством, которое им поставил другой продавец.

Иными словами, криптография подобна фармакологии. Ее достоверность имеет решающее значение. Плохой пеннициллин выглядит также, как и хороший пеннициллин. Вы можете сообщить о том, что ваш электронный бланк ненадежен, но как вы сообщите о том, что ваш криптографический пакет слаб? Текст, который был зашифрован с помощью слабого алгоритма шифрования выглядит столь же хорошо, как и текст, зашифрованный мощным алгоритмом. Такой текст похож на "змеиное масло" или лечение шарлатана. Однако, в отличие от продавцов патентованных лекарств, изготовители этого программного обеспечения обычно даже не подозревают, что их продукция — настоящее змеиное масло. Они могут быть хорошими специалистами по разработке программ, но они обычно даже не читали никакой академической литературы по криптографии. И они считают, что могут написать хорошее криптографическое программное обеспечение. А почему бы и нет? Несмотря ни на что, интуиция подсказывает, что это сделать достаточно легко. И создается впечатление, что их программное обеспечение работает хорошо.

Если кто-то считает, что он изобрел невзламываемую систему шифрования, тот либо невероятно редкий гений, либо наивен и неопытен.

Я помню диалог с Brian Snow, высокопоставленным специалистом по криптографии из NSA. Он сказал, что никогда не будет доверять алгоритму шифрования, разработанному человеком, который это не "заслужит", потратив предварительно достаточно времени на раскрытие кодов. В этом весьма много смысла. Я не встречал в мире коммерческой криптографии практически ни одного человека, которого можно было бы охарактеризовать таким образом. "Да", — сказал он с уверенной улыбкой, — "и это делает нашу работу в NSA значительно проще". Неприятная мысль. Я тоже не занимался этим.

У правительства, по слухам, тоже есть "змеиное масло". После Второй Мировой войны Соединенные Штаты продали шифровальные машины Enigma правительствам третьих стран. Но при этом они не сообщили им, что союзники за время войны расшифровали код машины Enigma, факт, который держался в секрете много лет. Даже сегодня многие системы Unix во всем мире используют шифр Enigma для шифрования файлов, частично оттого, что правительство создало все препятствия для использования лучших алгоритмов. Оно даже пыталось предотвратить первоначальную публикацию алгоритма RSA в 1977 году. И теперь направляет все коммерческие усилия на разработку эффективных секретных телефонов.

Основная работа правительственного NSA — это собирать информацию, преимущественно путем скрытого подключения к частным коммуникационным линиям (см. книгу James Bamford "The Puzzle Palace"). NSA обладает значительным мастерством и ресурсами для раскрытия кодов. Если люди не обладают мощной

криптографической системой для защиты своей информации, это делает задачу NSA значительно проще. Кроме того, у NSA есть обязанность одобрять и рекомендовать алгоритмы шифрования. Необходимо отметить это интересное противоречие, подобное назначению лисы для охраны курятника. NSA протолкнуло стандартный алгоритм шифрования, разработанный им же, и оно не собирается никому сообщать, как он работает, так как он является секретным. Однако, оно требует, чтобы все этому алгоритму доверяли и использовали его. Но любой специалист в области криптографии может вам подтвердить, что удачно разработанный алгоритм шифрования не должен быть засекречен, дабы оставаться надежным. В защите должны нуждаться только ключи. Как мы можем знать, насколько надежен алгоритм шифрования NSA? Для NSA совсем не сложно разработать алгоритм, который сможет взломать только оно, если этот алгоритм больше никто не увидит. Не является ли это преднамеренной продажей "змеиного масла"?

Я не могу быть полностью уверен в уроне надежности PGP, поскольку однажды я уже разрабатывал прекрасный алгоритм — в колледже. Если я снова буду так же уверен, это будет плохим сигналом. Но я в достаточной степени уверен, что PGP не содержит грубых ошибок. Криптографические алгоритмы разрабатывались людьми с высоким уровнем криптографического образования. Исходные тексты доступны для облегчения изучения PGP и с целью помочь рассеять опасения некоторых пользователей. Разработке предшествовали достаточно глубокие исследования и годы работы. К тому же я не

работаю на NSA. Я надеюсь, что не требуется слишком большого "скачка в доверие" для того, чтобы доверять надежности PGP.

Краткий справочник команд PGP

Здесь приведена краткая сводка команд PGP.

Зашифровать текстовый файл с открытым ключом получателя: pgp -e textfile her_userid

Для подписания текстового файла вашим секретным ключом: pgp -s textfile [-u your_userid]

Для подписи текстового файла вашим секретным ключом и, затем, зашифровки его с открытым ключом получателя:

pgp -es textfile her_userid [-u your_userid]

Для шифрования текстового файла стандартным криптографическим методом:

pgp -c textfile

Для расшифровки зашифрованного файла или для проверки целостности подписи подписанного файла:

pgp ciphertextfile [-o plaintextfile]

Для шифрования сообщения, предназначенного для нескольких адресатов:

pgp -e textfile userid1 userid2 userid3

Команды для работы с ключами

Сгенерировать вашу собственную уникальную пару секретный/открытый ключи:

pgp -kg

Для того, чтобы добавить содержимое файла открытого или секретного ключа в ваш каталог открытых или секретных ключей:

pgp -ka keyfile [keyring]

Для извлечения (копирования) ключа из каталога ключей: pgp -kx userid keyfile [keyring]

или: pgp -kxa userid keyfile [keyring]

Для просмотра оглавления каталога открытых ключей: pgp -kv[v] [userid] [keyring]

Для просмотра "отпечатка пальца" открытого ключа, чтобы помочь вам удостовериться в его истинности по телефону в разговоре с владельцем ключа:

pgp -kvc [userid] [keyring]

Для просмотра оглавления и проверки удостоверяющих подписей в вашем каталоге открытых ключей:

pgp -kc [userid] [keyring]

Для редактирования идентификатора пользователя или фразы пароля для вашего секретного ключа:

pgp -ke userid [keyring]

Для редактирования параметров доверия для открытого ключа: pgp -ke userid [keyring]

Удалить ключ или только идентификатор пользователя из вашего каталога открытых ключей:

pgp -kr userid [keyring]

Для подписи и удостоверения чьего-либо открытого ключа в вашем каталоге открытых ключей:

pgp -ks her_userid [-u your_userid] [keyring]

Для удаления выбранных подписей из идентификатора пользователя каталога ключей:

pgp -krs userid [keyring]

Для постоянной отмены вашего собственного ключа с помощью выпуска удостоверения отмены:

pgp -kd your_userid

Для отмены или восстановления открытого ключа в вашем каталоге открытых ключей:

pgp -kd userid

Сложные команды

Для дешифровки сообщения, оставляя подпись на нем нетронутой: pgp -d ciphertextfile

Для создания удостоверяющей подписи отдельно от документа: pgp -sb textfile [-u your_userid]

Для отделения удостоверяющей подписи от подписанного сообщения: pgp -b ciphertextfile

Опции команд, которые могут использоваться в комбинации с другими опциями

Для получения зашифрованного файла в формате ASCII radix-64 добавьте опцию -a при шифровании или подписании сообщения или извлечения ключа:

pgp -sea textfile her_userid

или: pgp -kxa userid keyfile [keyring]

Для полного удаления оригинального текстового файла после создания зашифрованного файла просто добавьте опцию -w (wipe) при шифровании или подписании сообщения:

pgp -sew her_userid message.txt

Для указания того, что что текстовый файл содержит текст ASCII, а не двоичный, и должен быть преобразован в локальный текстовый файл получателя, добавьте опцию -t (text) к другим:

pgp -seat message.txt her_userid

Для просмотра выводимого расшифрованного текста на вашем экране (подобно команде "more" в Unix), без записи его в файл, используют опцию -m (more) при расшифровке:

pgp -m ciphertextfile

Для задания возможности просмотра распакованного текста ТОЛЬКО на экране без возможности записи на диск, добавьте опцию -m:

pgp -steam her_userid message.txt

Для восстановления оригинального имени файла в процессе дешифровки, добавьте опцию -p:

pgp -p ciphertextfile

Для использования режима фильтра (как в Unix), читая из стандартного потока ввода и записывая в стандартный поток вывода, добавим опцию -f:

pgp -feast her_userid <inputfile >outputfile

Philip Zimmermann

Philip Zimmermann — инженер-консультант по разработке программного обеспечения с 18-летним стажем, специализируется в сложных системах реального времени, криптографии, идентификации и передаче данных. Его опыт включает разработку и внедрение

идентификационных систем для финансовых информационных сетей, защиты данных в сетях, протоколов управления ключами, сложных многозадачных систем реального времени, операционных систем и локальных компьютерных сетей.

Zimmerman также предлагает разработку на заказ версий криптографических и идентификационных продуктов и реализаций публичного ключа, таких, как NIST DSS, а также поддержку разработанных продуктов. Адрес его консультационной фирмы:

Boulder Software Engineering

3021 Eleventh Street

Boulder, Colorado 80304 USA

Телефон 303-541-0140 (голос или факс)

Internet: prz@sage.cgd.ucar.edu

Благодарности

Я хотел бы поблагодарить следующих людей за их содействие созданию Pretty Good Privacy. Хотя я и был автором PGP версии 1.0, основные части последующих версий были выполнены усилиями многих людей из разных стран под моим общим руководством.

Branko Lankester, Hal Finney и Peter Gutmann пожертвовали огромным количеством времени для расширения возможностей PGP 2.0 и переноса ее в Unix. Hal и Branko приложили просто героические усилия для реализации моих новых протоколов управления ключами. Branko пожертвовал времени больше, нежели кто-либо другой.

Hugh Kennedy перенес PGP в систему VAX/VMS, Lutz Frank — на Atari ST, а Cor Bosman и Colin Plumb — на Commodore Amiga.

Перевод PGP на другие языки был выполнен: Jean-loup Gailly во Франции, Armando Ramos в Испании, Felipe Rodriquez Svensson и Branko Lankester в Нидерландах, Miguel Angel Gallardo в Испании, Hugh Kennedy и Lutz Франк в Германии, David Vincenzetti в Италии, Harry Bush и Maris Gabalins в Латвии, Zygimantas Cepaitis в Литве, Peter Suchkow и Andrew Chernov в России и Александр Смишалев перевел ее на эсперанто. Peter Gutmann предложил перевести PGP на английский язык Новой Зеландии, но, в конце концов, мы решили, что можно будет получить PGP с английским языком США.

Jean-loup Gailly, Mark Adler и Richard B. Wales опубликовали код сжатия ZIP и предоставили разрешение на включение его в PGP. Ron Rivest разработал и опубликовал для свободного использования (public domain) подпрограммы MD5. Шифр IDEA(tm) был разработан Xuejia Lai и James L. Massey из ETH в Zurich, и используется в PGP с разрешения Ascom-Tech AG.

Charlie Merritt научил нас как использовать хорошую арифметику высокой точности для криптографии с открытым ключом, а Jimmy Upton пожертвовал нам алгоритм более быстрого умножения по модулю. Thad Smith реализовал еще более быстый алгоритм. Zhahai Stewart принес много полезных идей относительно форматов файлов PGP и других вещей, включая идею наличия более, чем одного идентификатора пользователя для ключа.

Я услышал идею относительно поручителей от Whit Diffie. Kelly Goen провел всю основную работу для первоначальной электронной публикации PGP 1.0.

Различные усилия по созданию кода приложили также Colin Plumb, Derek Atkins и Castor Fu. Помогли нам также Hugh Miller, Eric Hughes, Tim May, Stephan Neuhaus, и многие другие. В процессе реализации находятся два проекта переноса PGP на Macintosh.

С момента выхода версии 2.0 многие программисты присылали поправки и исправления ошибок, вносили исправления на других системах компьютеров. Их слишком много, чтобы выразить благодарность каждому индивидуально.

Процесс развития PGP преобразовался в примечательный социальный феномен, чья уникальная политическая привлекательность вдохновляет на совместные усилия все большее число программистов-добровольцев.

Проект русификации PGP 5.0

Ряд правительств серьезно наказывает своих граждан за использование шифрованных коммуникаций. В некоторых странах вас даже могут за это расстрелять. Но если вы живете в такой стране, возможно, PGP вам тем более пригодится.

PGP 5.0: Быстрый старт

Если вы используете PGP в первый раз, сначала вам нужно сгенерировать пару ключей, выбрав в меню Keys программы PGPkeys пункт New Key. Как правило, вам удастся сделать это автоматически через Помощник генерации ключа. Затем вам нужно будет послать открытый ключ другому пользователю. Для этого перетащите мышью ключ из главного окна PGPkeys в окно почтового сообщения. После этого пользователь, который получил ваш ключ, сможет шифровать направляемую вам почту. Чтобы посылать зашифрованные письма ему, вам потребуется получить его открытый ключ. Подписывать письма вы можете и без отправки своего открытого ключа другим пользователям, но тогда никто не сможет проверить вашу подпись. Вы также можете отправить свой открытый ключ на публично доступный сервер ключей, с которого этот ключ смогут получить другие пользователи.

Ключи Diffie-Hellman/DSS могут осложнить коммуникацию с пользователями ранних версий PGP. Diffie-Hellman/DSS — это новый тип ключей, являющийся по крайней мере столь же надежным, что и ключи RSA той же длины. Однако, ключи DH/DSS не поддерживаются более ранними версиями PGP, что означает, невозможность обмена зашифрованной почтой с пользователями, которые еще не перешли к использованию версии 5.0 или выше. Использование ключей DH/DSS значительно сокращает время, необходимое для шифрования и расшифровки.

Импорт файлов с ключами из ранних версий PGP. Ваши файлы с ключами должны быть скопированы в папку установки PGP 5.0 при выполнении установки. Чтобы импортировать другие файлы с ключами, лучше всего физически заместить файлы с ключами по умолчанию "pubring.pkr" и "secring.skr" вашими старыми файлами "pubring.pgp" и "secring.pgp" в то время, когда PGP не запущена. При этом, информация о приписанных ключам степенях доверия сохраняется. Другой способ — это просто перетащить старые файлы с ключами из окна Проводника (Explorer) в главное окно PGP или выбрать пункт Import... из меню Keys программы PGPkeys. При использовании этого способа информация о доверии не будет перенесена, так как если вы получили файл с открытыми ключами от кого-то другого, информация о его степени доверия к ним вам ни к чему. Если у вас на связке несколько закрытых ключей, вам нужно использовать команду Set Default из меню Keys программы PGPkeys, чтобы указать ключ, который при подписи других ключей, а также сообщений, будет использоваться по умолчанию.

Чтобы послать свой открытый ключ другому пользователю, просто переместите его мышью в любое текстовое окно, или отправьте его на сервер ключей, а затем попросите своих друзей подгрузить его, используя PGP 5.0 или браузер. Обычным является включение URL, указывающего на ключ, в стандартную подпись ваших сообщений. Такой URL выглядит следующим образом:

<http://swissnet.ai.mit.edu:11371/pks/lookup?op=get&search=0x272727>

Конечно, вам нужно заменить идентификатор ключа, которым заканчивается URL (0x27272727) на идентификатор вашего собственного ключа. Узнать идентификатор своего ключа вы можете, выбрав этот ключ в главном окне PGPkeys и использовав пункт Properties меню Keys этой программы.

Прием "Переместить и оставить" работает почти везде. Вы можете переместить ключи, идентификатор пользователя, подписи непосредственно на поверхность рабочего стола, перемещать идентификаторы пользователей из списка идентификаторов в список получателей и т.п.

Чтобы подписать ключ, выделите его и выберите пункт Sign из меню Keys в PGPkeys. Вы можете затем указать степень доверия, с которой вы относитесь к данному ключу, щелкнув на нем правой кнопкой мыши и выбрав из контекстного меню пункт Key Properties. Если вы укажете, что степень доверия к этому ключу является "полной" ("Complete"), другие ключи, подписанные его владельцем, будут считаться действительными.

Чтобы отозвать ключ, выделите его и выберите пункт Revoke из меню Keys в PGPkeys.

Имейте в виду, что новый интерфейс делает возможными многие вещи, которые раньше не были возможны (или занимали слишком много времени). Это включает подписывание одновременно нескольких ключей. Для этого, выделите все ключи, которые хотите подписать, и выберите Sign из меню Keys. Вы также можете удалять идентификаторы пользователя с ключей, удалять подписи, использовать перетаскивание мышкой для импорта связок ключей, на ходу управлять доверием и действительностью ключей.

Ответы на часто задаваемые вопросы

Сколько дискового пространства необходимо для успешной установки PGP 5.0 на компьютере?

Для успешной установки вам понадобится 15 MB.

Что обозначают различные значки в PGP Keys?

Один золотой ключ обозначает открытый ключ [из пары], сгенерированной по технологии DSS/Diffie-Hellman. Пара синих ключей обозначает вашу пару, состоящую из секретного и открытого ключей, сгенерированную по технологии RSA. Один синий ключ обозначает открытый ключ [из пары], сгенерированной по технологии RSA. Когда ключ или пара ключей изображены бледным цветом, это значит, что они временно недоступны для использования при шифровании и формировании подписей. Ключ, перечеркнутый красной линией, обозначает отозванный ключ.

Как мне импортировать и экспортировать ключи с сервера ключей?

