Содержание.

1. Введение.

Научно-техническая революция в последнее время приняла грандиозные масштабы в области информатизации общества на базе современных средств вычислительной техники, связи, а также современных методов автоматизированной обработки информации. Применение этих средств и методов приняло всеобщий характер, а создаваемые при этом информационно-вычислительные системы и сети становятся глобальными как в смысле территориальной распределенности, так и в смысле широты охвата в рамках единых технологий процессов сбора, передачи, накопления, хранения, поиска, переработки информации и выдачи ее для использования.

Информация в современном обществе – одна из самых ценных вещей в жизни, требующая защиты от несанкционированного проникновения лиц не имеющих к ней доступа.

1.1 Экскурс в историю электронной криптографии.

Появление в середине двадцатого столетия первых электронно-вычислительных машин кардинально изменило ситуацию в области шифрования (криптографии). С проникновением компьютеров в различные сферы жизни возникла принципиально новая отрасль - информационная индустрия.

В 60-х и частично в 70-х годах проблема защиты информации решалась достаточно эффективно применением в основном организационных мер. К ним относились прежде всего режимные мероприятия, охрана, сигнализация и простейшие программные средства защиты информации. Эффективность использования указанных средств достигалась за счет концентрации информации на вычислительных центрах, как правило автономных, что способствовало обеспечению защиты относительно малыми средствами.

"Рассосредоточение" информации по местам ее хранения и обработки, чему в немалой степени способствовало появление в огромных количествах дешевых персональных компьютеров и построенных на их основе локальных и глобальных национальных и транснациональных сетей ЭВМ, использующих спутниковые каналы связи, создание высокоэффективных систем разведки и добычи информации, обострило ситуацию с защитой информации.

Проблема обеспечения необходимого уровня защиты информации оказалась (и это предметно подтверждено как теоретическими исследованиями, так и опытом практического решения) весьма сложной, требующей для своего решения не просто осуществления некоторой совокупности научных, научно-технических и организационных мероприятий и применения специфических средств и методов, а создания целостной системы организационных мероприятий и применения специфических средств и методов по защите информации.

Объем циркулирующей в обществе информации стабильно возрастает. Популярность всемирной сети Интренет в последние годы способствует удваиванию информации каждый год. Фактически, на пороге нового тысячелетия человечество создало информационную цивилизацию, в которой от успешной работы средств обработки информации зависит благополучие и даже выживание человечества в его нынешнем качестве. Произошедшие за этот период изменения можно охарактеризовать следующим образом:

- объемы обрабатываемой информации возросли за полвека на несколько порядков;
- доступ к определенным данным позволяет контролировать значительные материальные и финансовые ценности; информация приобрела стоимость, которую даже можно подсчитать;

- характер обрабатываемых данных стал чрезвычайно многообразным и более не сводится к исключительно текстовым данным;
- информация полностью "обезличилась", т.е. особенности ее материального представления потеряли свое значение - сравните письмо прошлого века и современное послание по электронной почте;
- характер информационных взаимодействий чрезвычайно усложнился, и наряду с классической задачей защиты передаваемых текстовых сообщений от несанкционированного прочтения и искажения возникли новые задачи сферы защиты информации, ранее стоявшие и решавшиеся в рамках используемых "бумажных" технологий - например, подпись под электронным документом и вручение электронного документа "под расписку" - речь о подобных "новых" задачах криптографии еще впереди;
- субъектами информационных процессов теперь являются не только люди, но и созданные ими автоматические системы, действующие по заложенной в них программе;
- вычислительные "способности" современных компьютеров подняли на совершенно новый уровень как возможности по реализации шифров, ранее немыслимых из-за своей высокой сложности, так и возможности аналитиков по их взлому.

Перечисленные выше изменения привели к тому, что очень быстро после распространения компьютеров в деловой сфере практическая криптография сделала в своем развитии огромный скачок, причем сразу по нескольким направлениям:

- во-первых, были разработаны стойкие блочные с секретным ключом, предназначенные для решения классической задачи - обеспечения секретности и целостности, передаваемых или хранимых данных, они до сих пор остаются "рабочей лошадкой" криптографии, наиболее часто используемыми средствами криптографической защиты;

- во-вторых, были созданы методы решения новых, нетрадиционных задач сферы защиты информации, наиболее известными из которых являются задача подписи цифрового документа и открытого распределения ключей.

В современном мире информационный ресурс стал одним из наиболее мощных рычагов экономического развития. Владение информацией необходимого качества в нужное время и в нужном месте является залогом успеха в любом виде хозяйственной деятельности. Монопольное обладание определенной информацией оказывается зачастую решающим преимуществом в конкурентной борьбе и предопределяет, тем самым, высокую цену "информационного фактора".

Широкое внедрение персональных ЭВМ вывело уровень "информатизации" деловой жизни на качественно новую ступень. Ныне трудно представить себе фирму или предприятие (включая самые мелкие), которые не были бы вооружены современными средствами обработки и передачи информации. В ЭВМ на носителях данных накапливаются значительные объемы информации, зачастую носящей конфиденциальный характер или представляющей большую ценность для ее владельца.

1.2. Основные задачи криптографии.

Задача криптографии, т.е. тайная передача, возникает только для информации, которая нуждается в защите. В таких случаях говорят, что информация содержит тайну или является защищаемой, приватной, конфиденциальной, секретной. Для наиболее типичных, часто встречающихся ситуаций такого типа введены даже специальные понятия:

- государственная тайна;
- военная тайна;
- коммерческая тайна;
- юридическая тайна;

- врачебная тайна и т. д.

Далее мы будем говорить о защищаемой информации, имея в виду следующие признаки такой информации:

- имеется какой-то определенный круг законных пользователей, которые имеют право владеть этой информацией;
- имеются незаконные пользователи, которые стремятся овладеть этой информацией с тем, чтобы обратить ее себе во благо, а законным пользователям во вред.

1.3 Выводы по разделу 1.

Криптография - это набор методов защиты информационных взаимодействий от отклонений от их нормального, штатного протекания, вызванных злоумышленными действиями различных субъектов, методов, базирующихся на секретных алгоритмах преобразования информации, включая алгоритмы, не являющиеся собственно секретными, но использующие секретные параметры. Исторически первой задачей криптографии была защита передаваемых текстовых сообщений от несанкционированного ознакомления с их содержанием, что нашло отражение в самом названии этой дисциплины, эта защита базируется на использовании "секретного языка", известного только отправителю и получателю, все методы шифрования являются лишь развитием этой философской идеи. С усложнением информационных взаимодействий в человеческом обществе возникли и продолжают возникать новые задачи по их защите, некоторые из них были решены в рамках криптографии, что потребовало развития принципиально новых подходов и методов.

2. Криптографические средства защиты.

Криптографическими средствами защиты называются специальные средства и методы преобразования информации, в результате которых маскируется ее содержание. Основными видами криптографического закрытия являются шифрование и кодирование защищаемых данных. При этом шифрование есть такой вид закрытия, при котором самостоятельному преобразованию подвергается каждый символ закрываемых данных; при кодировании защищаемые данные делятся на блоки, имеющие смысловое значение, и каждый такой блок заменяется цифровым, буквенным или комбинированным кодом. При этом используется несколько различных систем шифрования: заменой, перестановкой, гаммированием, аналитическим преобразованием шифруемых данных. Широкое распространение получили комбинированные шифры, когда исходный текст последовательно преобразуется с использованием двух или даже трех различных шифров.

2.1 Принцыпы работы Криптосистемы.