Для того, чтобы экспортировать открытый ключ со своей связки на сервер ключей:

Откройте PGP Keys

Щелкните на [нужном] ключе правой кнопкой мыши

Выберите опцию Keyserver

Щелкните на пункте меню Send Selected Key

Для того, чтобы импортировать ключ с сервера:

Откройте PGP Keys

Откройте меню Keys

Выберите опцию Keyserver

Введите почтовый адрес или идентификатор ключа, который вы хотите найти

Где располагаются plug-ins в Eudora/Exchange?

Соответствующие кнопки, появляются, когда вы читаете сообщение или составляете новое сообщение.

Как мне распространить мой открытый ключ?

Предпочтительным способом является помещение вашего открытого ключа на сервер ключей. PGP 5.0 может делать это автоматически во время создания ключа. Вы также можете щелкнуть на ключе правой кнопкой мыши, выбрать Keyserver и щелкнуть на Send Selected Key. Чтобы отправить [открытый] ключ кому-нибудь по почте, переместите ключ с помощью мыши из PGPkeys в окно почтового сообщения.

Я получил чей-либо [открытый] ключ по почте. Как мне добавить его на свою связку ключей?

Если вы используете [в качестве почтовой программы MS] Exchange или Eudora, вы можете щелкнуть мышью на кнопке Extract PGP Key(s) from Email Message. Если вы используете другую почтовую программу, скопируйте фрагмент текста, содержащий ключ, в буфер обмена, затем перейдите в окно PGP keys и выберите из меню Edit пункт Paste. [Добавленный] ключ будет показан в виде значка в окне PGP keys.

Как мне зашифровать, расшифровать, подписать или проверить подпись файла, используя Проводник?

Щелкните правой кнопкой мыши на файле, выберите PGP, затем щелкните на операции, которую хотите выполнить.

Я не использую Exchange, Outlook, или Eudora, как мне зашифровать или подписать почтовое сообщение?

После того, как вы набрали текст сообщения, скопируйте его в буфер обмена, затем выберите PGPtray в системном меню, далее выберите Encrypt Clipboard, Sign Clipboard или Encrypt and Sign Clipboard. [Далее, вернитесь в окно почтовой программы и вставьте содержимое буфера обмена в текст сообщения].

Я не использую Exchange, Outlook, или Eudora, как мне расшифровать зашифрованное сообщение или проверить подписанное сообщение?

Скопируйте содержимое сообщения в буфер обмена, выберите PGPtray в системном меню, далее выберите Decrypt/Verify Clipboard.

Могу ли я использовать в PGP 5.0 ключи, созданные в более ранних версиях PGP?

Да. Вы можете перетащить мышью старые связки ключей в [окно] PGPkeys, или [в Проводнике] два раза щелкнуть мышью на файле со старой связкой ключей.

Совместима ли PGP for Personal Privacy 5.0 с предыдущими версиями PGP?

PGP 5.0 полностью совместима с предыдущими версиями PGP. Некоторые из предыдущих версий должны быть немного модернизированы (файлы модернизации доступны с нашего сервера () для улучшения совместимости с новыми типами ключей. Использование в версии 5.0 ключей, сгенерированных по технологии DSS/Diffie-Hellman ограничивают обратную совместимость, так как пользователь более ранней версии не сможет проверить вашу подпись, и будет не в состоянии [использовать ваш сгенерированный по этой технологии ключ] для шифровки направляемых вам сообщений. Пользователям, которые продолжают использовать старые версии PGP, мы рекомендуем провести бесплатную модернизацию до версий PGPmail 4.5.1 и PGPmail 4.0.1 для улучшения совместимости. Модернизация же до версии 5.0 обеспечит полную совместимость со всеми релизами PGP и предоставит все

преимущества новых ключей, генерируемых по технологии DSS/Diffie-Hellman keys.

Как признаки валидности и доверия перенести с моих [существующих] ключей, сгенерированных по технологии RSA на ключи, сгенерированные по технологии Diffie-Hellman?

Признаки валидности и доверия действующего ключа RSA будут автоматически перенесены на ключ Diffie-Hellman при подписи ключа DH ключом RSA если оба ключа обладают одним идентификатором пользователя и находятся на одной связке.

Почему в PGP включен дополнительный механизм DSS/Diffie-Hellman?

Дополнительный механизм DSS/Diffie-Hellman включен для обеспечения гибкости системы в будущем, а также потому, что позволяет значительно улучшить производительность системы.

Что такое PGP/MIME и когда он используется?

PGP/MIME представляет собой стандарт IETF, который позволяет пользователям PGP автоматически шифровать и подписывать приложения при отправке почтовых сообщений, кроме того, PGP/MIME предоставляет пользователям более удобный интерфейс. При получении сообщения в формате PGP/MIME тело сообщения заменяется иконкой, показывающей, было ли сообщение зашифровано и/или подписано. При двойном щелчке мышью, будет расшифровано сообщение или проверена подпись. ВАЖНО: формат PGP/MIME следует использовать только при обмене сообщениями с пользователями PGP версии 5.0 или более поздних.

Пользователи более ранних версий могут столкнуться с проблемами при расшифровке или поверке подписи сообщений в формате PGP/MIME.

Существует ли режим plug-in для Microsoft's Outlook Express?

В настоящее время PGP не работает в режиме plug-in с Outlook Express, поскольку программа этот режим не поддерживает.

Что представляет собой "MessageID" в сообщениях, зашифрованных PGP?

MessageID (идентификатор) использовался много лет назад во времена BBS и FIDOnet. Некоторые почтовые системы этого типа не могли обрабатывать длинные сообщения и PGP снабжалась дополнительной способность разбивать сообщения на части. MessageID позволял PGP снова склеивать разбитые на дробные части сообщения в правильном порядке при получении сообщения. В настоящее время это средство не имеет применения.

Почему ключи PGP 5.0 настолько длиннее ключей PGP 2.6.2?

Создаваемый по умолчанию открытый ключ PGP 5.0 на самом деле включает два открытых ключа: ключ DSS для формирования подписи и ключ Diffie-Hellman для шифрования. Кроме этого, в PGP 5.0 компонент Diffie-Hellman может быть в два раза длиннее ключа максимальной длины в версии 2.6.2.

Как мне проверить целостность двоичных файлов PGP 5.0, которые я получил?

Все распространяемые PGP, Inc. файлы, содержащие криптографические программы, подписаны с помощью корпоративного ключа PGP, Inc., так что каждый пользователь может проверить, не были ли эти файлы модифицированы после того, как их подписали. Эти подписи содержатся в директории "signatures", вложенной в директорию, в которую вы установили PGP 5.0. [Открытый] корпоративный ключ находится на связке, распространяемой вместе с PGP 5.0. Такие подписи называются "отделенными" ("detached signatures"), поскольку они размещаются в отдельных от подписываемых файлов файлах. Для проверки целостности включенных в состав PGP 5.0 двоичных файлов перейдите в директорию "signatures" и щелкните правой кнопкой по подписи. Выберите из контекстного меню PGP -> Verify Signature. Появится диалог, запрашивающий у вас имя файла, который вы хотите проверить (файла, соответствующего данной подписи). Обратитесь к списку файлов, приведенному ниже, для того, чтобы определить, где находится соответствующий файл. Например, щелкните правой кнопкой мыши на файле PGPkeys.exe.sig и выберите PGP -> Verify Signature из появившегося контекстного меню. В открывшемся окне диалога перейдите в директорию PGP50 и выберите файл PGPkeys.exe, который соответствует отделенной подписи PGPkeys.exe.sig. Нажмите кнопку Open.

Повторите эту операцию с каждым файлом для проверки их целостности.

Имейте в виду, что некоторые файлы (например, .dll) находятся в других директориях, так что для определения нахождения файлов обращайтесь к списку, приведенному ниже.

PGPkeys.exe ➪ директория, в которую установлена PGP 5.0

PGPtray.exe ➪ директория, в которую установлена PGP 5.0

PGPks.dll ➪ Windows\System PGPwctx.dll ➪ Windows\System PGPcmdlg.dll ➪ Windows\System PGPRecip.dll ➪ Windows\System PGP.dll ➪ Windows\System Simple.dll ➪ Windows\System Bn.dll ➪ Windows\System Keydb.dll ➪ Windows\System PGPExch.dll ➪ Windows\System

PGPplugin.dll ➪ Eudora\plugins

В случае, когда имеются основания предполагать возможность модификации двоичных файлов, входящих в поставку PGP 5.0, недобросовестным посредником, предлагаемая авторами документа процедура проверки целостности не является удовлетворительной. Если недобросовестный посредник модифицировал файлы, входящие в поставку, он также мог сгенерировать фальшивую пару ключей с идентификатором, совпадающим с идентификатором ключа PGP, Inc. Пользователю, подозревающему, что имеющаяся у него копия поставки PGP 5.0 является модифицированной недобросовестным посредником, в качестве первой меры рекомендуется удалить открытый ключ PGP, Inc., содержащийся на связке, входящей в поставку, и получить заведомо аутентичную копию ключа, например, с сервера www.pgp.com или с сервера ключей.

Это позволяет лишь снизить риск, но не решает проблему определения целостности файлов в общем виде, т.к. модифицированная злоумышленником программа может заведомо некорректно выполнять процедуру проверки подписи.

Советы

При использовании MS Outlook, вы должны запретить опцию Use Microsoft Word as the e-mail editor, чтобы PGP plug-in работал правильно. Это может быть сделано выбором в MS Outlook меню Tools, затем Options. Щелкните на вкладке E-mail и сбросьте флажок "Use Microsoft Word as the e-mail editor."

При использовании MS Exchange, вы должны запретить опцию Always send messages in Microsoft Exchange rich text format. Если эта опция не запрещена, MS Exchange разрушит целостность подписанных PGP сообщений, вставляя разметку RTF в уже подписанное сообщение. Для того, чтобы запретить эту опцию, выберите в MS Exchange меню Tools, далее выберите Address Book. Двойной щелчок на имени пользователя в адресной книге вызовет диалог, на первой вкладке которого содержится опция Always send messages in Microsoft Exchange rich text format. Сброс этой опции нужно выполнить для каждого пользователя.

Сообщение об ошибке "The decompression of %s failed. There may not be enough disk space available in your TEMP directory." Это проблема с программой установки InstallShield. Чтобы обойти ее, очистите временную директорию Windows [обычно, "Windows\TEMP"] и запустите программу установки еще раз.

Во время установки может появиться сообщение "Insert Disk 2". Если это мешает установке, очистите временную директорию Windows [обычно, "Windows\TEMP"] и запустите программу установки еще раз. Это также ошибка программы установки.

Известные ошибки в PGP 5.0

Curly CAPS-o-TILDE (231-123)

Поставляемая MIT версия PGP50freeware (не уверен относительно PGP50trial и коммерческой версии) содержит ошибку в нулевой кодировочной таблице, из-за которой программа распознает символ ASCII # 231 (X заглавная в кодировке cp1251, используемой Windows для представления русских текстов, и у строчная в стандартной кодировке Интернет KOI8) как ascii # 123 ({). Это приводит к искажению текстов при шифровании и подписи, а также к ошибкам при взаимодействии с другими версиями PGP.

Обсуждение

Рядом пользователей были высказаны предположения, что это не ошибка, а резервация, внесенная PGP, Inc. для предотвращения использования 5.0 в неанглоязычных регионах до выхода "международного релиза". Больше похоже все же на банальную опечатку.

Решение

1.1 Проверьте, содержит ли ваш экземпляр указанную ошибку. Для этого (1.1.1) скопируйте в Буфер обмена русский алфавит:

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮ Яабвгдеёжзийклмнопрстуфхцчшщъыьэюя

и (1.1.2) подпишите содержимое Буфера выбором пункта Sign Clipboard меню, выскакивающего при щелчке на значке PGPtray в Области системных индикаторов (Tray). Затем (1.1.3) проверьте содержимое Буфера командой Launch Associated Viewer того же меню. Если результат выглядит так:

-----BEGIN PGP SIGNED MESSAGE-----

АБВГДЕЁЖЗИЙКЛМНОПРСТУФ{ЦЧШЩЪЫЬЭЮ Яабвгдеёжзийклмнопрстуфхцчшщъыьэюя

-----BEGIN PGP SIGNATURE-----

Version: PGP for Personal Privacy 5.0

Charset: noconv

<...>

-----END PGP SIGNATURE-----

значит, ваш экземпляр содержит ошибку.

1.2 Найдите файл pgp.dll в папке Windows/SYSTEM и сделайте его резервную копию pgp.dll.old

1.3 Закройте PGPtray выбором команды Quit PGPtray из меню, выскакивающего при щелчке на значке PGPtray в Области системных индикаторов (Tray).

1.4 Откройте файл pgp.dll в папке Windows/SYSTEM любым редактором, позволяющий осуществлять поиск в

шестнадцатиричном формате и редактирование (подойдет hiew). Найдите цепочку символов

D1 D2 D3 D4 7B D6 D7

и замените в ней 7B на D5, сохраните результат и выйдите из редактора.

1.5 Запустите PGPtray снова командой Start|Programs|Accessories|Pretty Good Privacy|PGPtray (Пуск|Программы|Стандартные|Pretty Good Privacy|PGPtray). Повторите проверку, согласно процедуре 1.1 настоящего описания. Результат должен иметь вид:

-----BEGIN PGP SIGNED MESSAGE-----

АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯабвгдеёжзийклмнопрстуфхцчшщъыьэюя

-----BEGIN PGP SIGNATURE-----

Version: PGP for Personal Privacy 5.0

Charset: noconv

<...>

-----END PGP SIGNATURE-----

Проблемы совместимости с ранними версиями

Пользователи PGP показывают, что в ряде случаев наблюдается несовместимость PGP 5.0 с прежними версиями PGP при шифровании/расшифровке и подписи/проверке файлов, содержащих символы верхней половины кодовой таблицы, соответствующие кириллическим буквам.

Блок, подписанный ключом от версии 2.6.3i почему-то определяется фриварной версией как "bad сигнатура Igor Dorohin".

Верить в наше время нельзя никому. Даже себе.

Мне — можно.

Блок подписанный ключом от версии 2.6.3i почему-то определяется фриварной версией как "bad сигнатура.

MEOE> PGP 5.0 for Windows американского релиза содержит ошибку.

Я бодался в различных кодировках (причем умышленно исключая буквы "y" и "X" в соответствующих кодировках), но ничего не смог сделать. В аттаче образцы моего творчества.

MEOE> при использовании режима clearsig.

Даже при выключенном режиме это происходит.

MEOE> Если блок не содержит символа ascii #231, пожалуйста, пришлите архив с исходным и подписанным текстами.

Обсуждение

[skipped due to its volume — write pgp@volga.net for more info]

Решение

Еще ждет своего героя. Я зарезервировал cb$50.00. Поскольку Русский Альбом PGP -- общественная служба, у меня нет возможности предоставлять призы, деноминированные в коммерческих валютах. Любые акты спонсирования и "призования" (не Альбома, а третьих лиц, способствующих распространению продуктов PGP в русскоязычных землях) будут всячески приветствоваться.

PGPfone

Телефонные разговоры и обмен электронной почтой во всевозрастающей степени подвержены подслушиванию. Практически любая незашифрованная электронная коммуникация может быть перехвачена. PGPfone защищает телефонные разговоры по каналам Интернет и телефонным линиям, используя самые стойкие из существующих криптографических технологий. Помимо этого, используя Интернет в качестве среды голосового общения, вы можете значительно снизить свои расходы по сравнению с использованием обычной телефонной связи.

Характеристики

PGPfone позволяет "говорить на ухо" по телефону, даже если это ухо расположено в тысячах миль от вас.

PGPfone (Pretty Good Privacy Phone) — это программный продукт, который превращает ваш персональный компьютер или ноутбук в защищенный телефон. Для того, чтобы предоставить возможность вести защищенные телефонные разговоры в реальном времени (по телефонным линиям и каналам Интернет) в нем используется технология сжатия звука и стойкие криптографические протоколы. Звук вашего голоса, принимаемый через микрофон, PGPfone последовательно: оцифровывает, сжимает, шифрует и отправляет тому, кто находится на другом конце провода и также использует PGPfone. Все криптографические протоколы и протокол сжатия выбираются динамически и незаметно для

пользователя, предоставляя ему естественный интерфейс, подобный обычному телефону. Для выбора ключа шифрования используются протоколы криптографии с открытым ключом, так что предварительного наличия защищенного канала для обмена ключами не требуется.

Все, что нужно для запуска PGPfone, это:

- по-настоящему надежный модем, поддерживающий скорость передачи как минимум 14.4 Kbps по протоколу V.32bis (рекомендуется 28.8 Kbps по протоколу V.34);
- IBM PC-совместимый компьютер с процессором как минимум 66 MHz 486 (рекомендуется Pentium), звуковой картой и динамиками или наушниками, работающий под управлением Windows 95 или NT,

  или
- Apple Macintosh с процессором 25MHz 68LC040 или старше (рекомендуется PowerPC) под управлением System 7.1 или старше с установленными Thread Manager 2.0.1, ThreadsLib 2.1.2, и Sound Manager 3.0 (все эти программы доступны с FTP-сервера Apple)— работа PGPfone на 68030 Mac не гарантируется, но в некоторых ситуациях она возможна; также, он запустится не на всех 68040, в зависимости от того, установлено ли соответствующее звуковое оборудование.

Для интересующихся технологией: PGPfone не требует предварительного наличия защищенного канала для обмена криптографическими ключами. Стороны обмениваются ключами с использованием протокола обмена ключами Диффи-Хеллмана, который не дает тому,

кто перехватывает разговор, получить какую-либо полезную информацию, и в то же время позволяет сторонам обменяться информацией для формирования общего ключа, который используется для шифрования и расшифровки речевого потока. В PGPfone версии 1.0 для аутентификации обмена ключами используется биометрическая подпись (ваш голос), для шифрования речевого потока — алгоритмы тройной DES, CAST или Blowfish, а для сжатия речи — алгоритм GSM.

PGPfone 1.0 для Macintosh и Windows 95/NT распространяется бесплатно.

Существует также коммерческая версия PGPfone 2.0 (только для Macintosh и доступная "легально" лишь американским и канадским покупателям).

Характеристики PGPfone 2.0

- Выбор технологии сжатия речевого потока (GSM, GSM Lite and ADPCM) с возможностью динамической ее смены без разрыва связи. Это позволяет достичь оптимального качества звука.
- Возможность защищенного обмена файлами.
- Телефонная записная книжка.

Требования к системе

- MacOs 7.5 или старше, PowerPC Macintosh; или 68040 Macintosh с тактовой частотой не менее 33Mhz (для повышения качества звука рекомендуется более быстрый процессор);

- микрофон и наушники вместо колонок — в полнодуплексном режиме это позволяет избежать наводок от акустического короткого замыкания;

Где взять PGPfone?

- ftp.ifi.uio.no (Норвегия)
- web.mit.edu (США)
- ftp.ifi.uio.no (Норвегия)
- web.mit.edu (США)
- www.pgp.com (США)

PGPsdk

Что такое PGPsdk?

PGPsdk — это средство разработки для программистов на C, позволяющее разработчикам программного обеспечения встраивать в него стойкие криптографические функции. PGPsdk использован при разработке PGP 5.5 и сопутствующих продуктов. 28 октября 1997 г. PGP, Inc. объявила о поставке PGPsdk сторонним производителям программного обеспечения.

Поддерживаемые алгоритмы

- Diffie-Hellman
- CAST
- IDEA
- 3DES
- DSS
- MD5
- SHA1
- RIPEMD-160

Поддержка RSA требует отдельного лицензирования.

Реализуемые функции

- Шифрование и аутентификация (с использованием перечисленных алгоритмов).
- Управление ключами (создание, сертификация, добавление/удаление со связки, проверка действительности, определения уровня надежности).
- Интерфейс с сервером открытых ключей (запрос, подгрузка, удаление и отзыв ключа с удаленного сервера).
- Случайные числа (генерация криптографически стойких псевдослучайных чисел и случайных чисел, базируясь на внешних источниках).
- Поддержка PGP/MIME.
- Вспомогательные функции.

Платформы

- 32-разрядные платформы Microsoft (Microsoft Visual C++ 5.0).
- Mac OS (MetroWerks CodeWarrior Version 12).
- Unix (Solaris и Linux)

Почтовые серверы открытых ключей PGP

Почтовые серверы открытых ключей PGP созданы, чтобы пользователи PGP могли обмениваться открытыми ключами через почтовые системы Интернет и UUCP. Пользователи, обладающие доступом к WWW, возможно, предпочтут использовать WWW-интерфейс, доступный по URL http://www.pgp.net/pgpnet/www-key.html, а менеджеры узлов, ожидающие частых запросов, возможно, предпочтут иметь локальную копию всей связки ключей, доступной по URL ftp://ftp.pgp.net:pub/pgp/

Этот сервис предназначен лишь для облегчения обмена ключами между пользователями PGP, и никаких попыток гарантировать валидность ключей не предпринимается. Для проверки валидности ключей используйте сертификаты (процедура описана в документации PGP).

Запросы

Все серверы обрабатывают запросы, отправляемые в форме почтовых сообщений. Команды серверу подаются в поле Subject: сообщения. Не включайте эти команды в тело сообщения!

To: pgp-public-keys@keys.pgp.net

From: johndoe@some.site.edu

Subject: help

Достаточно отправить ключ на один сервер. После того, как сервер обработает его, он автомагически переправит его остальным серверам

Например, чтобы отправить ваш ключ на сервер или обновить уже присутствующий там ключ, пошлите на любой из серверов сообщение такого типа:

To: pgp-public-keys@keys.pgp.net

From: johndoe@some.site.edu

Subject: add

-----BEGIN PGP PUBLIC KEY BLOCK-----

Version: 2.6

<...>

-----END PGP PUBLIC KEY BLOCK-----

Скомпрометированные ключи

Создайте сертификат отзыва ключа (процедура описана в документации PGP) и отправьте отозванный ключ на сервер командой ADD.

Допустимые команды

HELP Возвращает сообщение-подсказку на английском языке.

HELP страна Возвращает перевод сообщения-подсказки (DE, EN, ES, FI, FR, HR, NO)

ADD Добавляет открытый ключ PGP, содержащийся в теле сообщения

INDEX1 Возвращает список всех ключей, известных серверу

INDEX идентификатор Возвращает список всех ключей, содержащих идентификатор (-kv)

VERBOSE INDEX1 Возвращает список всех ключей в расширенном формате (-kvv)

VERBOSE INDEX идентификатор Возвращает список всех ключей в расширенном формате, содержащих идентификатор (-kvv)

GET1 Получить всю связку открытых ключей (split)

GET идентификатор Получить только ключ с указанным идентификатором (-kxa)

MGET выражение2,3 Получить все соответствующие выражению ключи. Выражение должно содержать как минимум два символа.

LAST n3 Получить все ключи, обновленные в последние n дней

1. Будьте готовы к получению больших объемов почты. В результате отправки этих команд вы получите огромный объем информации. Будьте осторожны: не все почтовые системы могут обрабатывать сообщения такого объема. В этом случае все сообщения будут отправлены назад. По всей вероятности, большая часть этой информации вам никогда не понадобится. На 2 февраля 1997 г. размер возвращаемых файлов был таким:

INDEX возвращал одно сообщение размером 4MB

VERBOSE INDEX возвращал одно сообщение размером 8MB

GET возвращал всю связку открытых ключей, содержащую более 55 000 ключей, всего 18MB, в виде 99 сообщений размером более 200KB каждое.

Скорее всего, большая часть полученной таким образом информации будет бесполезной, поэтому используйте эту команду в варианте с идентификатором. Это уменьшит размер сообщения и облегчит жизнь вашему почтовому администратору и администратору сервера, а также даст Вам более полезную выборку.

Важно: PGP работает с большими связками чрезвычайно медленно. Добавление всей связки, полученной с сервера, может занять несколько дней.

Если вам действительно нужен индекс всех ключей или вся связка, пожалуйста, загрузите ее с FTP-сервера (ftp://ftp.pgp.net/pub/pgp/keys/ или одного из национальных серверов).

2. Регулярные выражения в команде "MGET"

Вот примеры использования регулярных выражений в команде MGET:

MGET michael Возвращает все ключи, идентификатор пользователя которых содержит "michael"

MGET iastate Возвращает все ключи, идентификатор пользователя которых содержит "iastate"

MGET E8F605A5|5F3E38F5 Вернуть ключи с указанными двумя идентификаторами.

Одно замечание, касающееся выражений. Их синтаксис не совпадает с использованием "козырных" символов ("*" и "?") в оболочках UNIX и MSDOS. "*" означает не "что угодно", а "ноль или более вхождений предшествующего символа или метасимвола":

a.* — все, что начинается с "a"

ab*c — ac, abc, abbc и т.д.

Поэтому не используйте "MGET .*" — используйте вместо этого "GET".

3. Ограничения на количество возвращаемых ключей.

Некторые серверы налагают ограничение на количество возвращаемых в ответ на запросы "MGET" и "LAST" ключей, чтобы не завалить вас слишком большим их количеством в случае, если вы сделаете опечатку.

Пользователи должны использовать адрес pgp-public-keys@keys.pgp.net или адрес своего национального сервера:

- pgp-public-keys@keys.uk.pgp.net
- pgp-public-keys@keys.de.pgp.net
- pgp-public-keys@keys.no.pgp.net
- pgp-public-keys@keys.us.pgp.net
- pgp-public-keys@keys.nl.pgp.net
- pgp-public-keys@keys.fi.pgp.net
- pgp-public-keys@keys.es.pgp.net
- pgp-public-keys@keys.pt.pgp.net
- pgp-public-keys@keys.hr.pgp.net

для обращений по почте и ftp.pgp.net:pub/pgp/ для обращений по FTP.

Пользователям рекомендуется использовать перечисленные выше адреса "*.pgp.net", поскольку предполагается, что они устойчивы и надежны.

Использование PGP в Linux

PGP является средством, созданным для обеспечения защиты и аутентификации информации в таких ненадежных коммуникационных сетях, как Internet. Защита гарантирует, что только получатель информации может воспользоваться ей. Оказавшись в чужих руках, она будет совершенно бесполезной, поскольку ее нельзя будет декодировать.

Аутентификация гарантирует, что если некоторая ифнормация была создана лицом "A" , то она действительно поступила от "A" и не была никем сфальсифицирована или изменена в пути.

PGP основана на криптографической системе, известной как открытый ключ, которая может быть использована на ненадежных каналах. Это делает ее идеальной для обеспечения защиты информации, передаваемой по таким сетям, как Internet.

Для того, чтобы интересоваться защитой передаваемых вами данных и, следовательно, нуждаться в средстве криптографии, вовсе не обязательно участвовать в крутом промышленном шпионаже :) Нечто совсем простое, например E-mail, может быть самой настоящей причиной начать использовать PGP. Давайте разберемся почему:

Можно сравнить E-Mail с почтовыми карточками. Любой, кому она попадет в руки, может прочитать ее,

потому что нет физических преград, которые смогут помешать этому. С другой стороны, письмо в конверте более защищено. Можно держать конверт в руках, но нельзя прочитать письма. Если кто-то захочет прочитать его, ему придется порвать конверт.

Можно провести аналогию конверта с PGP, которое действует как дополнение к нашей E-Mail. PGP не позволяет никому прочитать сообщение, это может сделать только тот, кому оно предназначено; это одно из многих преимуществ PGP.

Как работает криптография открытого ключа

В системах с открытым ключом каждый человек имеет два ключа, взаимно дополняющих друг друга; один является открытым ключом, а другой закрытым.

Открытый ключ может и должен быть свободно доступным, так как он является именно тем ключом, который остальной мир использует для передачи вам информации. Однако открытый ключ не угрожает безопасности закрытого ключа.

Рассмотрим это на примере двух друзей, Хуана и Педро. Хуан может безопасно послать информацию Педро, если он знает его открытый ключ. С другой стороны, Педро, используя свой закрытый ключ, способен декодировать сообщение, которое послал Хуан. Предположим, что есть еще один человек, Маркос, который перехватывает сообщение, которое Хуан послал Педро. Маркос не может ничего сделать с сообщением, поскольку у него нет закрытого ключа Педро. Даже сам

Хуан, отправитель и создатель сообщения, не может декодировать его, это может сделать только Педро, при помощи закрытого ключа.

Безопасность системы основана на надежном хранении каждым пользователем своих закрытых ключей, даже в тех случаях, когда открытый ключ широко известен. Если кто-то попытается сломать систему, не зная закрытого ключа получателя, ему потребуется так много лет, что в конце концов информация окажется бесполезной.

Как было сказано во вступлении, помимо защиты, PGP дает возможность аутентифицировать информацию. Рассмотрим почему:

Наш открытый ключ служит не только для кодирования сообщений, но и для "подписи" посылаемой информации; полная аналогия с подписями, которые часто ставят на бумажных документах.

Подписанный цифровым образом без закрытого ключа документ может быть аутентифицирован любым человеком, обладающим открытым ключом. Такая аутентификация предоставляет средства, позволяющие проверить, действительно ли сообщение поступило от человека, указанного в качестве отправителя, и что оно не было изменено или фальсифицировано.

Для обеспечения защиты и аутентификации могут использоваться оба процесса, как кодирование, так и подпись. Сначала документ подписывается нашим закрытым ключом и затем кодируется с помощью открытого ключа получателя.

По получении сообщения получатель выполняет шаги в обратном порядке, сначала декодируя документ своим закрытым ключом и потом проверяя нашу подпись нашим открытым ключом.

Все эти процессы могут быть автоматизированы, это мы покажем позднее.

Открытый ключ хранится в так называемом сетрификате ключа, который является самим открытым ключом вместе с именем владельца и датой его создания.

Закрытый ключ защищен паролем, который предотвращает его несанкционированное использование.

Оба ключа хранятся в файле, известном как кольцо ключей, в котором также хранятся различные сертификаты ключей. Обычно есть кольцо для открытых ключей и кольцо для закрытых.

Ключи имеют внутренний идентификатор ключа, который состоит из 64 последних бит ключа. При отображении информации о ключе на самом деле показываются последние 32 бита ключа. Эти идентификаторы ключа используются PGP, например, для определения ключа при декодировании сообщения.

При подписывании документа PGP формирует 128 бит, которые представляют документ. Эта подпись является своего рода контрольной суммой, или CRC, которая позволяет обнаружить изменения в документе. В отличие от обычных CRC или контрольных сумм, никто не может заново создать эту подпись чтобы узаконить любые изменения исходного документа. Подпись создается при помощи закрытого ключа отправителя и тот, кто хочет внести изменения, не имеет к нему доступа.

Версии PGP

Теперь, когда вы знаете, для чего нужен PGP, вы наверняка захотите им воспользоваться.

На этом этапе необходимо еще раз сказать о большой путанице, окружающей различные версии PGP. В виду политики Соединенных Штатов в отношении экспорта криптографического материала, появились несколько версий PGP, вместе с несколькими законами по их использованию. Я постараюсь прояснить всю эту неразбериху и перечислю различные версии, существующие на сегодняшний день.

PGP 2.3a

Это "классический" PGP. Его все еще можно использовать, хотя из-за несовместимости могут возникнуть некоторые проблемы при обработке ключей и сообщений, созданными версиями 2.6.x и более поздними, использующими ключи длиннее 1280 бит. Предполагается, что версия 2.3a не может использоваться за пределами США из-за патентных ограничений.

PGP 2.6ui

Это неофициальная версия PGP 2.3a, которая устраняет указанные выше проблемы несовместимости. Эта версия не является версией 2.6.x так как она базируется на исходных кодах 2.3a

PGP 2.62ui

Она основана на исходных кодах 2.6ui и является модификацией, в которой попытались достичь совместимости с последнимий нововведениями, появившимися в версиях 2.6.x.

MIT PGP 2.6.2

Это последняя официальная версия PGP. Ее сообщения можно прочитать предыдущими версиями до 2.5 и она использует библиотеку кодирования RSAREF. Экспортировать эту версию за пределы США незаконно, но, что любопытно, если она экспортирована, то ее можно свободно использовать.

PGP 2.6.3i

Основана на исходных кодах MIT PGP 2.6.2, которые были модифицированы для международного использования. В частности, не используется упомянутая выше библиотека кодировани RSAREF. Использование этой версии в США незаконно.

PGP 5.0

PGP 5.0 (ранее известная как PGP 3.0) является абсолютно новой версией PGP. Ее исходные коды были написаны абсолютно независимо. Добавлены новые опции, включая поддержку других криптографических алгоритмов, помимо RSA и IDEA. В нее входит графический пользовательский интерфейс для упрощения ее использования. Эта версия будет доступна в середине лета.

ViaCrypt PGP 2.7.1 y 4.0

Так как она является коммерческой, в поставку входит руководство и лицензия на личное использование. Исходные коды в поставку не входят.

PGP 4.5 и 5.0

В июне 1996 PGP Inc. купила ViaCrypt и начала разработку коммерческих версий PGP для Соединенных

Штатов и Канады. Самой последней версией является PGPMail 4.5.

Необходимо также учитывать, что в некоторых странах, например во Франции, Иране, Ираке, России и Китае, использование криптографии регулируется законодательством или запрещено.

Некоторые ссылки для различных дистрибутивов Linux

Бинарные файлы в формате ELF:
tonelli.sns.it/pub/Linux/pgp/pgp263.is.bin.tgz

Бинарные файлы и исходные коды для Red Hat:
ftp.replay.com/pub/replay/ub/redhat/i386/pgp-2.6.3i-1.i386.rpm

ftp.replay.com/pub/replay/ub/redhat/SRPMS/pgp-2.6.3i-1.src.rpm

Исходные коды:
ftp.dit.upm.es/mirror/ftp.ifi.uio.no/pub/pgp/src/pgp263is.tar.gz

Установка PGP

Предположим, вы раздобыли PGP. Также предположим, что вы загрузили исходный код версии 2.6.3i и что на вашем жестком диске находится файл pgp263is.tar.gz

Первым шагом будет создать каталог для исходных кодов:

mkdir pgp

Далее разворачиваем архив:

tar -C ./pgp -xzvf pgp263is.tar.gz

Теперь переходим в только что созданный каталог:

cd pgp

Теперь разворачиваем файл pgp263ii.tar, котором находится документация и исходный код программы. Это выполняется командой:

tar -xvf pgp263ii.tar

Сейчас вы готовы к компиляции PGP. Если вы загрузили не исходный код, а скомпилированную версию (a.out или ELF), вы можете пропустить этот этап. Если вы подготовили исходный код, то компиляция выполняется следующими командами:

cd src

make linux

Если все прошло хорошо, то makefile создаст исполняемый файл pgp. В случае глобальной установки вы можете скопировать его в /usr/local/bin, /usr/bin или куда пожелаете. Или вы можете оставить его в вашем домашнем каталоге.