Типичный пример изображения ситуации, в которой возникает задача криптографии (шифрования) изображён на рисунке №1:

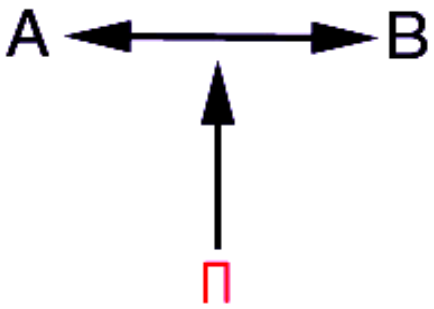


Рис. №1

На рисунке № 1 А и В - законные пользователи защищённой информации, они хотят обмениваться информацией по общедоступному каналу связи. П - незаконный пользователь (противник, хакер), который хочет перехватывать передаваемые по каналу связи сообщения и попытаться извлечь из них интересную для него информацию. Эту простую схему можно считать моделью типичной ситуации, в которой применяются криптографические методы защиты информации или просто шифрование.

Исторически в криптографии закрепились некоторые военные слова (противник, атака на шифр и др.). Они наиболее точно отражают смысл соответствующих криптографических понятий. Вместе с тем широко известная военная терминология, основанная на понятии кода (военно-морские коды, коды Генерального штаба, кодовые книги, кодобозначения и т. п.), уже не применяется в теоретической криптографии. Дело в том, что за последние десятилетия сформировалась теория кодирования - большое научное направление, которое разрабатывает и изучает методы защиты информации от случайных искажений в каналах связи.

Криптография занимается методами преобразования информации, которые бы не позволили противнику извлечь ее из перехватываемых сообщений. При этом по каналу связи передается уже не сама защищаемая информация, а результат ее преобразования с помощью шифра, и для противника возникает сложная задача вскрытия шифра. Вскрытие (взламывание) шифра - процесс получения защищаемой информации из шифрованного сообщения без знания примененного шифра. Противник может пытаться не получить, а уничтожить или модифицировать защищаемую информацию в процессе ее передачи. Это - совсем другой тип угроз для информация, отличный от перехвата и вскрытия шифра. Для защиты от таких угроз разрабатываются свои специфические методы. Следовательно, на пути от одного законного пользователя к другому информация должна защищаться различными

способами, противостоящими различным угрозам. Возникает ситуация цепи из разнотипных звеньев, которая защищает информацию. Естественно, противник будет стремиться найти самое слабое звено, чтобы с наименьшими затратами добраться до информации. А значит, и законные пользователи должны учитывать это обстоятельство в своей стратегии защиты: бессмысленно делать какое-то звено очень прочным, если есть заведомо более слабые звенья ("принцип равнопрочности защиты").

Придумывание хорошего шифра дело трудоемкое. Поэтому желательно увеличить время жизни хорошего шифра и использовать его для шифрования как можно большего количества сообщений. Но при этом возникает опасность, что противник уже разгадал (вскрыл) шифр и читает защищаемую информацию. Если же в шифре сеть сменный ключ то, заменив ключ, можно сделать так, что разработанные противником методы уже не дают эффекта.

2.1 Управление криптографическими ключами.

Под ключом в криптографии понимают сменный элемент шифра, который применяется для шифрования конкретного сообщения. В последнее время безопасность защищаемой информации стала определяться в первую очередь ключом. Сам шифр, шифрмашина или принцип шифрования стали считать известными противнику и доступными для предварительного изучения, но в них появился неизвестный для противника ключ, от которого существенно зависят применяемые преобразования информации. Теперь законные пользователи, прежде чем обмениваться шифрованными сообщениями, должны тайно от противника обменяться ключами или установить одинаковый ключ на обоих концах канала связи. А для противника появилась новая задача - определить ключ, после чего можно легко прочитать зашифрованные на этом ключе сообщения.

Вернемся к формальному описанию основного объекта криптографии (рис. №1). Теперь в него необходимо внести существенное изменение - добавить недоступный для противника секретный канал связи для обмена ключами (см. рис. №2).

Рис. №2

Создать такой канал связи вполне реально, поскольку нагрузка на него, вообще говоря, небольшая. Отметим теперь, что не существует единого шифра, подходящего для всех случаев. Выбор способа шифрования зависит от особенностей информации, ее ценности и возможностей владельцев по защите своей информации. Прежде всего подчеркнем большое разнообразие видов защищаемой информации: документальная, телефонная, телевизионная, компьютерная и т.д. Каждый вид информации имеет свои специфические особенности, и эти особенности сильно влияют на выбор методов шифрования информации. Большое значение имеют объемы и требуемая скорость передачи шифрованной информации. Выбор вида шифра и его параметров существенно зависит от характера защищаемых секретов или тайны. Некоторые тайны (например, государственные, военные и др.) должны сохраняться десятилетиями, а некоторые (например, биржевые) - уже через несколько часов можно разгласить. Необходимо учитывать также и возможности того противника, от которого защищается данная информация.

Одно дело - противостоять одиночке или даже банде уголовников, а другое дело - мощной государственной структуре.

Любая современная криптографическая система основана (построена) на использовании криптографических ключей. Она работает по определенной методологии (процедуре), состоящей из: одного или более алгоритмов шифрования (математических формул); ключей, используемых этими алгоритмами шифрования; системы управления ключами; незашифрованного текста; и зашифрованного текста (шифртекста).

2.1.1. Симметричная (секретная) методология.

В этой методологии и для шифрования, и для расшифровки отправителем и получателем применяется один и тот же ключ, об использовании которого они договорились до начала взаимодействия. Если ключ не был скомпрометирован, то при расшифровке автоматически выполняется аутентификация отправителя, так как только отправитель имеет ключ, с помощью которого можно зашифровать информацию, и только получатель имеет ключ, с помощью которого можно расшифровать информацию. Так как отправитель и получатель - единственные люди, которые знают этот симметричный ключ, при компрометации ключа будет скомпрометировано только взаимодействие этих двух пользователей. Проблемой, которая будет актуальна и для других криптосистем, является вопрос о том, как безопасно распространять симметричные (секретные) ключи.

Алгоритмы симметричного шифрования используют ключи не очень большой длины и могут быстро шифровать большие объемы данных.

Порядок использования систем с симметричными ключами:

- Безопасно создается, распространяется и сохраняется симметричный секретный ключ.

- Отправитель создает электронную подпись с помощью расчета хэш-функции для текста и присоединения полученной строки к тексту
- Отправитель использует быстрый симметричный алгоритм шифрования-расшифровки вместе с секретным симметричным ключом к полученному пакету (тексту вместе с присоединенной электронной подписью) для получения зашифрованного текста. Неявно таким образом производится аудентификация, так как только отправитель знает симметричный секретный ключ и может зашифровать этот пакет. Только получатель знает симметричный секретный ключ и может расшифровать этот пакет.
- Отправитель передает зашифрованный текст. Симметричный секретный ключ никогда не передается по незащищенным каналам связи.
- Получатель использует тот же самый симметричный алгоритм шифрования-расшифровки вместе с тем же самым симметричным ключом (который уже есть у получателя) к зашифрованному тексту для восстановления исходного текста и электронной подписи. Его успешное восстановление аутентифицирует кого-то, кто знает секретный ключ.
- Получатель отделяет электронную подпись от текста.
- Получатель создает другую электронную подпись с помощью расчета хэш-функции для полученного текста.
- Получатель сравнивает две этих электронных подписи для проверки целостности сообщения (отсутствия его искажения)

Доступными сегодня средствами, в которых используется симметричная методология, являются:

- Kerberos, который был разработан для аутентификации доступа к ресурсам в сети, а не для верификации данных. Он использует центральную базу данных, в которой хранятся копии секретных ключей всех пользователей.

- Сети банкоматов (ATM Banking Networks). Эти системы являются оригинальными разработками владеющих ими банков и не продаются. В них также используются симметричные методологии.

2.1.2. Асимметричная (открытая) методология.

В этой методологии ключи для шифрования и расшифровки разные, хотя и создаются вместе. Один ключ делается известным всем, а другой держится в тайне. Данные, зашифрованные одним ключом, могут быть расшифрованы только другим ключом.

Все асимметричные криптосистемы являются объектом атак путем прямого перебора ключей, и поэтому в них должны использоваться гораздо более длинные ключи, чем те, которые используются в симметричных криптосистемах, для обеспечения эквивалентного уровня защиты. Это сразу же сказывается на вычислительных ресурсах, требуемых для шифрования, хотя алгоритмы шифрования на эллиптических кривых могут смягчить эту проблему. Брюс Шнейер в книге "Прикладная криптография: протоколы, алгоритмы и исходный текст на C" приводит в таблице № 1 следующие данные об эквивалентных длинах ключей.

Длина симметричного ключа	Длина асимметричного ключа
56 бит	384 бит
64 бита	512 бит
80 бит	768 бит
112 бит	1792 бита
128 бит	2304 бита

Таблица № 1.

Для того чтобы избежать низкой скорости алгоритмов асимметричного шифрования, генерируется временный симметричный ключ для каждого

сообщения и только он шифруется асимметричными алгоритмами. Само сообщение шифруется с использованием этого временного сеансового ключа и алгоритма шифрования/расшифровки, ранее описанного. Затем этот сеансовый ключ шифруется с помощью открытого асимметричного ключа получателя и асимметричного алгоритма шифрования. После этого этот зашифрованный сеансовый ключ вместе с зашифрованным сообщением передается получателю. Получатель использует тот же самый асимметричный алгоритм шифрования и свой секретный ключ для расшифровки сеансового ключа, а полученный сеансовый ключ используется для расшифровки самого сообщения.

В асимметричных криптосистемах важно, чтобы сеансовые и асимметричные ключи были сопоставимы в отношении уровня безопасности, который они обеспечивают. Если используется короткий сеансовый ключ (например, 40-битовый DES), то не имеет значения, насколько велики асимметричные ключи. Асимметричные открытые ключи уязвимы к атакам прямым перебором отчасти из-за того, что их тяжело заменить. Если атакующий узнает секретный асимметричный ключ, то будет скомпрометирован не только текущее, но и все последующие взаимодействия между отправителем и получателем.

Порядок использования систем с асимметричными ключами:

- Безопасно создаются и распространяются асимметричные открытые и секретные ключи. Секретный асимметричный ключ передается его владельцу. Открытый асимметричный ключ хранится в базе данных и администрируется центром выдачи сертификатов. Подразумевается, что пользователи должны верить, что в такой системе производится безопасное создание, распределение и администрирование ключами. Более того, если создатель ключей и лицо или система, администрирующие их, не одно и то

же, то конечный пользователь должен верить, что создатель ключей на самом деле уничтожил их копию.

- Создается электронная подпись текста с помощью вычисления его хэш-функции. Полученное значение шифруется с использованием асимметричного секретного ключа отправителя, а затем полученная строка символов добавляется к передаваемому тексту (только отправитель может создать электронную подпись).
- Создается секретный симметричный ключ, который будет использоваться для шифрования только этого сообщения или сеанса взаимодействия (сеансовый ключ), затем при помощи симметричного алгоритма шифрования/расшифровки и этого ключа шифруется исходный текст вместе с добавленной к нему электронной подписью - получается зашифрованный текст (шифр-текст).
- Теперь нужно решить проблему с передачей сеансового ключа получателю сообщения.
- Отправитель должен иметь асимметричный открытый ключ центра выдачи сертификатов. Перехват незашифрованных запросов на получение этого открытого ключа является распространенной формой атаки. Может существовать целая система сертификатов, подтверждающих подлинность открытого ключа.
- Отправитель запрашивает у центра сертификатов асимметричный открытый ключ получателя сообщения. Этот процесс уязвим к атаке, в ходе которой атакующий вмешивается во взаимодействие между отправителем и получателем и может модифицировать трафик, передаваемый между ними. Поэтому открытый асимметричный ключ получателя "подписывается" у центра сертификатов. Это означает, что центр сертификатов использовал свой асимметричный секретный ключ для шифрования асимметричного откртого ключа получателя. Только центр сертификатов знает

асимметричный секретный ключ, поэтому есть гарантии того, что открытый асимметричный ключ получателя получен именно от него.

- После получения асимметричный открытый ключ получателя расшифровывается с помощью асимметричного открытого ключа и алгоритма асимметричного шифрования/расшифровки. Естественно, предполагается, что центр сертификатов не был скомпрометирован. Если же он оказывается скомпрометированным, то это выводит из строя всю сеть его пользователей. Поэтому можно и самому зашифровать открытые ключи других пользователей, но где уверенность в том, что они не скомпрометированы?
- Теперь шифруется сеансовый ключ с использованием асимметричного алгоритма шифрования-расшифровки и асимметричного ключа получателя (полученного от центр сертификатов и расшифрованного).
- Зашифрованный сеансовый ключ присоединяется к зашифрованному тексту (который включает в себя также добавленную ранее электронную подпись).
- Весь полученный пакет данных (зашифрованный текст, в который входит помимо исходного текста его электронная подпись, и зашифрованный сеансовый ключ) передается получателю. Так как зашифрованный сеансовый ключ передается по незащищенной сети, он является очевидным объектом различных атак.
- Получатель выделяет зашифрованный сеансовый ключ из полученного пакета.
- Теперь получателю нужно решить проблему с расшифровкой сеансового ключа.
- Получатель должен иметь асимметричный открытый ключ центра выдачи сертификатов.

- Используя свой секретный асимметричный ключ и тот же самый асимметричный алгоритм шифрования получатель расшифровывает сеансовый ключ.
- Получатель применяет тот же самый симметричный алгоритм шифрования-расшифровки и расшифрованный симметричный (сеансовый) ключ к зашифрованному тексту и получает исходный текст вместе с электронной подписью.
- Получатель отделяет электронную подпись от исходного текста.
- Получатель запрашивает у центр сертификатов асимметричный открытый ключ отправителя.
- Как только этот ключ получен, получатель расшифровывает его с помощью открытого ключа центр сертификатов и соответствующего асимметричного алгоритма шифрования-расшифровки.
- Затем расшифровывается хэш-функция текста с использованием открытого ключа отправителя и асимметричного алгоритма шифрования-расшифровки.
- Повторно вычисляется хэш-функция полученного исходного текста.
- Две эти хэш-функции сравниваются для проверки того, что текст не был изменен.

2.2 Алгоритмы шифрования

Алгоритмы шифрования с использованием ключей предполагают, что данные не сможет прочитать никто, кто не обладает ключом для их расшифровки. Они могут быть разделены на два класса, в зависимости от того, какая методология криптосистем напрямую поддерживается ими.

2.2.1 Симметричные алгоритмы

Для шифрования и расшифровки используются одни и те же алгоритмы. Один и тот же секретный ключ используется для шифрования и расшифровки. Этот тип алгоритмов используется как симметричными, так и асимметричными криптосистемами.