Аналогично, файл справки pgp.1 копируется в /usr/man/man1 в случае глобальной установки.

Основная конфигурация

По умолчанию PGP ищет кольца ключей и некоторые конфигурационные файлы в каталоге ~/.pgp,

поэтому первым шагом мы создадим этот каталог в нашем HOME:

cd

mkdir .pgp

Обратимся к дистрибутиву и найдем файл config.txt, который отвечает за конфигурацию некоторых аспектов PGP. Чтобы у вас была личная конфигурация, вы должны скопировать этот файл в ваш только что созданный ~/.pgp.

Или, вместо ~/.pgp/config.txt, вы можете переименовать его в .pgprc и сохранить в вашем домашнем каталоге, то есть ~/.pgprc.

Среди прочего, этот файл может определять используемый язык при помощи параметра Languaje, возможные варианты:

Languaje = en (Английский)

Languaje = es (Испанский)

Languaje = ja (Японский)

В этом файле есть другие параметры. Чтобы воспользоваться преимуществами этой опции, вы должны скопировать файл languaje.txt в ~/.pgp

Далее рекомендуется скопировать персонализированный файл справки на вашем языке в ~/.pgp. В случае испано-говорящих пользователей можно скопировать файл es.hlp.

Создание пары ключей

Чтобы начать использовать PGP, нужно создать вашу собственную пару ключей (открытый/закрытый). Чтобы это сделать, выполните команду:

pgp -kg

Вас попросят выбрать максимальный размер ключа (512, 768 или 1024 байта), чем больше ключ, тем более надежным он будет, правда ценой небольшого снижения быстродействия.

После выбора размера ключа вас попросят задать идентификатор открытого ключа. Обычно здесь люди указывают свои имена или e-mail адрес. В моем случае, я написал:

Angel Lopez Gonzalez <alogo@mx2.redestb.es>

Далее идет пароль, который будет защищать ваш закрытый ключ. Выберите фразу, которую вы легко сможете запомнить. Это необходимо для защиты закрытого ключа. Например, если кто-нибудь украдет его, он будет бесполезен без пароля.

Наконец, программа попросит вас в произвольном порядке нажать несколько клавиш на клавиатуре чтобы она могла создать последовательность случайных чисел. Программа задает последовательность бит на основе интервалов между нажатиями клавиш.

Через несколько секунд PGP создаст ключи и известит вас об этом сообщением. После того, как ключи были сгенерированы должным образом, их необходимо сохранить в каталоге ~/.pgp в виде файлов: pubring.pgp и secring.pgp

Первый, pubring.pgp, является кольцом с открытым ключем. На данный момент в нем хранится только ваш ключ.

Второй, secring.pgp является, как вы можете понять, кольцом закрытых ключей, на данный момент в нем содержится только ваш закрытый ключ.

Необходимо помнить, что безопасность методов открытого ключа опирается на безопасность закрытого ключа; поэтому, обязательно храните его в надежном месте и следите за тем, чтобы никто не смог его получить из кольца закрытых ключей. Проверьте права доступа к secring.pgp и установите такие права доступа, чтобы только вы могли читать и записывать, причем остальные не должны иметь доступ вообще.

Наконец, необходимо упомянуть, что редактировать и изменять и идентификаторы ключей и пароли закрытых ключей можно с помощью команды:

pgp -ke идентификатор [кольцо]

Добавление ключей к кольцу

Теперь вам, вероятно, захочется добавить открытые ключи ваших друзей к вашему кольцу. Для этого вам потребуется получить эти кольца: с сервера ключей, непосредственно от этого человека, при помощи команды finger, по e-mail, и т.д. Вспомним, что открытые ключи распространяются свободно и нет необходимости передавать их по безопасному каналу, как в случае с криптологическими методами с одним ключем.

Если в вашем файле Somekey.pgp содержится ключ и вы хотите добавить его в ваши кольца, процедура очень проста:

pgp -ka Somekey [кольцо]

По умолчанию расширение .pgp указывает на файл с ключем и имена pubring.pgp и secring.pgp даются файлам, содержащим кольца открытых и закрытых ключей, соответственно.

После добавления ключа PGP может сообщить вам, что добавленный ключ не полностью сертифицирован; это означает, что данный ключ не обязательно может принадлежать заявленному владельцу.

Если есть "уверенность", что ключ действительно принадлежит этому человеку, или потому что он или она дали его вам лично или по безопасному каналу, то вы сами можете сертифицировать его. Это означает, что мы удостоверяем сертифицированность ключа.

Это облегчает передачу нашего ключа человеку, который нам доверяет и абсолютно уверен в том, что мы передали ему правильный ключ.

Для этого процесса придумали даже имя, доверие в сети. В Соединенных Штатах пользователи PGP даже устраивают собирания для обмена открытыми ключами и их подписи.

Рассмотрим эту концепцию на примере. Возьмем все тех же двух друзей, Хуана и Педро. Хуан дает свой открытый ключ Педро. Педро уверен, что ключ, который ему дал Хуан, верен, так как они доверяют друг другу. Когда он приходит домой, он добавляет его к своему кольцу открытых ключей, и он может его

сертифицировать, поскольку ключ действительно принадлежит Хуану, поэтому он подписывает его своим закрытым ключом.

Теперь на сцене появляются еще два человека: Луис и Мария. Луис получает от Педро ключ Хуана и позднее пересылает его Марии. Мария не доверяет Луису, но видит, что ключ Хуана сертифицирован Педро. Мария может проверить открытый ключ Хуана благодаря подписи Педро. У нее есть открытый ключ Педро, который он дал ей лично, поэтому она может доверять ключу Луиса, проверив подлинность подписи Педро. Теперь мы знаем, как Мария может довериться ключу, данному ей такой ненадежной личностью, как Луис.

Это запутано, но необходимо для защиты единственного слабого места этого типа криптографии: факта фальсификации открытого ключа.

Удаление ключа из кольца

Продолжим наше небольшое путешествие по PGP. Следующим шагом после того, как к кольцу были добавлены ключи, мы узнаем как их удалить. Это можно сделать командой:

pgp -kr идентификатор [кольцо]

Например: "pgp -kr juan " удалит любой ключ, у которого в идентификаторе содержится "juan". По умолчанию исследуется кольцо открытых ключей.

Выделение ключа

После сохранения ключей друзей в вашем открытом кольце нам неоходимо послать им свой открытый ключ. Прежде всего его необходимо выделить из кольца:

pgp -kx идентификатор файл [кольцо]

Например: "pgp -kx angel mykey" выделяет открытый ключ, идентифицированный подстрокой "angel" в файле mykey.

Созданный файл mykey.pgp не в формате ASCII (попробуйте использовать cat для его просмотра). Однако, если кому-нибудь потребуется создать файл ключа в формате ASCII чтобы послать, к примеру, по e-mail, или добавить дополнительную информацию к базе данных finger, ему потребуется напечатать:

pgp -kxa identifies file [ring]

Например: "pgp -kxa angel mykey" выделяет открытый ключ, идентифицированный подстрокой "angel", в файл "mykey.asc".

Вместе с ключом также выделяются все сертификаты, которые его подтверждают.

Содержание кольца

Чтобы просмотреть ключи, содержащиеся в кольце, наберите команду:

pgp -kv [идентификатор] [кольцо]

Еще раз заметим, что кольцом по умолчанию является pubring.pgp, открытое кольцо.

Если идентификатор не указан явно, то показываются все ключи кольца.

Чтобы просмотреть все сертификаты каждого ключа, необходимо набрать:

pgp -kvv [идентификатор] [кольцо]

Кодирование сообщения

Мы рассмотрели, как использовать ключи. Теперь давайте попробуем использовать это для чего-нибудь интересного. Давайте посмотрим, как декодировать файл:

pgp -e файл идентификатор

Приведем пример: учитель хочет послать своему коллеге экзаменационные задачи по e-mail и не хочет, чтобы студенты перехватили это сообщение :). Пусть имя второго учителя будет Маркос и идентификатор его открытого ключа содержит его имя. Первый учитель наберет:

pgp -e exam.doc marcos

Эта команда создает файл с именем exam.pgp, содержащий файл exam.doc, закодированный так, что только Маркос может его декодировать с помощью своего закрытого ключа.

Помните, что созданный файл, exam.pgp, не является ASCII файлом, поэтому для отправки его по E-Mail может потребоваться добавить еще одну опцию -a для того, чтобы выходной закодированный файл был в формате ASCII, например так:

pgp -ea exam.doc marcos

По причинам безопасности, нам иногда может потребоваться удалить оригинал. PGP может делать это автоматически при помощи опции -w:

pgp -eaw exam.doc marcos

Кодирование сообщения для нескольких получателей

Теперь представьте, что наш учитель хочет послать эти задачи своим коллегам по кафедре. Чтобы это сделать, ему необходимо просто набрать вместо одного несколько идентификаторов:

pgp -ea exam.doc marcos juan alicia

Заметьте, что опция -a тоже используется, поэтому выходной файл будет в формате ASCII и его можно будет послать по e-mail.

Как сообщение подписывается

Как уже говорилось, цифровая подпись в сообщении является аналогом обычной подписи на бумаге. Подпись документа позволит получателю удостовериться в его аутентичности и в том, что сообщение не было изменено.

Чтобы подписать документ, необходимо использовать ваш закрытый ключ:

pgp -s файл [-u идентификатор]

Если у нас есть несколько закрытых ключей в нашем secring.pgp, мы можем выбрать один из них при помощи идентификатора.

Когда наш учитель из примера решает подписать экзаменационные задачи, чтобы сообщить, что их послали не студенты-шутники :) он набирает следующее:

pgp -s exam.doc

Эта команда создает файл с именем exam.doc.pgp, который не является ASCII-текстом, потому что PGP пытается сжать файл. Если, с другой стороны, вы хотите подписать файл, оставив текст читабельным и с подписью в конце, то процедура будет выглядеть:

pgp -sta exam.doc

Эта последняя команда очень полезна при подписи электронной почты, которую и дальше можно будет читать без использования PGP или тем, кому не хочется проверять подпись.

Кроме того, можно подписать документ и затем закодировать его при помощи следущей команды:

pgp -es файл идентификатор_получателя

[-u мой_идентификатор]

Например:

pgp -es exam.doc marcos -u angel

Здесь файл exam.doc кодируется и подписывается и сохраняется в файле exam.pgp. Для кодирования файла используется открытый ключ, идентифицируемый подстрокой "marcos", поэтому только этим ключом можно декодировать этот файл. Затем я идентифицирую мой закрытый ключ строкой "angel", так как в моем кольце есть несколько ключей.

Даже в этом случае можно создать файл в формате ASCII, используя опцию -a.

Кроме того, нас может заинтересовать возможность создания подписи файла отдельно от данных. Чтобы это сделать, воспользуемся опцией -b:

pgp -sb exam.doc

Эта команда создает новый файл exam.sig, содержащий только подписи.

Декодирование

Для декодирования файла и/или проверки его подписи используется команда:

pgp входной_файл [-o выходной_файл]

По умолчанию предполагается, что входной файл имеет расширение .pgp. Выходной файл является необязательным параметром и будет содержать декодированный файл. Если выходной файл не указан, декодированный файл будет сохранен в файле входной_файл без расширения .pgp.

Однако, после декодирования файла нам необходимо указать стандартный выход для декодированного файла. Это достигается использованием опции -m:

pgp -m файл

Существует еще одна возможность -- использовать каналы ввода и вывода с опцией -f:

pgp -fs идентификатор < входной_файл > выходной_файл

Еще одним интересным сценарием является декодирование подписанного сообщения, посланного нам кем-нибудь, с сохранением подписи, например для кодирования его еще раз для того, чтобы послать его кому-нибудь еще. Чтобы это сделать, нужно использвать опцию -d:

pgp -d exam

Здесь мы берем файл exam.pgp и декодируем его, но при этом оставляем оригинальную подпись в файле. Теперь можно переходить к кодированию его открытым ключом того человека, который после получения может проверить аутентичность исходного сообщения.

Обработка текстовых файлов

Часто PGP используется для кодирования электронной почты, которая чаще всего представляет собой текст. Проблема текстовых файлов заключается в том, что на разных машинах текст представляется по разному; например в MSDOS все строки заканчиваются символами возврата каретки и перевода строки, в Линуксе только перевод строки, в Macintosh только возврат каретки... и т.д. Чтобы избежать несовместимости платформ, нам необходимо сказать PGP, что мы хотим закодировать текстовый файл, а не бинарный файл, с тем, чтобы после разархивирования его можно было адаптировать к особенностям платформы получателя. Для кодирования текстового файла для e-mail используется опция -t. Например:

pgp -sta текстовый_файл идентификатор

"Отпечатки (fingerprints)"

Отпечаток является последовательностью из 16 бит, которая идентифицирует ключ уникальным образом. Можно проверить, принадлежит ли имеющийся у вас ключ именно тому человеку, сравнив каждый из 16 бит вместо всех 1024 байт, которые составляют ключ.

Для просмотра отпечатка ключа используется команда:

pgp -kvc идентификатор [кольцо]

Использование PGP в командной строке

PGP имеет опции, которые особенно полезны при использовании PGP в командной строке в автоматизирующих скриптах.

+batchmode

При использовании этой опции PGP не будет спрашивать ничего сверх крайне необходимого. Используйте эту опцию для автоматической проверки подписи. При отсутствии подписи в файле возвращается код ошибки 1; если файл подписан и подпись правильна, то возвращается 0.

pgp +batchmode файл

force

Использование этой опции одобряет любую операцию по переписыванию файла или удалению ключа.

pgp +force +kr marcos

В командной строке желательно обойтись без запросов паролей при кодировании файла. Например, чтобы избежать вопросов во время кодирования мы может просто обойти это задав переменную окружения PGP-PASS.

Здесь приведен пример:

PGPPASS="пароль"

export PGPPASS

pgp -s file.txt marcos

Еще одним способом передачи пароля PGP в не-интерактивном режиме является использование опции -z.

Как здесь:

pgp -sta exams.txt angel -z "пароль"

Еще одна полезная операция в командной строке — это изменение "разговорчивости" PGP при помощи опции +verbose. Она задает тихий режим -- то есть отсутствие информационных сообщений, только сообщения об ошибках:

pgp file.pgp +verbose=0

Интеграция в почтовые клиенты

Интеграция PGP в почтовые клиенты для автоматического кодирования, декодирования и подписи проста и почти не зависит от используемого почтового клиента.

В качестве примера я расскажу про интеграцию PGP в Pine. Надеюсь, читатель использует именно этот почтовый клиент.

Хотя я буду описывать работу PGP с Pine, основные принципы применимы ко всем другим клиентам. Конфигурация, конечно, будет отличаться для каждой почтовой программы.

Для автоматической декодировки почты перед чтением необходим фильтр для обработки сообщения и вывода его на экран. Кроме этого, можно создать макрос, который объединит декодирование и вывод на экран.

В случае Pine, у него есть опция для определения фильтров, которые выполняются до вывода сообщения на

экран. Эта опция называется 'display-filters' и находится в конфигурационном меню Pine. В эту опцию мы добавим новый фильтр, который выглядит так:

_BEGINNING("-----BEGIN PGP MESSAGE-----")

_ /usr/local/bin/pgp

Каждое закодированное PGP сообщение заключается двумя определенными строчками -- "-----BEGIN PGP MESSAGE-----" и "-----END PGP MESSAGE-----" -- с тем, чтобы если вы захотите узнать, имеет ли сообщение в теле закодированный текст, то достаточно найти одну из указанных выше строчек. Фильтр, определенный в Pine, делате именно это. Перед отображением самого сообщения он проверяет тело сообщения на наличие строки "-----BEGIN PGP MESSAGE-----" с тем лишь ограничением, что она должна быть в начале какой-либо строки. Если он находит ее, он выполняет программу: /usr/local/bin/pgp

Затем, если в теле действительно есть закодированное сообщение, будет выполнено декодирование PGP. У вас спросят пароль и вы сможете прочитать сообщение. Если вы хотите еще больше автоматизировать этот процесс, уменьшив время, необходимое на то, чтобы каждый раз указывать пароль, то вам потребуется определить переменную среды PGP-PASS или использовать опцию -z как было показано выше.

Теперь нам требуется только задать фильтр, который закодирует наше сообщение с открытыми ключами получателей из нашего открытого кольца до отправки сообщения.

Pine помогает нам еще раз, в нем есть конфигурационная опция 'sending-filters.' Ниже приведен фильтр, который надо задать для этой опции:

/usr/local/bin/pgp -etaf _RECIPIENTS_

После написания сообщения и нажатия CTRL-X для отправки, Pine спросит нас, хотим ли мы послать его сразу, без применения заданных фильтров. Чтобы послать сообщение без кодирования, просто ответьте утвердительно, но если вы хотите послать сообщение закодированным, то нажмите CTRL-N или CTRL-P, далее вам предложат список всех заданных фильтров. В нашем случае это будет только фильтр PGP, приведенный выше.

PGP Enterprise Security 3.0

В сентябре 1998 года компания Network Associates начала поставки криптографического пакета PGP Enterprise Security 3.0, новой версии популярной программы Pretty Good Privacy для шифрования электронной почты и файлов, разработанной специально для корпораций.