Таблица № 2.

Тип	Описание
DES (Data Encrypt Standard)	Популярный алгоритм шифрования, используемый стандарт шифрования данных правительством США. Шифруется блок из 64 бит, используется 64-битовый кл (требуется только 56 бит), 16 проходов Может работать в 4 режимах: • Электронная кодовая книга (ECB-Electronic Code Book обычный DES, использует два различных алгоритма. • Цепочечный режим (CBC-Cipher Block Chaining), котором шифрование шифрование блока данных зависит результатов шифрования предыдущих блоков данных. • Обратная связь по выходу (OFB-Output Feedbac используется как генератор случайных чисел. • Обратная связь по шифратору (CFB-Cipher Feedbac используется для получения кодов аутентифика сообщений.
3-DES или тройной DES	64-битный блочный шифратор, использует DES 3 раза тремя различными 56-битными ключами. Достаточно стоек ко всем атакам

Каскадный 3-DE	Стандартный тройной DES, к которому добавлен механи обратной связи, такой как CBC, OFB или CFB Очень стоек ко всем атакам.
FEAL (быстрый алгоритм шифрования)	Блочный шифратор, используемый как альтернатива DES Вскрыт, хотя после этого были предложены новые версии.
IDEA (международный алгоритм шифрования)	64-битный блочный шифратор, 128-битовый ключ, проходов Предложен недавно; хотя до сих пор не прошел полн проверки, чтобы считаться надежным, считается бо лучшим, чем DES

ipjack	Разработано АНБ в ходе проектов правительства CI "Clipper" и "Capstone". До недавнего времени был секретным, но его стойкость зависела только от того, что он был секретным. 64-битный блочный шифратор, 80-битовые клю используются в режимах ECB, CFB, OFB или CBC, прохода
RC2	64-битный блочный шифратор, ключ переменного размера Приблизительно в 2 раза быстрее, чем DES Может использоваться в тех же режимах, что и DI включая тройное шифрование. Конфиденциальный алгоритм, владельцем которого являет RSA Data Security
RC4	Потоковый шифр, байт-ориентированный, с ключ переменного размера. Приблизительно в 10 раз быстрее DES. Конфиденциальный алгоритм, которым владеет RSA D Security
RC5	Имеет размер блока 32, 64 или 128 бит, ключ с длиной о до 2048 бит, от 0 до 255 проходов Быстрый блочный шифр Алгоритм, которым владеет RSA Data Security
CAST	64-битный блочный шифратор, ключи длиной от 40 до бит, 8 проходов Неизвестно способов вскрыть его иначе как путем прямо перебора.
Blowfish.	64-битный блочный шифратор, ключ переменного разме до 448 бит, 16 проходов, на каждом проходе выполняю

	перестановки, зависящие от ключа, и подстанов зависящие от ключа и данных. Быстрее, чем DES Разработан для 32-битных машин
Устройство с одноразовыми ключами	Шифратор, который нельзя вскрыть. Ключом (который имеет ту же длину, что и шифруем данные) являются следующие 'n' бит из массива случай созданных бит, хранящихся в этом устройстве. отправителя и получателя имеются одинаковые устройст После использования биты разрушаются, и в следующий используются другие биты.
Поточные шифр	Быстрые алгоритмы симметричного шифрования, обыч оперирующие битами (а не блоками бит). Разработаны как аналог устройства с одноразовы ключами, и хотя не являются такими же безопасными, оно, по крайней мере практичны.

2.2.2 Асимметричные алгоритмы

Асимметричные алгоритмы используются в асимметричных криптосистемах для шифрования симметричных сеансовых ключей (которые используются для шифрования самих данных).

Используется два разных ключа - один известен всем, а другой держится в тайне. Обычно для шифрования и расшифровки используется оба этих ключа. Но данные, зашифрованные одним ключом, можно расшифровать только с помощью другого ключа.

Таблица № 3.

Тип	Описание
RSA	Популярный алгоритм асимметричного шифрован стойкость которого зависит от сложности факторизац больших целых чисел.
ECC (криптосисте на основе эллиптических кривых)	Использует алгебраическую систему, которая описывается терминах точек эллиптических кривых, для реализац асимметричного алгоритма шифрования. Является конкурентом по отношению к друг асимметричным алгоритмам шифрования, так как п эквивалентной стойкости использует ключи меньшей длины имеет большую производительность. Современные его реализации показывают, что эта систе гораздо более эффективна, чем другие системы с открыты ключами. Его производительность приблизительно на поряд выше, чем производительность RSA, Диффи-Хеллмана и DS
Эль-Гамаль.	Вариант Диффи-Хеллмана, который может быть использо как для шифрования, так и для электронной подписи.

2.3 Хэш-функции

Хэш-функции являются одним из важных элементов криптосистем на основе ключей. Их относительно легко вычислить, но почти невозможно расшифровать. Хэш-функция имеет исходные данные переменной длины и возвращает строку фиксированного размера (иногда называемую дайджестом сообщения - MD), обычно 128 бит. Хэш-функции используются для обнаружения модификации сообщения (то есть для электронной подписи).

Таблица № 4.

Тип	Описание
MD2	Самая медленная, оптимизирована для 8-битовых машин
MD4	Самая быстрая, оптимизирована для 32-битных машин Не так давно взломана
MD5	Наиболее распространенная из семейства MD-функций. Похожа на MD4, но средства повышения безопасности дела ее на 33% медленнее, чем MD4 Обеспечивает целостность данных Считается безопасной
SHA (Secure Hash Algorithm	Создает 160-битное значение хэш-функции из исходных данн переменного размера. Предложена NIST и принята правительством США как станда Предназначена для использования в стандарте DSS

2.4 Механизмы аутентификации

Эти механизмы позволяют проверить подлинность личности участника взаимодействия безопасным и надежным способом.

Таблица № 5.

Тип	**Описание**
Пароли или PIN-коды (персональные идентификационные номера)	Что-то, что знает пользователь и что также зн другой участник взаимодействия. Обычно аутентификация производится в 2 этапа. Может организовываться обмен паролями д взаимной аутентификации.
Одноразовый пароль	Пароль, который никогда больше не используется.

	Часто используется постоянно меняющееся значен которое базируется на постоянном пароле.
CHAP (протокол аутентификации запрос-ответ)	Одна из сторон инициирует аутентификацию помощью посылки уникального и непредсказуемо значения "запрос" другой стороне, а другая сторо посылает вычисленный с помощью "запроса" секрета ответ. Так как обе стороны владеют секрето то первая сторона может проверить правильно ответа второй стороны.
Встречная проверка (Callback)	Телефонный звонок серверу и указание име пользователя приводит к тому, что сервер затем с звонит по номеру, который указан для этого име пользователя в его конфигурационных данных.

2.5 Электронные подписи и временные метки

Электронная подпись позволяет проверять целостность данных, но не обеспечивает их конфиденциальность. Электронная подпись добавляется к сообщению и может шифроваться вместе с ним при необходимости сохранения данных в тайне. Добавление временных меток к электронной подписи позволяет обеспечить ограниченную форму контроля участников взаимодействия.

Таблица № 6.