Новая версия программного обеспечения поддерживает цифровые сертификаты X.509, что позволит программе PGP взаимодействовать с более широким кругом средств корпоративной безопасности. Новая версия также более тесно интегрирована с сервером компании и инструментарием управления, что облегчит администраторам контроль за системой безопасности предприятия. Разработанная семь лет назад Филом Зиммерманом (Phil Zimmermann) как свободно распространяемая криптопрограмма, PGP долгое время рассматривалась как средство для ученых и энтузиастов-одиночек в Internet. Но, получив в прошлом году заказов на PGP на 36 млн дол., в Network Associates намерены изменить подобную точку зрения.

"Изначально, PGP предназначалась для индивидуального пользователя, так что масштаб возможных корреспондентов не превышал пары сотен человек. Новая же версия продукта может масштабироваться до сотен тысяч пользователей", —

сообщил Джефф Хэрелл (Jeff Harrell), менеджер по продуктам безопасности компании Network Associates.

В настоящее время в PGP используется собственная технология обмена открытым ключом, но протокол X.509 позволит пользователям взаимодействовать с более широким спектром межсетевых экранов и средств организации виртуальных частных сетей. Кроме того, новые возможности PGP Enterprise Security 3.0 обеспечивают пользователям-клиентам защищенные функции печати и разделения файлов.

Новая версия продукта теперь может использоваться с более широким спектром программ электронной почты, включая продукты от Novell, Microsoft и Qualcomm.

В PGP Enterprise Security 3.0 реализована возможность "рассечения ключа", что позволяет корпорации восстанавливать данные, если исходный криптографический ключ по тем или иным причинам недоступен. В PGP пользователи могут "рассечь" и поделить частный ключ среди группы людей количеством до 10 человек.

В состав пакета PGP Enterprise Security 3.0 входят клиентская криптопрограмма PGP Desktop Security 6.0, средство управления Policy Management Agent, серверная программа Certificate Server, а также инструментарий разработчика программного обеспечения.

Программа PGPfreeware 6.0 — бесплатно распространяемый клиентский компонент пакета PGP Enterprise Security 3.0 — появилась в Internet еще до того, как о ее выпуске сообщила американская фирма-производитель Network Associates (на сайте

www.scramdisk.clara.net, откуда ее может свободно скачать любой желающий).

Подобно всем остальным программам с сильной криптографией (где длина криптоключа превышает 40 бит), PGPfreeware 6.0 является в США предметом строгих экспортных ограничений. Однако, это уже далеко не первый случай, когда популярное шифрсредство очень быстро просачивается за границу благодаря Сети. Как прокомментировал представитель Network Associates, "это случается всякий раз, когда выходит очередная версия данного продукта. И происходит это несмотря на все предпринимаемые нами меры предосторожности".

Тонкости и хитрости

Установка и применение программы PGP

В PGP применяется принцип использования двух взаимосвязанных ключей: открытого и закрытого. К закрытому ключу имеете доступ только вы, а свой открытый ключ вы распространяете среди своих корреспондентов.

Великолепное преимущество этой программы состоит также в том, что она бесплатная и любой пользователь, имеющий доступ к Интернету, может ее «скачать» на свой компьютер в течение получаса. PGP шифрует сообщение таким образом, что никто кроме получателя сообщения, не может ее расшифровать. Создатель PGP Филипп Циммерман открыто опубликовал код программы, который неоднократно был исследован специалистами крипто-аналитиками высочайшего класса и ни один из них не нашел в программе каких-либо слабых мест.

Филипп Циммерман следующим образом объясняет причину создания программы: «Людям необходима конфиденциальность. PGP распространяется как огонь в прериях, раздуваемый людьми, которые беспокоятся о своей конфиденциальности в этот информационный век. Сегодня организации по охране прав человека используют программу PGP для защиты своих людей за рубежом. Организация Amnesty International также использует ее».

Пользователям сети «Интернет» рекомендуется использовать эту программу именно по той же причине, почему люди предпочитают посылать друг другу письма в конвертах, а не на открытках, которые могут быть легко прочитаны почтовыми служащими. Дело в том, что электронные сообщения, в том виде и формате, который существует на сегодняшний день, легко могут быть прочитаны и архивированы любым человеком, имеющим доступ к серверу Интернет провайдера (поставщика услуг сети «Интернет»). В настоящий момент спецслужбам проще и дешевле подключиться к электронным адресам большого количества лиц, нежели к телефонным разговорам. Здесь вообще ничего делать не надо. Все сделает компьютер. Агенту спецслужбы или другому заинтересованному человеку остается только сесть за компьютер и просмотреть все ваши сообщения. Научно-технический прогресс облегчил задачу таким людям, однако, этот же самый прогресс предоставил возможность пользователям сети «Интернет» скрыть свои сообщения от третьих лиц таким образом, что даже суперкомпьютер стоимостью несколько десятков миллионов долларов не способен их расшифровать.

Как?

Когда пользователь шифрует сообщение с помощью PGP, то программа сначала сжимает текст, что сокращает время на отправку сообщения через модем и увеличивает надежность шифрования. Большинство приемов криптоанализа (взлома зашифрованных сообщений) основаны на исследовании «рисунков», присущих текстовым файлам, что помогает взломать ключ. Сжатие

ликвидирует эти «рисунки» и таким образом повышает надежность зашифрованного сообщения. Затем PGP генерирует сессионный ключ, который представляет собой случайное число, созданное за счет движений вашей мышки и нажатий на клавиши клавиатуры.

Как только данные будут зашифрованы, сессионный ключ зашифровывается с помощью публичного ключа получателя сообщения, который отправляется к получателю вместе с зашифрованным текстом.

Расшифровка происходит в обратной последовательности. Программа PGP получателя сообщения использует закрытый ключ получателя для извлечения временного сессионного ключа, с помощью которого программа затем дешифрует зашифрованный текст.

Ключи

Ключ – это число, которое используется криптографическим алгоритмом для шифрования текста. Как правило, ключи — это очень большие числа. Размер ключа измеряется в битах. Число, представленное 1024 битами – очень большое. В публичной криптографии, чем больше ключ, тем его сложнее взломать.

В то время как открытый и закрытый ключи взаимосвязаны, чрезвычайно сложно получить закрытый ключ исходя из наличия только открытого ключа, однако это возможно при наличии большой компьютерной мощности. Поэтому крайне важно выбирать ключи подходящего размера: достаточно большого для обеспечения безопасности и достаточно малого для

обеспечения быстрого режима работы. Кроме этого, необходимо учитывать личность того, кто намеревается прочитать ваши зашифрованные сообщения, насколько он заинтересован в их расшифровке, каким временем он обладает, и какие у него имеются ресурсы.

Более большие ключи будут более надежными в течение более длительного срока времени. Поэтому если вам необходимо зашифровать информацию с тем, чтобы она хранилась в течение нескольких лет, то необходимо использовать более крупный ключ.

Ключи хранятся на жестком диске вашего компьютера в зашифрованном состоянии в виде двух файлов: одного для открытых ключей, а другого — для закрытых. Эти файлы называются «кольцами» (keyrings). В течение работы с программой PGP вы, как правило, будете вносить открытые ключи ваших корреспондентов в открытые «кольца». Ваши закрытые ключи хранятся в вашем закрытом «кольце». При потере вашего закрытого «кольца» вы не сможете расшифровать любую информацию, зашифрованную с помощью ключей, находящихся в этом «кольце».

Цифровая подпись

Огромным преимуществом публичной криптографии также является возможность использования цифровой подписи, которая позволяют получателю сообщения удостовериться в личности отправителя сообщения, а также в целостности (верности) полученного сообщения. Цифровая подпись исполняет ту же самую функцию, что и ручная подпись. Однако ручную подпись легко подделать.

Цифровую же подпись почти невозможно подделать.

Хэш-функция

Еще одно важное преимущество использования PGP состоит в том, что PGP применяет так называемую «хэш-функцию», которая действует таким образом, что в том случае какого-либо изменения информации, пусть даже на один бит, результат «хэш-функции» будет совершенно иным. С помощью «хэш-функции» и закрытого ключа создается «подпись», передаваемая программой вместе с текстом. При получении сообщения получатель использует PGP для восстановления исходных данных и проверки подписи.

При условии использования надежной формулы «хэш-функции» невозможно вытащить подпись из одного документа и вложить в другой, либо каким-то образом изменить содержание сообщения. Любое изменение подписанного документа сразу же будет обнаружено при проверке подлинности подписи.

Парольная фраза

Большинство людей, как правило, знакомы с парольной системой защиты компьютерных систем от третьих лиц.

Парольная фраза – это сочетание нескольких слов, которое теоретически более надежно, чем парольное слово. В виду того, что парольная фраза состоит из нескольких слов, она практически неуязвима против так называемых «словарных атак», где атакующий пытается разгадать ваш пароль с помощью компьютерной

программы, подключенной к словарю. Самые надежные парольные фразы должны быть достаточно длинными и сложными и должны содержать комбинацию букв из верхних и нижних регистров, цифровые обозначения и знаки пунктуации.

Парольная фраза должна быть такой, чтобы ее потом не забыть и чтобы третьи лица не могли ее разгадать. Если вы забудете свою парольную фразу, то уже никогда не сможете восстановить свою зашифрованную информацию. Ваш закрытый ключ абсолютно бесполезен без знания парольной фразы и с этим ничего не поделаешь.

Основные шаги

1. Установите программу на свой компьютер. Руководствуйтесь краткой инструкцией по инсталляции программы, приведенной ниже.

2. Создайте закрытый и открытый ключ. Перед тем, как вы начнете использовать программу PGP, вам необходимо генерировать пару ключей, которая состоит из закрытого ключа, к которому имеете доступ только вы, и открытый ключ, который вы копируете и свободно передаете другим людям (вашим корреспондентам).

3. Распространите свой открытый ключ среди своих корреспондентов в обмен на их ключи. Ваш открытый ключ, это всего лишь маленький файл, поэтому его можно либо воткнуть в сообщение, копировать в файл, прикрепить к почтовому сообщению или разместить на сервере.

4. Удостовериться в верности открытого ключа. Как только вы получите открытые ключи своих

корреспондентов, то их можно внести в «кольцо» открытых ключей. После этого вам необходимо убедиться в том, что у вас действительно открытый ключ вашего корреспондента. Вы можете это сделать, связавшись с этим корреспондентом и, попросив его зачитать вам по телефону «отпечатки пальцев» (уникальный идентификационный номер) его открытого ключа, а также сообщив ему номер вашего ключа. Как только вы убедитесь в том, что ключ действительно принадлежит ему, вы можете его подписать и таким образом подтвердить ваше доверие к этому ключу.

5. Шифрование и удостоверение корреспонденции вашей цифровой подписью. После генерации пары ключей и обмена открытыми ключами вы можете начать шифрование и удостоверение ваших сообщений и файлов своей цифровой подписью. Если вы используете почтовую программу, которая поддерживается программой PGP, то вы можете шифровать и дешифровать всю вашу корреспонденцию, находясь прямо в этой программе. Если же ваша почтовая программа не поддерживается программой PGP, то вы можете шифровать вашу корреспонденцию другими способами (через буфер обмена или шифрованием файлов целиком).

6. Дешифровка поступающих к вам сообщений и проверка подлинности отправителя. Когда кто-либо высылает вам зашифрованное сообщение, вы можете дешифровать его и проверить подлинность отправителя этого сообщения и целостность самого сообщения. Если ваша почтовая программа не поддерживается PGP, то вы можете сделать это через буфер обмена.

7. Уничтожение файлов. Когда вам необходимо полностью удалить какой-либо файл, вы можете исполнить команду wipe (стереть). Таким образом, удаленный файл уже невозможно будет восстановить.

Инсталляция

Ниже приводятся заголовки сообщений, появляющиеся при инсталляции программы (нажатии на инсталляционный файл с расширением .exe) и команды, которые необходимо исполнять при инсталляции:

PGP Installation program
Нажмите на Next

Software License agreement
Нажмите на Yes

User information
Name_____
Company _____
Введите свое имя, название компании и нажмите на Next

Setup: choose installation directory
Нажмите на Next

Select components:
Здесь необходимо выбрать компоненты для установки:

- Program files
- Eudora Plugin

- Microsoft Exchange/Outlook plugin
- Microsoft Outlook Express plugin
- User's manual Adobe
- PGP disk for Windows

Выделите те компоненты, которые необходимо установить. Если вы не используете почтовую программу Eudora, то ее не нужно выделять. Если вы используете Microsoft Exchange/Outlook для работы в сети «Интернет», то выделите ее. То же самое касается Microsoft Outlook, почтовой программы, встроенной в Windows-98.

Нажмите на Next

Check setup information
Нажмите на Next

Начинается копирование программных файлов на жесткий диск компьютера.

Для того чтобы программа автоматически запустила операцию создания ключей после перезагрузки компьютера нажать на кнопку "Yes I want to run PGP keys"

Нажмите на Finish.

Restart Windows для перезагрузки Windows
Нажмите на OK

Компьютер перезапустится и на этом программа установки завершится.

Теперь необходимо установить на компьютер два ключа:

public key — открытый ключ

private key — закрытый ключ

Генерим ключи

После перезагрузки компьютера в нижнем правом углу (панель задач) появится значок PGP — символ амбарного замка. Поставьте на него мышку, нажмите на мышку и выберите в открывшемся меню команду Launch PGP keys.

- Зайдите в меню KEYS и выполните команду NEW KEY
- Нажмите на next
- Введите свое имя и электронный адрес
- Нажмите на next
- Выберите размер ключа 2048 и нажмите на next
- Затем выделите фразу key pair never expires (срок действия ключевой пары никогда не истекает) и нажмите на next.
- Два раза введите секретный пароль и нажмите на next.

Программа начнет генерировать пару ключей. Если программе не хватает информации, то она может попросить нажать на несколько клавиш наугад и подвигать мышку. Это необходимо выполнить.

Затем программа сообщит, что процесс генерации ключей закончен.

- Нажмите на next.
- Потом еще раз нажмите на next.
- Затем нужно нажать на команду done.

На этом процесс создания пары ключей закончился и можно начинать пользоваться программой.

Теперь после установки программы необходимо обменяться со своими корреспондентами открытыми ключами. Для этого необходимо исполнить команду LAUNCH PGP KEYS, выделить свой ключ (файл со своим именем) в окошке, нажать на правую кнопку мышки и выбрать команду EXPORT.

Появится окошко, с помощью которого можно указать путь, где сохранить файл с названием <ваше имя.asc>

Этот файл необходимо выслать своему корреспонденту, в обмен на его открытый ключ.

Как только вы получите открытый ключ своего корреспондента, надо его запустить, нажав на него двойным щелчком мышки, выделить его в окошке и выполнить команду IMPORT.

Теперь можно пересылать друг другу зашифрованные сообщения, которые шифруются открытым ключом получателя сообщения.

Как послать зашифрованное сообщение

После того, как открытый (публичный) ключ вашего корреспондента установится на вашем компьютере, сообщение можно отправлять получателю следующим образом:

Составляем сообщение в почтовой программе Outlook Express.

После того, как сообщение готово к отсылке, нажимаем один раз либо на третий значок справа на панели Outlook Express с изображением желтого конверта и замка (при этом кнопка просто вдавливается и больше ничего не происходит), либо в меню tools нажимаем на encrypt using PGP и затем нажимаем на команду в меню file под названием send later.

Тогда сразу же появится окошко программы PGP под названием Recipient selection, в котором необходимо найти и выделить мышкой публичный ключ своего корреспондента (получателя сообщения, который обычно именуется именем получателя) и нажать на O'K.

Сразу же после этого программа автоматически зашифрует сообщение и поместит его в папку исходящих сообщений outbox

Теперь можно заходить в Интернет и отправлять все сообщения, готовые к отправке.

Расшифровка

Открываем полученное зашифрованное сообщение и нажимаем на второй справа значок на панели Outlook Express, либо на команду меню PGP decrypt message. Через несколько секунд сообщение будет расшифровано и появится в окошке.

Существует еще один способ использования PGP, который чуть-чуть сложнее, чем шифрование через Outlook Express. Этот способ можно применять в том случае, если не удается установить PGP вместе с программой Outlook Exress.

Создаем сообщение в Outlook Express, затем выделяем его через команды edit — select all и копируем в буфер Windows через команду copy.

После этого ставим мышку на значок PGP в панели задач, нажимаем на мышку и исполняем команду encrypt clipboard.

Появляется окно диалога с PGP под названием key selection dialog.

Необходимо выделить адрес (открытый ключ) корреспондента (ключ получателя сообщения)) в этом окне и щелкнуть по нему мышкой два раза, чтобы он появился внизу, потом нажимаем на O'K и программа зашифрует все содержимое clipboard.

После этого заходим в сообщение с текстом, который был ранее выделен, ставим мышку на поле сообщения, нажимаем на правую кнопку мышки и исполняем команду paste.

В результате зашифрованное содержимое clipboard заменяет предыдущее сообщение и на этом процесс шифровки закончился. Теперь можно отправлять сообщение обычным образом.

Расшифровывать полученные сообщения можно таким же образом: т.е. выделяем полученный зашифрованный текст, копируем его в буфер Windows clipboard, заходим мышкой в меню PGP через панель задач Windows и выбираем команду decrypt and verify clipboard.

Появляется окно программы PGP, в которое необходимо ввести пароль, вводим пароль в это окно,

нажимаем на O'K и перед нами предстает расшифрованное сообщение.

Естественно, перед тем, как это сделать, необходимо создать пару ключей, как было описано ранее.

Также кроме этого способа можно применить еще один способ шифрования (третий способ).

Можно создать текст в каком-либо редакторе, например блокноте, и сохранить его в виде файла. После этого в проводнике выделяем файл, нажимаем на правую кнопку мышки и видим, что в нижней части команды опций появилась еще одна команда под названием PGP, после чего, поставив мышку на PGP, мы увидим раскрывающееся меню, состоящее из 4 команд:

- encrypt
- sign
- encrypt and sign
- wipe

Нажимаем на первую команду и перед нами появляется диалог выбора открытого ключа корреспондента, выбираем ключ, нажимаем на O'K, вводим пароль и файл зашифрован.

После этого рекомендуется выполнить еще одну команду в меню PGP: wipe (стереть, уничтожить оригинальный файл). Иначе, какой смысл шифровать файл, если на диске компьютера остался первоначальный файл?

После этой операции у файла остается то же самое имя, но меняется тип расширения на <*.pgp>

Теперь этот файл можно прикрепить к сообщению и отправить вместе с ним.

В результате мы узнали, что существует три основных способа шифрования информации:

- Первый — самый удобный, напрямую в почтовой программе;
- Второй — через копирование текста в буфер обмена Windows;
- Третий — через шифрование всего файла, который затем прикрепляется к сообщению.

При работе с программой PGP появляется следующая проблема: при шифровании исходящих сообщений открытым ключом своего корреспондента, отправитель сообщений не может их потом прочитать, ввиду того, что исходящее сообщение шифруется с помощью закрытого ключа отправителя и открытого ключа его корреспондента, т.е. только получатель может прочитать такое сообщение. В результате получается, что отправитель не может впоследствии прочитать свои сообщения, отправленные им ранее.

В настройках PGP есть опция, позволяющая зашифровывать свои исходящие сообщения таким образом, чтобы их можно было потом прочитать (взять из архива и прочитать).

Для этого надо щелкнуть мышкой по символу PGP на панели задач, исполнить команду PGP preferences, зайти в General и поставить галочку напротив команды Always encrypt to default key

Кроме этого нужно зайти в PGP keys, выбрать мышкой свой ключ, зайти в меню keys и исполнить команду set as default key

Здесь же можно изменить свою парольную фразу: выделить мышкой свой ключ, нажать на правую кнопку мышки, исполнить команду key properties , change passphrase и поменять свою парольную фразу.

Парольную фразу рекомендуется менять, по крайней мере, раз в полгода, хотя если вы постарались создать надежную парольную фразу и исключили какую-либо возможность разгадки этой фразы кем бы то ни было, то этого можно и не делать.

Кроме того, там же (в key properties) можно увидеть fingerprint или своеобразные "отпечатки пальцев", состоящие из комбинации цифр и букв.

Эти отпечатки пальцев (идентификатор ключа) хороши тем, что можно предотвратить незаконное вторжение какими-либо людьми в вашу переписку. Т.е. кто-либо может перехватить ваш открытый ключ при отправке вашему корреспонденту или кому-либо еще и заменить своим открытым ключом. Когда ваш корреспондент получит этот ключ, то он будет думать, что это ваш ключ, когда в действительности это ключ третьего лица. Вы зашифровываете свое сообщение этим открытым ключом и в результате получается, что ваше сообщение не доходит до вашего корреспондента, а прочитывается другой третьей стороной, которая затем меняет это сообщение и отсылает вам под видом ответа от вашего корреспондента.

Для того чтобы исключить такие проблемы, владельцы открытых ключей созваниваются по телефону и

зачитывают друг другу отпечатки своих ключей. В таком случае достигается 100% надежность того, что информация не попала в чужие руки.

PGP диск

PGP диск – это удобное приложение, которое позволяет вам отвести некоторую часть вашего жесткого диска для хранения конфиденциальной информации. Это зарезервированное место используется для создания файла под именем <PGP disk>.

Хотя это всего лишь один файл, он действует подобно вашему жесткому диску в том отношении, что он выполняет функцию хранения ваших файлов и исполняемых программ. Вы можете его себе представить в виде флоппи дискеты или внешнего жесткого диска. Для того, чтобы использовать программы и файлы, находящиеся на нем, вы его устанавливаете <mount>, после чего его можно использовать также, как любой другой диск. Вы можете установить программы внутри этого диска либо копировать на него файлы. После того, как вы отключите <unmount> этот диск, он станет недоступным для третьих лиц и для того, чтобы открыть его, необходимо ввести парольную фразу, которая известна только вам. Но даже разблокированный диск защищен от несанкционированного доступа. Если ваш компьютер зависнет во время использования диска, то его содержание будет зашифровано.

Одним из наиболее важных преимуществ и удобств использования программы PGPdisk является тот факт, что теперь нет необходимости шифровать большое количество файлов, в которых находится конфиденциальная

информация. Теперь можно переместить все конфиденциальные файлы и даже программы на такой диск и таким образом избежать необходимости каждый раз расшифровывать какой-либо файл при его открытии.

Для того, чтобы установить новый PGP диск, необходимо выполнить следующие команды:

Пуск – Программы – PGP – PGPdisk

после чего появится окно программы со следующими командами:

- new – создать новый PGP диск
- mount – установить созданный диск путем ввода парольной фразы
- unmount – закрыть диск (зашифровать), который был ранее установлен
- prefs – опции настройки

Как создать новый PGP диск

1. Запустите программу PGPdisk

2. Исполните команду New, после чего на экране появится мастер создания PGP диска.

3. Нажмите на next

4. Появится окошко создания PGP диска, в котором необходимо указать путь, где новый диск под названием <New PGPdisk1> надо сохранить.

5. Нажмите на кнопку Save и файл под этим названием сохранится на диске, выбранном вами (по умолчанию на диске C).

6. Под надписью <PGPdisk Size field> введите цифру, обозначающую размер PGP диска и не забудьте выбрать килобайты или мегабайты там же.

7. Под надписью <PGPdisk Drive Letter Field> подтвердите букву, которую вы присвоите новому диску.

8. Нажмите на next

9. Введите парольную фразу, которую в дальнейшем необходимо будет вводить для установки нового диска. Введите парольную фразу два раза.

10. Нажмите на next

11. При необходимости подвигайте мышку или нажимайте на кнопки на клавиатуре для того, чтобы программа сгенерировала новый ключ

12. Нажмите на next. Столбик покажет вам инициализацию создания нового диска.

13. Еще раз нажмите на next, с тем, чтобы окончательно установить новый PGP диск.

14. Нажмите на Finish.

15. Введите название нового диска.

16. Нажмите на Start

17. Нажмите на OK (на диске еще нет данных). Компьютер скажет вам, когда закончится форматирование диска.

18. Нажмите на кнопку Close на окне форматирования. Теперь ваш новый диск появится на том диске, который вы ранее указали (по умолчанию диск C). Для того, чтобы открыть диск, надо дважды нажать на него мышкой.

Как установить PGP диск

Как только новый диск будет создан, программа PGP автоматически его установит с тем, чтобы вы могли начать его использовать. После того, как вы закончили работу с конфиденциальной информацией, необходимо отключить диск. После отключения диск его содержимое будет зашифровано в виде зашифрованного файла.

Для открытия PGP диска надо дважды щелкнуть по нему мышкой и дважды ввести парольную фразу в появившееся окно программы. Вы сможете убедиться в том, что PGP диск открылся, зайдя в мой компьютер и увидев, что рядом с диском C появился диск D. В том случае, если у вас уже есть диск D, то новый диск получит следующую букву E и т.д. Зайти на новый диск можно через мой компьютер или другую оболочку просмотра файлов.

Использование установленного PGP диска

На диске PGP можно создавать файлы, каталоги, перемещать файлы или каталоги, либо стирать, т.е. можно делать те же самые операции, что и на обычном диске.

Закрытие PGP диска

Закройте все программы и файлы, имеющиеся на диске PGP, т.к. невозможно закрыть диск, если файлы на этом диске до сих пор еще открыты. Теперь зайдите в мой компьютер выделите мышкой диск PGP, нажмите на правую кнопку мышки и выберите команду <unmount> в появившемся меню <PGP disk>.

Как только диск будет закрыт, то он исчезнет из моего компьютера и превратится в зашифрованный файл на диске C.

Еще один важный момент, на который необходимо обратить внимание, это настройки программы, которые позволяют автоматически закрыть диск в случае не обращения к диску в течение какого-либо периода времени. Для этого надо исполнить команду <prefs> в программе PGPdisk и в появившемся меню под названием <auto unmount> (автоматическое закрытие) выделить флажками все три команды:

- auto unmount after __ minutes of inactivity (автоматически закрыть после __ минут бездействия). Здесь также необходимо указать количество минут.
- auto unmount on computer sleep (автоматически закрыть при переходе компьютера в спящее состояние)
- prevent sleep if any PGPdisks could not be unmounted (не позволить компьютеру перейти в состояние спячки, если PGP диск не был закрыт)

Смена парольной фразы

1. Убедитесь в том, что PGP диск не установлен. Невозможно сменить парольную фразу в том случае, если диск установлен.

2. Выберите команду <Change Passphrase> из меню <File>

3. Выберите тот диск, парольную фразу для которго вы хотите изменить.

4. Введите старую парольную фразу. Нажмите на OK. Появится окошко для ввода новой парольной фразы.

5. Введите новую парольную фразу. Минимальная длина парольной фразы: 8 знаков

6. Нажмите на OK. Окошко новой парольной фразы <New passphrase> закроется.

Удаление парольной фразы

1. Убедитесь в том, что PGP диск не установлен.

2. Выберите команду <Remove passphrase> из меню <File>. Появится окошко, которое попросит вас ввести парольную фразу, которую необходимо отменить.

3. Введите пароль и нажмите на OK.

Система для шифрования с двумя ключами

Все желающие (PGP распространяется свободно) переписывают из любых источников саму систему (PGP) и ее исходные тексты (если есть необходимость, исходные тексты также распространяются свободно). Все функции системы выполняются с помощью одной-единственной программки — она делает все. Дальнейшие действия проследим на примере двух корреспондентов: меня (Пети) и Васи. Итак, Петя и Вася захотели использовать PGP и раздобыли его (хотя бы на нашем PGP support site — DIO-GEN).

Первое действие — генерация секретного ключа пользователя (RSA-key). Для этого система запрашивает фразу пароля, которая потом будет неоднократно использоваться при каждом использовании секретного

ключа. Кроме того, запрашивается собственно имя (идентификатор) пользователя, куда входит и его адрес (E-mail, FIDO), а основании этих данных и с использованием ряда случайных чисел (которые получаются путем измерения интервалов времени между нажатиями клавиш человеком) генерируется СЕКРЕТЫЙ ключ пользователя. Это просто бинарная последовательность. Этот ключ рекомендуется хранить достаточно тщательно, несмотря на то, что воспользоваться им сможет только тот, кто знает фразу пароля. Затем, на основании этого СЕКРЕТОГО ключа PGP генерирует ОБЩИЙ (публичный) ключ (тоже RSA). а этом предварительные действия закончены.

Следующий этап. Конечно, наши герои, Петя и Вася могли бы встретиться однажды, договориться о совсем секретном пароле и потом шифровать свои послания этим паролем. о порой это не совсем удобно, не всегда получится и не всегда секретно, и потом, если у Вас ТЫСЯЧИ адресатов:(... Так эта система делает все простым и гениальным. Петя, создав оба ключа, посылает общий ключ Васе (ну, и всем остальным, кому надо). То же самое делает и Вася. Итак, у Пети есть свой собственный секретный ключ, общий ключ Васи (и другие общие ключи, в принципе, сколько угодно). И все. PGP:).

Теперь Петя хочет написать Васе СОВСЕМ СЕКРЕТОЕ письмо. То есть, чтобы его смог прочитать ТОЛЬКО Вася. Петя берет общий ключ Васи и с помощью все той же PGP шифрует письмо (этим ключом). (На самом деле, шифруется не само письмо, а временный пароль, которым уже шифруется письмо, но это не принципиально). И посылает то, что получилось адресату (Васе). Петя может быть спокоен — теперь

зашифрованное общим ключом Васи письмо может прочитать ТОЛЬКО Вася, так как только у него есть секретный ключ, которому однозначно соответствует общий. Даже написавший письмо Петя не прочтет его.

Общий ключ генерируется на основе секретного, но обратный процесс невозможен в реальное время на реальной технике для ключей длиной >256 байт. Так же действует и Вася. Таким образом, первое, что позволяет делать эта система — это писать шифрованные письма адресатам, с которыми вы никогда не встречались или не имеете канала секретной связи.

Второе. Петя хочет "подписать" какой-то текст (программу, файл...), то есть поставить на него какую-то цепочку байт, по которой получивший сможет определить: ага, этот файл был однозначно завизирован Петей, и никто эту подпись не подделывал (что-то отдаленно напоминает ARJ security envelope). Петя берет свой файл, свой секретный ключ и с помощью PGP генерирует ЭЛЕКТРОУЮ ПОДПИСЬ. Тоже просто последовательность байт. о последовательность абсолютно уникальная, так как для ее построения используется алгоритм "message digest", это отдаленный аналог CRC того файла, который подписывается, но у двух файлов может быть одинаковый CRC, а вот MD — не может. Теперь Петя может послать эту подпись вместе с файлом Васе (или отдельно). Вася (человек недоверчивый), берет файл, берет подпись (если она отдельно от файла), берет общий ключ Пети, запускает PGP, а она ему: "да, данный файл действительно был подписан таким-сяким тогда-то". Чему можно доверять почти на 100%.

Это два основных принципа использования шифрования с общим ключом. Естественно, оба они могут быть использованы вместе: то есть, вы шифруете свое письмо адресату с полмощью ЕГО общего ключа, а затем подписываете его своей электронной подписью с помощью СВОЕГО секретного ключа.

Теперь я расскажу, как это дело организуется на практике. Для более подробного описания ваших действий по предохранению информации, ключей, о предотвращении возмонжости их подделки — читайте первую часть документации по PGP (зря я ее переводил?).

Key-server. Это станция из числа многих, раскиданных по всему свету. Для того, чтобы Вам самому не рассылать свой общий ключ "всем-всем-всем" (что, как Вы понимаете, довольно трудоемко), эту часть работы они берут на себя. а сервере Вы может зарегистрировать свой общий ключ — он будет рассылаться всем, кто этого пожелает, а также получить все необходимые общие ключи, чтобы писать письма. В PGP введен институт"удостоверения" общих ключей, это составная часть мероприятий по защите ключей from tampering. То есть, если я уверен, что вот этот ключ мне передал (однажды) Миша Браво (переслал по почте — именно он, либо привез лично), то я могу удостоверить его своей как бы "электронной подписью" (удостоверением), и при проверке сообщений, подписанных Мишей Браво будут упоминаться все удостоверившие (таким образом, при удостоверении ключа я (ты, он...) беру на себя ответственность за его достоверность. Степень достоверности общего ключа определяется PGP по количеству и степени достоверности "удостоверяющих подписей" (причем все делается автоматически).

Для удобства работы PGP групирует ключи в keyrings (каталоги), таким образом, Вы можете иметь у себя на машине только два файла (кроме системных PGP): каталог секретных ключей и каталог общих ключей. Причем последний можно получить на нашем сервере. И вообще, программа очень неплохо эргономически построена, несмотря на интерфейс командной строки. Все вещи делаются за один присест. А для обработки пришедшего файла (что бы там ни было: подпись, ключ, письмо зашифрованное) достаточно просто дать команду "pgp file-name" — и она сама в нем разберется и разложит все новые ключи и подписи по местам.

Еще одно — весьма важное — свойство PGP: она очень неплохо делает ASCII armour. Вам сие действие должно быть знакомо: UUкодирование файлов в эху, да-да. Вкратце: это Вы берете любой файл (текстовый, двоичный) и кодируете его так, чтобы в результате получились только символы первой половины таблички ASCII. Это необходимо при пересылке файлов в эхах и нетмэйлом (ну, и E-mailom). Так вот, PGP это делает с одновременной упаковкой (по алгоритму ZIP, лицензия была предоставлена разработчикам), так что файлы получаются меньше, чем у UUENCODE/UUDECODE, к тому же их не надо предварительно архивировать. Таким образом можно пересылать и тексты и программы, естественно. PGP может файл зашифровать, а может и не зашифровывать, подпись может быть добавлена в сообщение, а может быть послана отдельно. у, и, разумеется, зашифрованные файлы и подпись могут быть созданы в виде двоичных файлов.

Приложения

Краткий путеводитель по миру PGP

Начиная с 1991 года аббревиатура PGP стала своеобразным символом. Во-первых, она обозначает определенное семейство продуктов (в основном, разработанных Филипом Зиммерманном, или под его руководством). Во-вторых, она ассоциируется с бескомпромиссным подходом к стойкости криптографии для массового использования, применением определенных алгоритмов, протоколов и форматов, прошедших долгую практическую проверку в сетях публичного пользования. В-третьих, она указывает на де-факто стандарт шифрования и цифровой подписи для электронной почты, который, вероятно, вскоре станет официальным стандартом Интернета.

Продукты

За семь лет триумфального шествия PGP были созданы десятки программ, в той или иной степени использующие компоненты технологии PGP, или согласованных с ней. Описать или даже перечислить их в одной статье не представляется возможным, поэтому здесь представлены лишь важнейшие продукты.