Тип	Комментарии
DSA (Digital Signature Authorizati	Алгоритм с использованием открытого ключа д создания электронной подписи, но не для шифрования. Секретное создание хэш-значения и публичная провер ее - только один человек может создать хэш-значен сообщения, но любой может проверить ее корректность

	Основан на вычислительной сложности взят логарифмов в конечных полях.
RSA	Запатентованная RSA электронная подпись, котор позволяет проверить целостность сообщения и лично лица, создавшего электронную подпись. Отправитель создает хэш-функцию сообщения, а зат шифрует ее с использованием своего секретного клю Получатель использует открытый ключ отправителя д расшифровки хэша, сам рассчитывает хэш для сообщен и сравнивает эти два хэша.
MAC (код аутентификации сообщения)	Электронная подпись, использующая схемы хэширован аналогичные MD или SHA, но хэш-значение вычисляет с использованием как данных сообщения, так секретного ключа.
DTS (служба электронных временных меток)	Выдает пользователям временные метки, связанные данными документа

2.6. Стойкость шифра.

Способность шифра противостоять всевозможным атакам на него называют стойкостью шифра. Под атакой на шифр понимают попытку вскрытия этого шифра. Понятие стойкости шифра является центральным для криптографии. Хотя качественно понять его довольно легко, но получение строгих доказуемых оценок стойкости для каждого конкретного шифра - проблема нерешенная. Это объясняется тем, что до сих пор нет необходимых для решения такой проблемы математических результатов. Поэтому стойкость конкретного шифра оценивается только путем всевозможных попыток его

вскрытия и зависит от квалификации криптоаналитиков, атакующих шифр. Такую процедуру иногда называют проверкой стойкости. Важным подготовительным этапом для проверки стойкости шифра является продумывание различных предполагаемых возможностей, с помощью которых противник может атаковать шифр. Появление таких возможностей у противника обычно не зависит от криптографии, это является некоторой внешней подсказкой и существенно влияет на стойкость шифра. Поэтому оценки стойкости шифра всегда содержат те предположения о целях и возможностях противника, в условиях которых эти оценки получены. Прежде всего, как это уже отмечалось выше, обычно считается, что противник знает сам шифр и имеет возможности для его предварительного изучения. Противник также знает некоторые характеристики открытых текстов, например, общую тематику сообщений, их стиль, некоторые стандарты, форматы и т.д.

Из более специфических приведем еще три примера возможностей противника:

- противник может перехватывать все шифрованные сообщения, но не имеет соответствующих им открытых текстов;
- противник может перехватывать все шифрованный сообщения и добывать соответствующие им открытые тексты;
- противник имеет доступ к шифру (но не к ключам!) и поэтому может зашифровывать и дешифровывать любую информацию;

2.7 Выводы по разделу 2.

Подводя итоги вышесказанного, можно уверенно заявить, что криптографическими системами защиты называються совокупность различных методов и средств, благодаря которым исходная информация кодируеться, передаеться и расшифровываеться.

Существуют различные криптографические системы защиты, которые мы можем разделить на две группы: с использованием ключа и без него. Криптосистемы без применения ключа в совремом мире не используються т.к. очень дорогостоющие и ненадёжные.

Были расмотренны основные методологии: семметричная и асиметричная. Обе методологии используют ключ (сменный элемент шифра).

Симметричные и асиметричные алгоритмы, описанные выше, сведены в таблицу, из которой можно понять какие алгоритмы наиболее подходят к той или иной задаче.

Остальная информация пердставленная во второй главе очень разнообразна. На её основе сложно сделать вывод, какие алгоритмы хеш-функций, механизмов аутетификации и электронных подписей наиболее продвинутые, все они в разной ситуации могут показать себя с лучшей стороны.

На протяжении многих веков среди специалистов не утихали споры о стойкости шифров и о возможности построения абсолютно стойкого шифра.

Практическая часть

Система шифрования Цезаря - частный случай шифра простой замены. Метод основан на замене каждой буквы сообщения на другую букву того же алфавита, путем смещения от исходной буквы на K букв.

Шифры сложной замены

Шифр Гронсфельда состоит в модификации шифра Цезаря числовым ключом. Для этого под буквами сообщения записывают цифры числового ключа. Если ключ короче сообщения, то его запись циклически повторяют. Шифротекст получают примерно также как в шифре Цезаря, но отсчитывают не третью букву по алфавиту (как в шифре Цезаря), а ту, которая смещена по алфавиту на соответствующую цифру ключа.

Пусть в качестве ключа используется группа из трех цифр - 314, тогда

Сообщение СОВЕРШЕННО СЕКРЕТНО

Ключ 3143143143143143143

Шифровка ФПИСЬИОССАХИЛФИУСС

В шифрах многоалфавитной замены для шифрования каждого символа исходного сообщения применяется свой шифр простой замены (свой алфавит).

	АБВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭЮЯ_
А	АБВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭЮЯ_
Б	_АБВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭЮЯ
В	Я_АБВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭЮ
Г	ЮЯ_АБВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭ
.	……
Я	ВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭЮЯ_АБ
_	БВГДЕЁЖЗИКЛМНОПРСТУФХЧШЩЪЫЬЭЮЯ_А

Каждая строка в этой таблице соответствует одному шифру замены аналогично шифру Цезаря для алфавита, дополненного пробелом. При шифровании сообщения его выписывают в строку, а под ним ключ. Если ключ оказался короче сообщения, то его циклически повторяют. Шифротекст получают, находя символ в колонке таблицы по букве текста и строке, соответствующей букве ключа. Например, используя ключ АГАВА, из сообщения ПРИЕЗЖАЮ ШЕСТОГО получаем следующую шифровку:

Сообщение	ПРИЕЗЖАЮ_ШЕСТОГО
Ключ	АГАВААГАВААГАВАА
Шифровка	ПНИГЗЖЮЮЮАЕОТМГО

В компьютере такая операция соответствует сложению кодов ASCII символов сообщения и ключа по модулю 256.

Внешний вид текстового редактора "Блокнот" с функцией шифрования-дешифрования классическими криптографическими методами представлен на рисунке 1.