PGP 2.x.x и ее расширения

PGP 2 — это "классическая" PGP. С помощью команд строчного интерфейса ее пользователь может

выполнять все базовые криптографические функции, а именно:

- генерацию пары из закрытого/открытого ключа;
- шифрование файла с помощью открытого ключа любого пользователя PGP (в том числе своего);
- расшифровку файла с помощью своего закрытого ключа;
- наложение цифровой подписи с помощью своего закрытого ключа на файл (аутентификация файла) или на открытый ключ другого пользователя (сертификация ключа);
- проверку (верификацию) своей подписи или подписи другого пользователя с помощью его открытого ключа.

Награды PGP

1997 "Crossroads 98 A-List Award" (единственный промышленный приз, вручаемый компанией Open Systems Advisors, Inc. на основании опроса пользователей, использующих продукт в критическом контексте).

1996 "Приз за качество в информационных технологиях" от PC Week

1996 "Лучший продукт для защиты" от Network Computing

1996 Вторая премия за "Лучший Интернет-продукт" от Network Computing

1995 "Один из 10 самых важных продуктов 1994 г." от InformationWeek

1995 Крайслеровская премия "За инновацию в разработках".

Справка: Начиная с 1994 г., только с сервера MIT подгружается по 500–1000 копий PGP в день. Общее количество установленных копий всех версий оценивается в 2–5 млн.

Награды Фила Зиммермана

1996 Премия Норберта Винера от CPSR ("Компьютерные профессионалы за социальную ответственность").

1995 Приз "За личные достижения" от Internet World

1995 Приз "Pioneer Award" от Фонда электронных рубежей (EFF).

Простейшие функции управления "связками ключей"

Кроме того, PGP 2 реализует простейшие функции управления "связками ключей", включая добавление и удаление ключа со связки, возможность гибкой настройки параметров, а также встроенные функции сжатия исходного открытого текста по алгоритму ZIP и кодирования шифровки в формате RADIX-64, позволяющем пересылать сообщения по 7-битным каналам, не прибегая к дополнительному армированию (uuencode и пр.)

В качестве криптоалгоритма с открытым ключом (как для шифрования, так и для наложения подписи) в этой версии был использован RSA, в качестве алгоритма хеширования файла для наложения — MD5, а в качестве симметричного криптоалгоритма — IDEA. Популярность PGP имела своей обратной стороной то, что фирмы,

претендующие на обладание патентами на эти алгоритмы (RSA Data Security, Inc., США, — на первые два, и Ascom-Systec, Швейцария, — на третий) засыпали Зиммерманна судебными исками. История многочисленных обращений в суд, соглашений, отзывов и повторной подачи исков могла бы составить тему отдельного исследования.

Конечно, на популярность PGP очень сильно повлияли косвенные обстоятельства, такие, как попытки преследования Зиммерманна властями (косвенная аттестация качества программы спецслужбами), тяжбы с RSA и Ascom-Systec (привлекшие внимание всех тех, кто обеспокоен распространением "прав интеллектуальной собственности" на алгоритмы) и, наконец, бесплатное ее распространение.

Однако этих обстоятельств было бы недостаточно, чтобы сделать PGP 2 де-факто криптографическим стандартом для электронной почты, удерживать внимание пользователей в течение почти семи лет (своеобразный рекорд для всей компьютерной отрасли) и оставить позади всех конкурентов.

Суть заключается не в этом, а, как ни странно, в том, что Зиммерманн написал хорошую программу. С самого начала PGP соответствовала всем требованиям, предъявляемым гражданскими криптологами к криптографическому программному обеспечению, а именно:

- использование проверенных алгоритмов, выдержавших попытки взлома в течение достаточного времени;

- длина ключей, достаточная, чтобы исключить снижение безопасности в результате увеличения вычислительных ресурсов потенциальных оппонентов (удваивающихся каждый год, согласно закону Мура) в течение длительного периода времени (512-битные и длиннее ключи PGP 2.6 считаются относительно безопасными и сегодня);
- локальная генерация и локальный менеджмент ключей, исключающие их попадание в чужие руки;
- гибкая схема удостоверения действительности ключей, допускающая как распределенное управление доверием ("сеть доверия"), так и централизованную архитектуру сертификации;
- и, наконец, открытость и доступность для проверки и критики не только алгоритмических решений и форматов файлов, но и исходного текста самой программы.

PGP очень хорошо документирована. В отличие от большинства программных продуктов, документированы не только достоинства и сильные стороны программы, но также и уязвимые места; соответствующая глава сохранилась и в документации на коммерческие версии.

Все это очень высоко подняло планку качества для криптографических продуктов, и хотя за последние годы были предложены десятки альтернативных программ, ни одной из них не удалось приблизиться по количеству используемых копий к PGP 2. И сегодня эта версия конкурирует лишь с другими версиями PGP и ее клонами.

Позднее была построена версия PGP 2.6.3i ("международный релиз"), не использующая библиотеку

RSAREF (это бесплатно распространяемая RSA, Inc., то есть, американского производства, библиотека криптографических функций, примененная в версиях PGP начиная с 2.5).

Ресурсы и документация PGP 2.6 переведены на два десятка языков. Русскую локализацию PGP выполнил Андрей Чернов, а первый том "Руководства пользователя" переведен на русский Петром Сучковым.

Изначально ориентированная на массовых пользователей MS-DOS и Unix, PGP 2.6 была затем перенесена и на другие платформы, включая OS/2, MacOS, BeOS и даже Amiga и Atari. Доступ ко всем этим вариантам программы, к исходным текстам, а также к русским ресурсам можно получить со страниц "Русского Альбома PGP".

Недостатками этой версии принято считать неразвитость механизма управления связками ключей (они хранятся в виде простых последовательностей в файлах, и работа со связками, состоящими из сотен и тысяч ключей становится невыносимо медленной), ограниченный набор использованных в ней криптографических алгоритмов и ограничение на длину ключа. С распространением графических пользовательских интерфейсов (ГПИ) ОС и оболочек, а также почтовых программ, использующих ГПИ, консольный интерфейс PGP 2 стал преградой на пути приобщения к криптографии массового пользователя — тех, на кого, собственно, и была рассчитана программа.

Последний недостаток отчасти компенсировался появлением независимо разработанных программ —

графических оболочек для PGP 2.6 и модулей сопряжения этой программы с популярными почтовыми пакетами.

PGP 2.7–4.5

Зиммерманн планировал приступить к разработке PGP 3 в 1993 году. Предполагалось, что третья версия программы предоставит пользователю достаточно широкий выбор криптографических алгоритмов, в то же время сохраняя обратную совместимость со второй версией. Третья версия должна была поддерживать связки ключей как базы данных и предоставлять пользователю графический интерфейс.

Однако PGP 3 осталась бумажным проектом. В "Обращении к пользователям PGP" Зиммерманн пишет, что это произошло "по разным причинам, и не в последнюю очередь из-за трехлетнего уголовного расследования, предпринятого против меня американским правительством. Последнее обстоятельство действительно замедлило всю работу. Оно отняло почти всех моих добровольных помощников, усилия которых были столь полезны при работе над PGP 2.0 и последующими версиями... Это привело к формированию работавшей, в основном, бесплатно группы правовой защиты, созданию Фонда правовой защиты и трем годам почти ежедневных интервью. Пресса была в массе своей настроена против такого преследования, и вопрос о политике в области криптографии возбудил гнев всей компьютерной промышленности".

Летом 1993 года Зиммерманн предоставил лицензию на коммерческую разработку технологии PGP компании ViaCrypt, принадлежащей Lemcom, известному поставщику контроллеров для мэйнфреймов IBM.

ViaCrypt начала поставки коммерческой версии PGP для персонального использования в ноябре 1993 года. В августе следующего года у PGP 4 появился графический интерфейс, а в апреле 1996 года была выпущена версия PGPmail 4.5, ориентированная на корпоративное использование.

Несмотря на все усовершенствования, PGP 4.x остались маргинальными продуктами, но квартальный объем их продаж все же превысил один млн. долларов.

В июле 1996 года молодая и агрессивная компания Pretty Good Privacy, Inc., основанная за четыре месяца до этого Зиммерманном, Дэном Линчем (Dan Lynch) и Джонатаном Сейболдом (Jonathan Seybold), приобрела Lemcom вместе с дочерней ViaCrypt и получила, таким образом, исключительные права на коммерческое использование торговой марки, принадлежащей первоначально одному из ее основателей.

Pretty Good Privacy, Inc.

Инкорпорирована в марте 1996 года для разработки и поставки коммерческих версий программного обеспечения PGP.

Президент

Фил Данкелбергер (Phil Dunkelberger) окончил Вестмондский колледж в Санта-Барбаре (Калифорния) с особым отличием по специальности политология. Руководил продажами в Apple Computers и Symantec. Затем руководил операциями Center View Software.

Совет директоров

Джонатан Сейболд (Jonathan Seybold), председатель совета директоров, окончил с отличием Оберлин-колледж по специальности экономика, изучение которой продолжил в Йеле. Один из основателей отрасли электронных публикаций. Остается президентом стратегической консультационной фирмы Deep Harbor Group.

Филип Зиммерманн — директор, главный технолог.

Дэниэл Линч (Daniel Lynch), директор, закончил университет Лойола-Мэримаунт и получил степень магистра в университете Калифорнии (Лос-Анджелес) по математике. Один из пионеров глобального сетевого строительства, возглавлял DARPA при ее переходе на TCP/IP. Основатель Cybercash, Inc. и Interop Co.

Марк Дж. Горлин (Marc J. Gorlin), директор, окончил с отличием университет Джорджии по специальности журналистика. В PGP занят формированием стратегических партнерств.

Исполнительный совет

Боб Кон (Bob Kohn), вице-президент по деловому развитию, закончил юридическую школу Лойолы и университет Калифорнии (Нортбридж) по специальностям деловое администрирование и экономика. Член юридической коллегии Калифорнии и профессор права в Монтери-колледже. Работал старшим вице-президентом в Borland International, Inc., а до этого — в Candle Corp. и Ashton-Tate.

Оливия Диллан (Olivia Dillan), вице-президент по разработке продуктов, окончила с отличием

Хантер-колледж по специальности информатика. Занимала ключевые должности в Internet Profiles Corp., Oracle, ASK/Ingres и RadioMail.

Мириам К. Фрейзер (Miriam K. Frazer), вице-президент по финансам и операциям, окончила университет Санта-Клары по специальности деловое администрирование. Работала на высших должностях в AT&T Paradyne Corp., Telematics International, Inc. и Software Publishing Corp.

Поглощения

- июль 1996 года — Lemcom и ViaCrypt (сумма сделки не объявлена);
- декабрь 1996 года — PrivNet (сумма сделки не объявлена).

Стратегические партнеры

- FTP Software, Inc. (поставщик TCP/IP решений для корпоративных сетей);
- Schlumberger Electronic Transactions (ведущий поставщик смарт-карт и основанных на них систем);
- First Virtual Holdings, Inc. (пионер электронного маркетинга и торговли);
- QUALCOMM, Inc. (поставщик самого популярного пакета электронной почты Eudora);
- Allegro Group, Inc. (эта компания поставляет шлюз защищенной коммуникации для Novell GroupWise, а количество пользователей GroupWise приближается к 10 млн. человек).

- Tech Data Corp. (один из крупнейших поставщиков микрокомпьютерного оборудования, оперирует на территории США, Канады, Европы, Центральной и Южной Америки).
- CompUSA, Inc. (один из крупнейших розничных торговцев компьютерами в США, владеющий 139 компьютерными супермаркетами в 62 крупных городах, а также выполняющий сборку заказных конфигураций компьютеров марки CompUSA PC).

PGP for Personal Privacy 5.0

PGP, Inc. начала бизнес с захвата стратегических позиций в области коммерческой криптографии. 15 декабря 1996 года ею приобретена PrivNet — компания, основанная в 1995 году и специализирующаяся на защите приватности онлайновых сеансов связи. PrivNet разработала, в частности, многообещающую технологию Internet Fast Forward (IFF), на основе которой строятся системы фильтрации рекламного трафика и нейтрализуются зловредные cookies, но не смогла довести ее до коммерческого продукта. Возможно, более важным, чем все разработки PrivNet, является для PGP, Inc. тщательно подобранный коллектив разработчиков, в полном составе перешедший в PGP.

Многие из них имеют опыт разработки программ на основе исходного кода PGP 2.

В течение 1996–97 годов компания заключает ряд перспективных соглашений с корпорациями "цифрового поколения", ориентированными на работу в области электронных коммуникаций, коммерции и финансов.

Одновременно с простраиванием вертикальных рынков, PGP, Inc. продолжила разработку продуктов для массового пользователя. Новая серия продуктов PGP 5 обладает многими качествами, которыми предполагалось снабдить оставшуюся на бумаге PGP 3.

Эта серия открывается выпущенной в июне "PGP для Персональной приватности 5.0". PGP 5.0 реализована под Unix, 32-битные Windows и MacOS.

PGP 5.0 предоставляет пользователю выбор между использованием "старых" (совместимых с PGP 2) ключей RSA и "новых" ключей DHE/DSS, причем для технологии DHE поддерживаются ключи длиной до 4 кбит.

PGP 5.0 для систем с ГПИ "прозрачно" интегрируется в ОС и почтовый пакет. (Далее приводятся примеры из версии для Windows). Программа оперирует с тремя типами объектов:

- с файлами (доступ к функциям шифрования/расшифровки и наложения/проверки подписи открывается из менеджера файлов ОС, в Windows это Проводник);
- с текстовым содержимым буфера обмена (доступ к криптографическим функциям открывается из меню, выскакивающего при нажатии на значок программы PGPtray в Области системных индикаторов Панели задач);
- с почтовыми сообщениями.

Для реализации последней возможности почтовая программа должна быть снабжена особой вставкой (plug-in). Если для используемого пакета такая вставка еще не разработана, криптографические операции над

сообщением можно производить через буфер обмена, хотя это и менее "прозрачно". Поскольку в поставку не входит консольный интерфейс, вставки и интерфейсы, предназначенные для работы с PGP 2, с пятой версией работать не будут. В поставку PGP 5.0 для Windows входят вставки для MS Exchange/Outlook и Eudora. Уже разработаны вставки для Mail'97 и Pegasus Mail для Windows. После выхода PGPsdk (см. ниже), вероятно, они будут доступны для большинства используемых почтовых программ.

Отдельная программа PGPkeys, также вызываемая из меню PGPtray, предназначена для управления ключами. Пары из закрытого и открытого ключа, принадлежащие пользователю, а также открытые ключи других пользователей PGP, находящиеся на связках, теперь наглядно представлены в виде списка объектов в окне, с которыми можно выполнять те же действия, что и с другими объектами интерфейса Windows (например, перетаскивать мышкой в другое окно, или вызывать контекстное меню нажатием правой кнопки).

PGP 5.0 для Windows поставляется в виде одного exe-файла объемом 3,6 Мбайт, для полной (со вставками для MS Exchange/Outlook и Eudora) установки требуется 15 Мбайт дискового пространства, для компактной — всего 6 Мбайт, в которые входит и pdf-файл со 130-страничным "Руководством пользователя". Требования: Windows 95 или NT, 8 Мбайт оперативной памяти, 7–15 Мбайт свободного дискового пространства.

Выпущены коммерческий, пробный (trial) и бесплатный (freeware) варианты PGP 5.0. Бесплатная поставка не включает функции генерации RSA-ключей

(хотя с существующими RSA-ключами работает). Кроме того, исходные тексты PGP 5.0 изданы в виде книги (точнее, многотомного издания, состоящего из более чем 6000 страниц), и группа европейцев во главе с хозяином "Международных страниц PGP" Стале Шумахером (Stele Schumacher), студентом университета Осло, взялась выпустить международный релиз 5.0i (под это UNINETT (норвежская академическая сеть) выделила особый грант, а норвежская национальная библиотека предоставила сканеры).

Одновременно с подготовкой международного релиза начата локализация продукта и перевод документации на ряд европейских языков.

Первым языком, на который переведено "Руководство пользователя PGP", стал русский, черновик перевода доступен на страницах "Русского Альбома PGP", а окончательный "официальный" документ будет опубликован после выхода международного релиза PGP 5.0 для Windows, в соответствии с системой лицензионных соглашений, предоставленных PGP, Inc. в электронной форме в Норвегии. Анонс о публикации будет размещен также на страницах "Русского Альбома PGP".

Русская локализация ресурсов PGP 5.0 для Windows находится в альфа-стадии, скриншотами с ресурсов альфа-версии PGP 5.0iRu проиллюстрирован этот раздел статьи.

Важное примечание для русских пользователей PGP 5.0

Предполагается, что пользователи за пределами США и Канады начнут переход на пятую версию после выхода PGP 5.0i для соответствующей платформы. Однако, поскольку PGP 5.0 freeware получила широкое

распространение и спонтанная миграция пользователей уже началась, следует иметь в виду следующие четыре момента.

Все варианты PGP 5.0, включая изданный в печатном виде текст, содержат ошибку в таблице конверсии символов, из-за которой символ ASCII #213 ("у" русское в стандартной кодировке KOI8-R и "Х" русское в кодировке Windows CP-1251) воспринимается программой как символ ASCII #123 ("{"). Это может вести к искажению подписываемого или шифруемого текста, а также к невозможности проверки "открытой" подписи, наложенной с помощью более ранних версий PGP.

PGP 5.0 не содержит таблиц конверсии, кроме "нулевой таблицы". В то же время, порядок применения криптопреобразований и конверсии сегодня, к сожалению, не определен никаким документом. Поэтому возможны конфликтные ситуации, связанные как с разным порядком работы модулей почтовых пакетов отправителя и получателя, так и с невозможностью почтового пакета правильно распознать кодировку сообщения. Более того, расшифровка или верификация текстового вывода PGP 2.6.xi при кодовой таблице, отличной от нулевой, невозможна. При кодовой таблице ASCII (задаваемой параметром charset в файле конфигурации PGP 2.6.xi) проблем не возникает.

При необходимости сгенерировать RSA-ключ (необходимый для коммуникации с пользователями ранних версий), это можно сделать с помощью PGP 2.3.6i, а затем "зацепить" сгенерированную пару на связку PGP 5.0 freeware.

PGP Business Security Suite

Следующим в новой серии PGP стал "PGP 5.5 для безопасности бизнеса", а точнее, "Комплект для безопасности бизнеса", включающий в себя, кроме PGP 5.5, Сервер сертификатов (PGP Certificate Server) 1.0 и Агент управления политикой (PGP Policy Management Agent) 1.0. Комплект начал поставляться в октябре.

Особые свойства первого и последнего компонента "Комплекта", позволяющие корпорациям вводить обязательное депонирование содержимого электронной переписки своих сотрудников, и стали предметом ожесточенных споров криптологического сообщества в последние месяцы, о которых рассказано в статье "Новое лицо PGP".

На фоне этих споров практически незамеченным осталось то, что эти продукты, в отличие от PGP 5.0, написаны с использованием универсального криптографического интерфейса разработчика (cAPI) PGPsdk, который через месяц был представлен в качестве коммерческого продукта (см. ниже).

Кроме этого, версия 5.5 включает:

- утилиту PGPwipe, предназначенную для "физического" затирания файлов на диске.
- возможность объединения открытых ключей в группы, позволяющие удобно шифровать сообщения, рассылаемые циркулярно;
- дополнительные возможности взаимодействия с серверами открытых ключей, включая обращение к нескольким серверам и динамическую синхронизацию баз данных;

- возможность поиска ключей на сервере по нескольким параметрам;
- утилиту PGPtools, используемую для вызова компонентов PGP;
- возрожденную возможность многоуровневого посредничества при заверении ключей;
- графический интерфейс под X-Window в Unix-среде.

ПозжеPGP, Inc. пообещала представить бесплатную версию PGP 5.5. К моменту выхода журнала она должна стать доступной американским пользователям.

PGPdisk для Macintosh

Для пользователей Macintosh PGP, Inc. выпустила в этом году программу PGPdisk, шифрующую содержимое целого диска. Эта программа существует только в коммерческом варианте и легально доступна только в США и Канаде.

Сторонние разработки

Из многочисленных сторонних разработок хочется отметить две:

Private Idaho — почтовая программа, автоматизирующая не только базовые операции PGP, но и обращение к почтовым серверам открытых ключей, и даже программирование прохождения сообщением "лабиринта" анонимных римэйлеров.

Private Idaho написана Джоэлем Мак-Намарой (Joel McNamara), она доступна с http://www.eskimo.com/~joelm/pi.html.

"Партизанская" PGP (Guerilla PGP, PGP x.xG) — это модификация PGP 2, снимающая ограничение на длину RSA-ключа в 1024 кбит и содержащая массу других усовершенствований. Программа разработана на основе исходного кода PGP 2.6 Ноэлем Беллом (Noel Bell) и доступна с ftp://users.aol.com/ejnbell/

Ряд других сторонних разработок доступен со страниц "Конференции пользователей PGP".

Онлайн серверы открытых ключей

Для любой криптосистемы основной проблемой является обмен открытыми ключами. PGP предусматривает гибкую систему сертификации открытых ключей ("сеть доверия"), исключающую массовую их компрометацию.

Сообщество любой популярной криптосистемы нуждается также в некотором механизме для быстрого поиска и передачи ключей, их сертификатов и сертификатов отзыва ключей. Начиная с версии 5.0, в программу интегрирована связь с онлайновыми серверами открытых ключей. Такие серверы поддерживаются рядом организаций, включая PGP, Inc. и МТИ. Поскольку передача ключей обеспечивается посредством не только особого протокола hkp, но и http, к серверу открытых ключей можно обратиться и посредством обычного браузера WWW.

Существующие серверы открытых ключей завязаны в сеть, поэтому ключ или сертификат, подгруженный на один из них, вскоре становится известен и всем остальным.

Почтовые серверы открытых ключей и PGP.net

До разворачивания инфраструктуры онлайновых серверов ключей уже существовала сеть серверов, собирающих и распространяющих открытые ключи PGP по электронной почте. Эта сеть базируется в домене pgp.net. Сообщество пользователей PGP считает целесообразным продолжать ее поддержку в интересах пользователей, не имеющих полного доступа в Интернет (включая членов FIDO и FIDO-образных сетей). В настоящее время заканчивается доработка программного обеспечения почтовых серверов, которая позволит использовать их и для обмена новыми (DH/DSS) ключами PGP 5.x.

Инструкции по использованию серверов pgp.net можно найти на страницах "Русского Альбома PGP", или получить, отправив сообщение, содержащее "HELP" в поле Subject, по адресу pgp-public-keys@keys.pgp.net.

"Лабиринт" анонимных римэйлеров

Римэйлер — это программа, принимающая почтовое сообщение, удаляющая адрес отправителя и отправляющая ее дальше по указанному адресу. Этой цели могут служить и криптографически не защищенные римэйлеры, однако в этом случае адрес отправителя может стать известен администратору римэйлера или тому, кто сможет перехватить входящий пул сообщений римэйлера.

Поэтому в критических случаях рекомендуется использовать криптографически защищенные римэйлеры. Сообщение римэйлеру шифруется его открытым ключом, затем, вместе с адресом этого римэйлера, шифруется ключом другого римэйлера и так далее. Анонимность отправленного таким образом через "лабиринт" сообщения

гарантирована настолько, насколько вероятно, чтобы администраторы всех римэйлеров, входящих в "лабиринт", сговорились, чтобы определить вашу идентичность.

Список адресов, инструкции и другую информацию о работе с римэйлерами PGP можно найти на странице http://www.cs.berkeley.edu/~raph/remailer-list.html.

Соглашения и стандарты

В настоящий момент к PGP относятся два зарегистрированных документа Интернет: RFC 1991 ("Формат обмена сообщениями PGP") и RFC 2015 ("Обеспечение безопасности MIME с помощью PGP", определяющий формат PGP/MIME).

Введение новых алгоритмов и возможностей в PGP 5.x потребовало пересмотра этих документов. В сентябре этого года группа, работающая над их пересмотром, получила статус официальной рабочей группы IETF, и результат ее работы (стандарт open PGP) будет представлен к утверждению в качестве стандарта Интернет для защиты электронной почты, то есть, получит более высокий статус.

До сих пор попытки создать единый стандарт защиты электронной почты Интернет оканчивались неудачей. Наиболее серьезный документ до сих пор был представлен рабочей группой по S/MIME, он базируется на технологии RSA, и ранние версии S/MIME поддерживаются многими производителями ПО. Однако в сентябре консорциум электронной почты IETF отверг версию 2 S/MIME в качестве предложенного стандарта.

Тому было две причины. Во-первых, возможность реализации этого стандарта сильно ограничена тем, что

RSA, Inc. удерживает патент, покрывающий технологию RSA, который действителен до сентября 2000 года. Во-вторых, вторая версия S/MIME допускает использование длины ключей, не являющейся, по современным представлениям, достаточной для стойкости шифрования. Это связывают с желанием RSA, Inc. продолжать поставки "кастрированных" в соответствии с правилами, установленными американским правительствам, RSA-модулей для "экспортных" версий программного обеспечения таких производителей, как Microsoft или Netscape, которые считались бы соответствующими стандарту.

IETF же, как международная организация, стремится к тому, чтобы правила отдельных национальных правительств не влияли на ее политику (так называемая "Денверская доктрина"). И, конечно же, принятие стандарта, основанного на патентованной технологии (если владелец патента не предоставляет права на бесплатное его использование) является недопустимым ("Мюнхенская доктрина" — эти два неформальных правила называются так по месту соответствующих пленарных сессий IESG).

Позже рабочая группа по подготовке S/MIME была воссоздана, и к настоящему времени подготовила проект S/MIME 3.

Принятие в качестве стандарта open PGP откроет производителям широкую возможность маневра. Заложенный в нем богатый выбор алгоритмов позволит им создавать стандартное ПО с использованием как патентованных, так и находящихся в общем пользовании алгоритмов. К последним относятся, прежде всего,

базовые алгоритмы Диффи-Хеллмана ("Криптографический аппарат и метод") и Хеллмана-Меркле ("Аппарат криптографии с открытым ключом"), срок действия патентов на которые, выданных Стэнфордскому университету и лицензированных в разное время компаниям Cylinc, RSA, Inc. и партнерству PKP, истек в марте и августе этого года, соответственно.

Следующим логическим шагом станет, по-видимому, стабилизация способов использования PGP с другими протоколами, как общими (http, ftp и т.п.), так и специализированными (такими, как протоколы, используемые при цифровых денежных транзакциях, цифровом голосовании и т. п.). PGP/MIME уже принят за основу ряда проектов, разрабатываемых рабочей группой по электронному обмену данными (EDI) EITF.

Лицензирование торговой марки и международное партнерство

Как и другие американские компании, PGP, Inc. все еще лишена возможности поставлять стойкие криптографические решения и услуги, связанные с их разработкой, за пределы США и Канады1. Компания не пошла по пути "кастрации" криптомодулей для экспорта, подобно RSA, Inc. 2, и это не только проявление солидарности с зарубежными пользователями, но, возможно, и очень сильный маркетинговый ход: ведь PGP, таким образом, остается символом бескомпромиссных с точки зрения стойкости криптографических решений.

Каким же образом возможны разработка и применение этой технологии за рубежом?

Во-первых, PGP, Inc. продолжает публиковать исходные тексты программного обеспечения в виде книг, экспорт которых правительство запретить не может. Во-вторых, все публикуемые бесплатные продукты PGP снабжены лицензией, позволяющей в некоммерческих целях компилировать это обеспечение, а скомпилированная за рубежом программа, с юридической точки зрения, перестает быть "американским товаром". В-третьих, PGP, Inc. предоставила право на использование названий PGP/MIME и open PGP консорциуму электронной почты Internet.

И, в-четвертых, зарубежным производителям PGP предлагает подписать соглашение об использовании торговой марки и договор о партнерстве. Договор освобождает подписавшую его сторону от претензийPGP, Inc. по поводу использования опубликованных исходных текстов в коммерческих продуктах.

В октябре PGP Inc. распространила этот порядок и на отдельных пользователей. Лицо или фирма, желающие регулярно использовать бесплатный международный релиз PGP 5.0 в деловых целях, могут теперь приобрести у PGP, Inc. лицензию на его коммерческое использование.

К настоящему времени объявлено о подписании договора о партнерстве с германской компанией Gluck & Kanja GmBH, которая начала поставки коммерческих PGP-совместимых продуктов в Европе.

Стандартизация в Internet

Разработкой стандартов Internet занимается Инженерная команда Internet (Internet Engineering Task Force, IETF). IETF — это широкое сообщество разработчиков и продвинутых пользователей технологий

Internet, желающих участвовать в развитии инфраструктуры сети. Членство в IETF неформальное, к ней фактически принадлежит каждый, кто участвует в одной из рабочих групп — на ее "физических" встречах во время сессий IETF или, чаще, посредством соответствующей почтовой конференции (списка рассылки).

Когда возникает необходимость формализовать какую либо процедуру, формат данных, протокол и т. п., в одной из областей IETF (IETF Areas) создается тематическая рабочая группа IETF (IETF Working Group). Результатом работы такой группы становится так называемая заявка на обсуждение (Request for Comments, RFC). Регистрации RFC (а каждая из них имеет свой уникальный номер) обычно предшествуют одна или несколько публикаций проектов (Internet Drafts).

Следует заметить, что не все RFC определяют стандарты Internet: число первых перевалило за две тысячи, в то время как стандартов на сегодняшний день зарегистрировано всего 53.

Целью регистрации RFC, кроме продвижения к стандарту, может быть, например, описание принятой практики выполнения каких-либо задач (Best Current Practice, BCP) или обобщение информации по какой либо теме (For Your Information, FYI). Так, процедура стандартизации Internet подробно описана в документе BCP 9 (его третья ревизия зарегистрирована как RFC 2026).

Каждая спецификация, претендующая на то, чтобы стать стандартом Internet, проходит три "уровня зрелости": (1) предложение стандарта (Proposed Standard), (2) проект

стандарта (Draft Standard) и собственно (3) стандарт (Internet Standard).

Продвижение стандарта начинается по инициативе рабочей группы, передающей заявку директору соответствующей области и в секретариат IETF, откуда она попадает в Инженерно-организационную группу Internet (Internet Engineering Steering Group, IESG).

IESG — более узкий круг профессионалов, чем IEFT. В ее задачу входит поддержание и развитие документального сопровождения инфраструктуры Internet.

Статус предложения стандарта спецификация получает решением IESG. Для этого требуется, чтобы необходимость в стандартизации была четко зафиксирована сообществом разработчиков и пользователей, чтобы спецификация прошла достаточное количество обсуждений и решала имеющиеся проблемы. Наличия реализации предложения стандарта не требуется.

Для того, чтобы спецификация поднялась на уровень проекта стандарта (этот статус присваивается также решением IESG), требуется наличие как минимум двух независимых реализаций предложения стандарта, являющихся совместимыми.

Полностью "созревший" стандарт, который регулярно используется независимыми разработчиками и в отношении применения которого накоплен достаточный положительный опыт, может быть "повышен" в статусе до стандарта Internet и получить порядковый номер в списке STD.

Стандарт, проходящий эти уровни, может быть технической спецификацией (Technical Specification, TS) или спецификацией применения (Applicability Statement,

AS). TS — это нормативное описание протокола, формата, сервиса или процедуры. AS — это нормативное определение условий применения одной или более TS для реализации конкретных возможностей в рамках Internet.

Лучший способ освоить процесс стандартизации в Internet — принять участие в одной из существующих или вновь создаваемых рабочих групп. Ознакомиться со списком можно на страницах сервера IETF (www.ietf.org).

С принятыми документами STD, BCP, FYI и другими RFC можно ознакомиться на страницах сервера INTERNIC (www.internic.net).

Средства разработки: PGPsdk

Поскольку с самого начала Зиммерманн публиковал все свои программы в виде исходных текстов (этого требует сама природа криптографии — неопубликованное решение по определению считается ненадежным), PGP давно стала одним из основных учебников криптографии. Первоначально написанная на переносимом C, она провоцировала "заимствование" отдельных блоков и модулей, и кто знает, в скольких продуктах (в том числе, коммерческих), они были использованы3.

В конце октября PGP, Inc. объявила о начале поставки PGPsdk. Этот продукт представляет собой библиотеку криптографических объектов для использования на 32-разрядных платформах Microsoft (Microsoft Visual C++ 5.0), Mac OS (MetroWerks CodeWarrior Version 12) и Unix (Solaris и Linux).

PGPsdk является на сегодня самым полным криптографическим средством разработки и реализует шифрование и аутентификацию, управление ключами

(создание, сертификация, добавление/удаление со связки, проверка действительности, определение уровня надежности), интерфейс с сервером открытых ключей (запрос, подгрузка, удаление и отзыв ключа с удаленного сервера), работу со случайными числами (генерация криптографически стойких псевдослучайных чисел и случайных чисел, базируясь на внешних источниках), а также поддерживает PGP/MIME.

Шифрование и аутентификация выполняются с помощью следующих алгоритмов: Diffie-Hellman, CAST, IDEA, 3DES, DSS, MD5, SHA1 и RIPEMD-160. Поддержка RSA требует отдельного лицензирования.

PGPsdk пока доступен только американцам на коммерческой основе.

14 ноября 1997 года министерство торговли США выдало Pretty Good Privacy, Inc. разрешение на продажу программного обеспечения для шифрования электронной почты с ключами длиной до 128 бит банкам во всем мире. Эта экспортная лицензия — шире, чем лицензии, полученные другими компаниями, поскольку они обычно даются лишь на программное обеспечение, шифрующее непосредственно финансовые транзакции.

Криптомодули, включаемые по лицензии RSA в продукты таких производителей популярного ПО, как Microsoft или Netscape, направляемые на экспорт, содержат ограничение на длину ключа 40 битами. Такая, с позволения сказать, "защита" может быть преодолена не только спецслужбой или криминальной группировкой, но и любым оппонентом, имеющим доступ к ресурсам корпоративной вычислительной сети средних размеров.

В частности, на основе PGP одним из российских банков была построена система накопления транзакций при операциях со смарт-картами.

Интернет-ресурсы PGP

В Интернет можно найти огромное количество связанной с PGP информации. Неплохие ее каталоги расположены на страницах:

PGP, Inc. (www.pgp.com);

PGP.net (http://www.pgp.net);

международного сервера PGP (http://www.pgpi.com);

конференции пользователей PGP (http://pgp.rivertown.net);

Русского Альбома PGP (http://www.geocities.com/SoHo/Studios/1059/).

Содержание