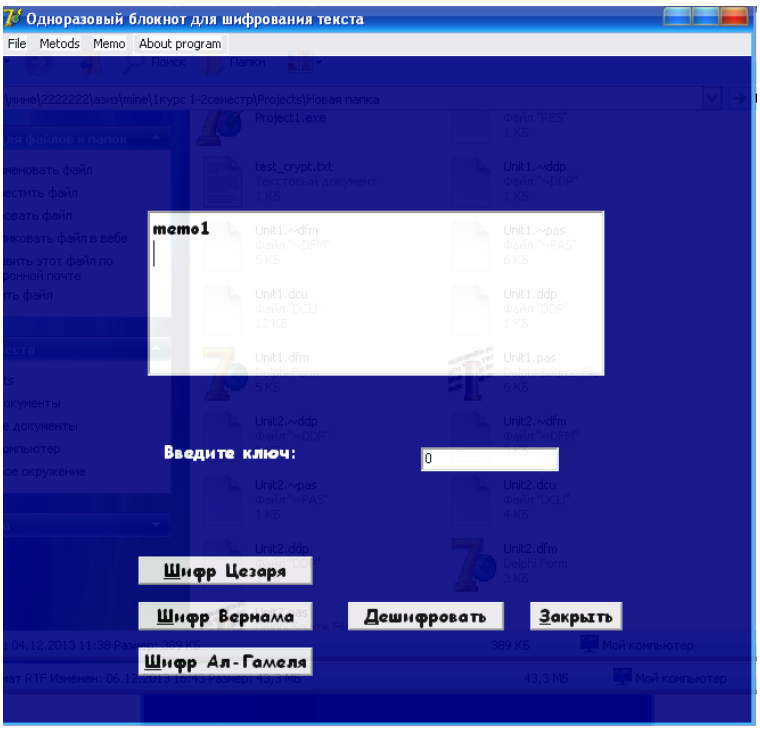


Рисунок 2 - Главная форма текстового редактора

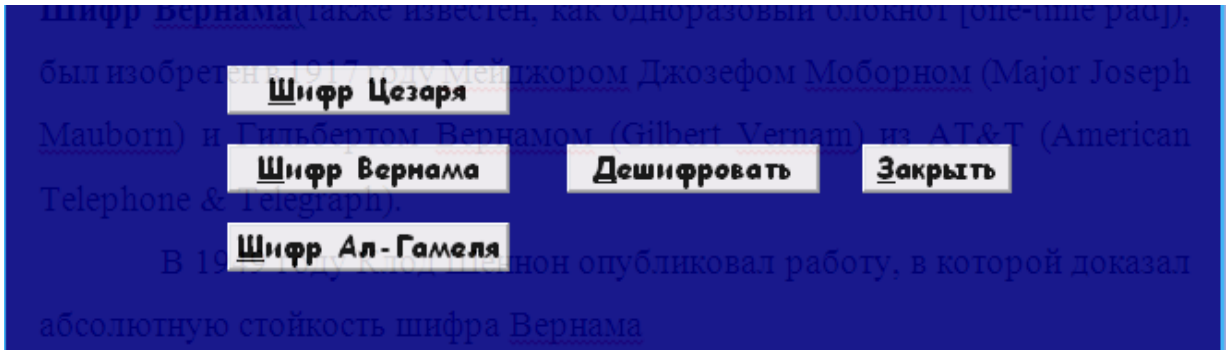


Рисунок 3 Основные кнопки для шифрования и дешифрования

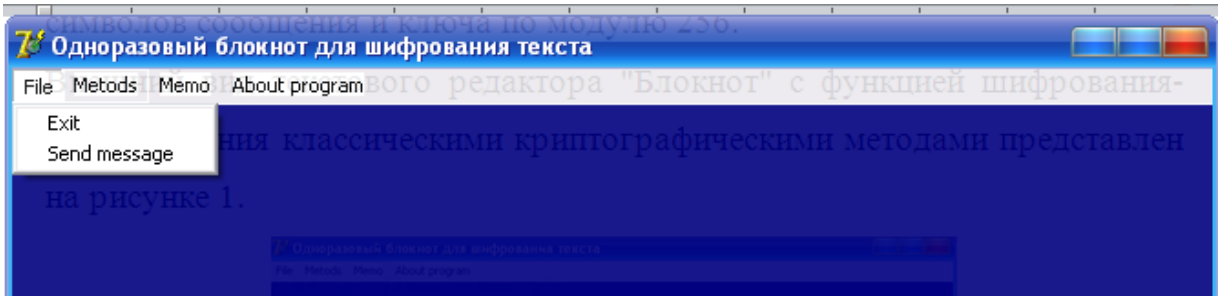


Рисунок 4 Меню файл программы и ее компоненты

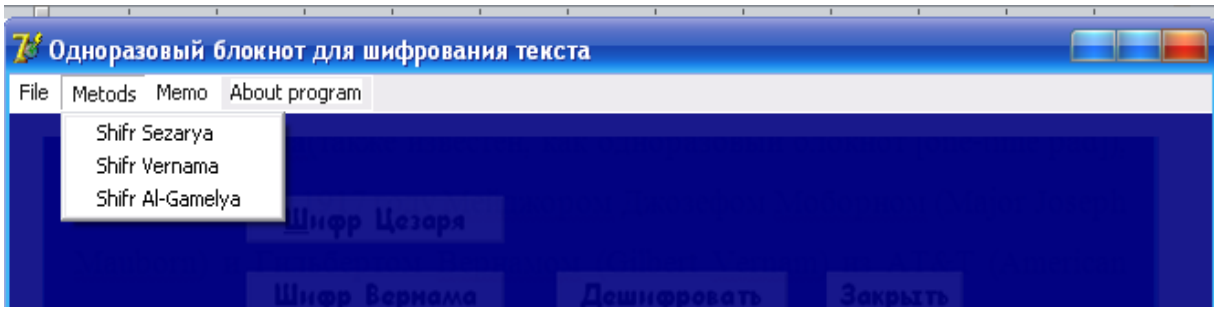


Рисунок 5 Меню Методы и ее компоненты

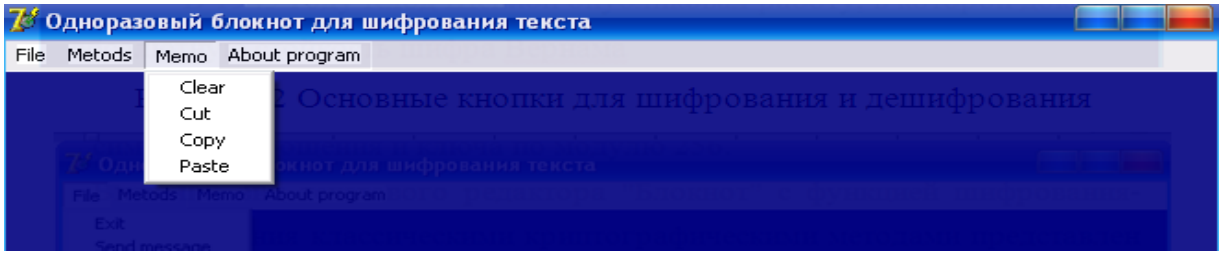
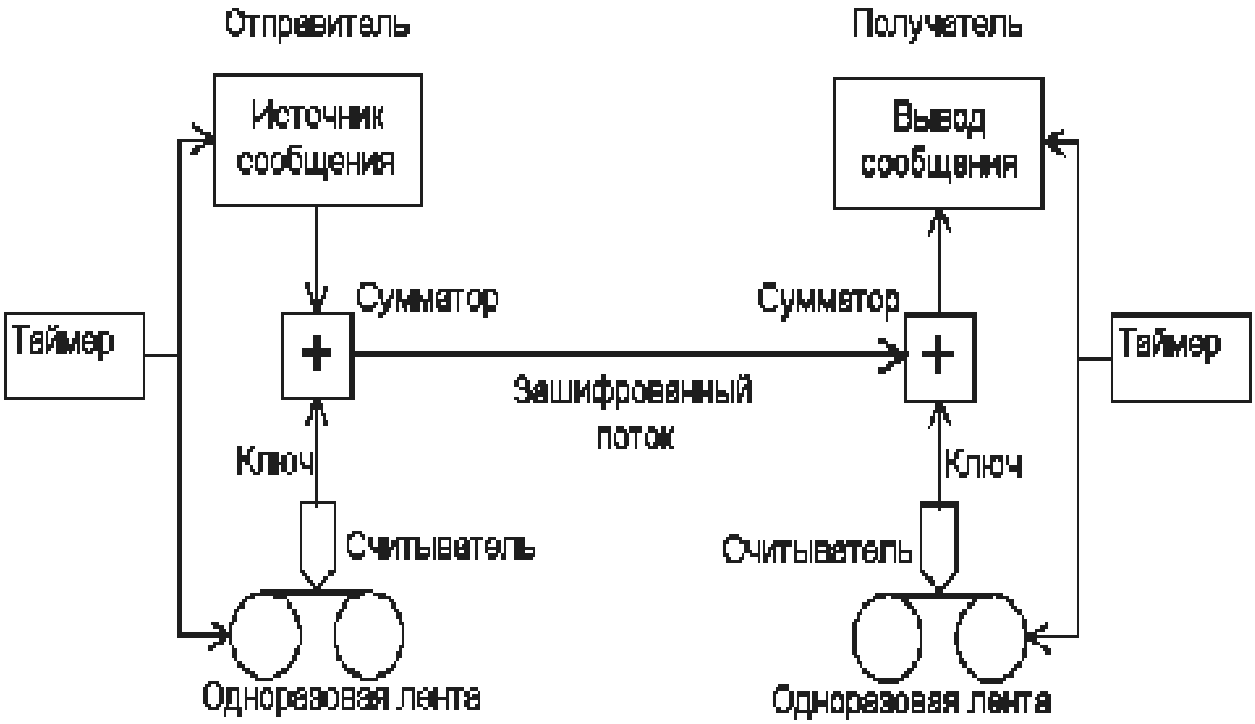


Рисунок 6 Меню Мемо и ее компоненты

Шифр Вернама(также известен, как одноразовый блокнот [one-time pad]), был изобретен в 1917 году Мейджором Джозефом Моборном (Major Joseph Mauborn) и Гильбертом Вернамом (Gilbert Vernam) из AT&T (American Telephone & Telegraph).

В 1949 году Клод Шеннон опубликовал работу, в которой доказал абсолютную стойкость шифра Вернама



Криптосистема Эль Гамаля

Данная система является альтернативой RSA и при равном значении ключа обеспечивает ту же криптостойкость.

В отличие от RSA метод Эль-Гамаля основан на проблеме дискретного логарифма. Этим он похож на алгоритм Диффи-Хелмана.

Шифросистема ElGamal — была предложена в 1984 году

Схему ElGamal можно использовать как для цифровых подписей, так и для шифрования. Его безопасность основана на трудности вычисления дискретных логарифмов в конечном поле.

Заключение.

Криптография сегодня - это важнейшая часть всех информационных систем: от электронной почты до сотовой связи, от доступа к сети Internet до электронной наличности. Криптография обеспечивает подотчетность, прозрачность, точность и конфиденциальность. Она предотвращает попытки мошенничества в электронной коммерции и обеспечивает юридическую силу финансовых транзакций. Криптография помогает установить вашу личность, но и обеспечивает вам анонимность. Она мешает хулиганам испортить сервер и не позволяет конкурентам залезть в ваши конфиденциальные документы. А в будущем, по мере того как коммерция и коммуникации будут все теснее связываться с компьютерными сетями, криптография станет жизненно важной.

Но присутствующие на рынке криптографические средства не обеспечивают того уровня защиты, который обещан в рекламе. Большинство продуктов разрабатывается и применяется отнюдь не в сотрудничестве с криптографами. Этим занимаются инженеры, для которых криптография - просто еще один компонент программы. Но криптография - это не компонент. Нельзя обеспечить безопасность системы, «вставляя» криптографию после ее разработки. На каждом этапе, от замысла до инсталляции, необходимо осознавать, что и зачем вы делаете.

Для того, чтобы грамотно реализовать собственную криптосистему, необходимо не только ознакомится с ошибками других и понять причины, по которым они произошли, но и, возможно, применять особые защитные приемы программирования и специализированные средства разработки.

На обеспечение компьютерной безопасности тратятся миллиарды долларов, причем большая часть денег выбрасывается на негодные продукты. К сожалению, коробка со слабым криптографическим продуктом выглядит так же, как коробка со стойким. Два криптопакета для электронной почты могут

иметь схожий пользовательский интерфейс, но один обеспечит безопасность, а второй допустит подслушивание. Сравнение может указывать сходные черты двух программ, но в безопасности одной из них при этом зияют дыры, которых лишена другая система. Опытный криптограф сможет определить разницу между этими системами. То же самое может сделать и злоумышленник.

На сегодняшний день компьютерная безопасность - это карточный домик, который в любую минуту может рассыпаться. Очень многие слабые продукты до сих пор не были взломаны только потому, что они мало используются. Как только они приобретут широкое распространение, они станут притягивать к себе преступников. Пресса тут же придаст огласке эти атаки, подорвав доверие публики к этим криптосистемам. В конце концов, победу на рынке криптопродуктов определит степень безопасности этих продуктов.

Литература.

1. А.Ю.Винокуров. ГОСТ не прост..,а очень прост, М., Монитор.–1995.
2. А.Ю.Винокуров. Еще раз про ГОСТ., М., Монитор.–1995.–N5.
3. А.Ю.Винокуров. Алгоритм шифрования ГОСТ 28147-89, его использование и реализация для компьютеров платформы Intel x86., Рукопись, 1997.
4. А.Ю.Винокуров. Как устроен блочный шифр?, Рукопись, 1995.
5. М.Э.Смид, Д.К.Бранстед. Стандарт шифрования данных: прошлое и будущее. /пер. с англ./ М., Мир, ТИИЭР.–1988.–т.76.–N5.
6. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования ГОСТ 28147–89, М., Госстандарт, 1989.
7. Б.В.Березин, П.В.Дорошкевич. Цифровая подпись на основе традиционной криптографии//Защита информации, вып.2.,М.: МП "Ирбис-II",1992.
8. W.Diffie,M.E.Hellman. New Directions in cryptography// IEEE Trans. Inform. Theory, IT-22, vol 6 (Nov. 1976), pp. 644-654.
9. У.Диффи. Первые десять лет криптографии с открытым ключом. /пер. с англ./ М., Мир, ТИИЭР.–1988.–т.76.–N5.
10. Водолазкий В., "Стандарт шифрования ДЕС", Монитор 03-04 1992 г. С.
11. Воробьев, "Защита информации в персональных ЭВМ", изд. Мир, 1993
12. Ковалевский В., "Криптографические методы", Компьютер Пресс 05.93
13. Мафтик С., "Механизмы защиты в сетях ЭВМ", изд. Мир, 1993 г.

Листинг программы

```
UNIT UNIT1;

INTERFACE

USES
  WINDOWS, MESSAGES, SYSUTILS, VARIANTS, CLASSES, GRAPHICS,
CONTROLS, FORMS,
  DIALOGS, STDCTRLS, MENUS, EXTCTRLS;

TYPE
  TFORM1 = CLASS(TFORM)
    BUTTON1: TBUTTON;
    BUTTON2: TBUTTON;
    MEMO1: TMEMO;
    BUTTON3: TBUTTON;
    BUTTON4: TBUTTON;
    EDIT1: TEDIT;
    LABEL1: TLABEL;
    BUTTON5: TBUTTON;
    MAINMENU1: TMAINMENU;
    METODS1: TMENUITEM;
    MEMO2: TMENUITEM;
    CLEAR1: TMENUITEM;
    SHIFRZAMENI1: TMENUITEM;
    SHIFRVERNAMA1: TMENUITEM;
    SHIFRALGAMELYA1: TMENUITEM;
    FILE1: TMENUITEM;
    EXIT1: TMENUITEM;
    CUT1: TMENUITEM;
    COPY1: TMENUITEM;
    PASTE1: TMENUITEM;
    ABOUTPROGRAM1: TMENUITEM;
    PROCEDURE BUTTON1CLICK(SENDER: TOBJECT);
    PROCEDURE BUTTON2CLICK(SENDER: TOBJECT);
    PROCEDURE BUTTON3CLICK(SENDER: TOBJECT);
    PROCEDURE SCROLLBAR1SCROLL(SENDER: TOBJECT; SCROLLCODE:
TSCROLLCODE;
```

```
    VAR SCROLLPOS: INTEGER);
   PROCEDURE BUTTON4CLICK(SENDER: TOBJECT);
   PROCEDURE BUTTON5CLICK(SENDER: TOBJECT);
   PROCEDURE CLEAR1CLICK(SENDER: TOBJECT);
   PROCEDURE SHIFRZAMENI1CLICK(SENDER: TOBJECT);
   PROCEDURE EXIT1CLICK(SENDER: TOBJECT);
   PROCEDURE CUT1CLICK(SENDER: TOBJECT);
   PROCEDURE COPY1CLICK(SENDER: TOBJECT);
   PROCEDURE PASTE1CLICK(SENDER: TOBJECT);
   PROCEDURE SHIFRVERNAMA1CLICK(SENDER: TOBJECT);
   PROCEDURE SHIFRALGAMELYA1CLICK(SENDER: TOBJECT);
   PROCEDURE ABOUTPROGRAM1CLICK(SENDER: TOBJECT);
  PRIVATE
   { PRIVATE DECLARATIONS }
  PUBLIC
   { PUBLIC DECLARATIONS }
  END;  VAR
  FORM1: TFORM1;
STR,KEY,SHIFR:STRING;
PRSHIFR,PR:ARRAY[1..256] OF STRING;
J,K,L,M,N,S,X,Y :INTEGER;
QWER:CHAR;
IMPLEMENTATION
USES UNIT2;
{$R *.DFM}
FUNCTION CRYPT(VARSTR: WIDESTRING):WIDESTRING;
VAR
 K: INTEGER;
 S: STRING;
BEGIN
   RANDSEED:=100;
   S:=VARSTR;
   FOR K:=1 TO LENGTH(S) DO
    S[K]:=CHR(ORD(S[K]) XOR (RANDOM(255)+1));
 CRYPT:=S;  END;
```

```
PROCEDURE TFORM1.BUTTON1CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.TEXT := CRYPT(MEMO1.TEXT);
  MEMO1.LINES.SAVETOFILE('D:TEST_CRYPT.TXT');
END;
PROCEDURE TFORM1.BUTTON2CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.LINES.LOADFROMFILE('D:TEST_CRYPT.TXT');
   MEMO1.TEXT := CRYPT(MEMO1.TEXT);
END;
PROCEDURE TFORM1.BUTTON3CLICK(SENDER: TOBJECT);
BEGIN  CLOSE; END;
PROCEDURE TFORM1.SCROLLBAR1SCROLL(SENDER: TOBJECT; SCROLLCODE:
TSCROLLCODE;
  VAR SCROLLPOS: INTEGER);
BEGIN
MEMO1.SHOW;
END;
PROCEDURE TFORM1.BUTTON4CLICK(SENDER: TOBJECT);
VAR I:INTEGER;
BEGIN
STR:=MEMO1.TEXT;
KEY:=EDIT1.TEXT;
N:=LENGTH(STR);
M:=LENGTH(KEY);
IF M<N THEN X:=N-M;
MEMO1.CLEAR;
WHILE N>=0 DO
BEGIN
IF N>M THEN S:=M ELSE S:=N;
FOR I:=1 TO S DO
BEGIN
  L:=ORD(STR[I]);
  K:=ORD(KEY[I]);
QWER:= CHR(L XOR K);
```

```
PRSHIFR[I]:=QWER;
MEMO1.TEXT:=MEMO1.TEXT+PRSHIFR[I];
END;
IF (N-M)>M THEN X:=M ELSE X:=N-M;
DELETE(STR,1,X);
N:=N-M; END; END;
PROCEDURE TFORM1.BUTTON5CLICK(SENDER: TOBJECT);
VAR S:STRING;
PROCEDURE CODE(VAR TEXT: STRING; PASSWORD: STRING;
  DECODE: BOOLEAN);
VAR
  I, PASSWORDLENGTH: INTEGER;
  SIGN: SHORTINT;
BEGIN
  PASSWORDLENGTH := LENGTH(PASSWORD);
  IF PASSWORDLENGTH = 0 THEN
    EXIT;
  IF DECODE THEN
    SIGN := -1   ELSE     SIGN := 1;
  FOR I := 1 TO LENGTH(TEXT) DO
    TEXT[I] := CHR(ORD(TEXT[I]) + SIGN *
      ORD(PASSWORD[I MOD PASSWORDLENGTH + 1]));
END; BEGIN
S := MEMO1.TEXT;
  CODE(S, EDIT1.TEXT, FALSE);
  MEMO1.TEXT := S;
END;
PROCEDURE TFORM1.CLEAR1CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.CLEAR; END;
PROCEDURE TFORM1.SHIFRZAMENI1CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.TEXT := CRYPT(MEMO1.TEXT);
   MEMO1.LINES.SAVETOFILE('D:TEST_CRYPT.TXT');
END;
```

```
PROCEDURE TFORM1.EXIT1CLICK(SENDER: TOBJECT);
BEGIN CLOSE;
END;
PROCEDURE TFORM1.CUT1CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.CUTTOCLIPBOARD;
END;
PROCEDURE TFORM1.COPY1CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.COPYTOCLIPBOARD;
END;
PROCEDURE TFORM1.PASTE1CLICK(SENDER: TOBJECT);
BEGIN
MEMO1.PASTEFROMCLIPBOARD;
END;
PROCEDURE TFORM1.SHIFRVERNAMA1CLICK(SENDER: TOBJECT);
VAR I:INTEGER;
BEGIN
STR:=MEMO1.TEXT;
KEY:=EDIT1.TEXT;
N:=LENGTH(STR);
M:=LENGTH(KEY);
IF M<N THEN X:=N-M;
MEMO1.CLEAR;
WHILE N>=0 DO
BEGIN
IF N>M THEN S:=M ELSE S:=N;
FOR I:=1 TO S DO
BEGIN
  L:=ORD(STR[I]);
  K:=ORD(KEY[I]);
QWER:= CHR(L XOR K);
PRSHIFR[I]:=QWER;
MEMO1.TEXT:=MEMO1.TEXT+PRSHIFR[I];
END;
```

```
IF (N-M)>M THEN X:=M ELSE X:=N-M;
DELETE(STR,1,X);
N:=N-M;
END;
END;
PROCEDURE TFORM1.SHIFRALGAMELYA1CLICK(SENDER: TOBJECT);
VAR S:STRING;
PROCEDURE CODE(VAR TEXT: STRING; PASSWORD: STRING;
  DECODE: BOOLEAN);
VAR
  I, PASSWORDLENGTH: INTEGER;
  SIGN: SHORTINT;
BEGIN
  PASSWORDLENGTH := LENGTH(PASSWORD);
  IF PASSWORDLENGTH = 0 THEN
    EXIT;
  IF DECODE THEN
    SIGN := -1
  ELSE
    SIGN := 1;
  FOR I := 1 TO LENGTH(TEXT) DO
    TEXT[I] := CHR(ORD(TEXT[I]) + SIGN *
      ORD(PASSWORD[I MOD PASSWORDLENGTH + 1]));
END;
BEGIN
S := MEMO1.TEXT;
  CODE(S, EDIT1.TEXT, FALSE);
  MEMO1.TEXT := S;
END;
PROCEDURE TFORM1.ABOUTPROGRAM1CLICK(SENDER: TOBJECT);
BEGIN
FORM2.SHOWMODAL;
END;
END.
```