МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
**«Санкт-Петербургский государственный технологический институт
(технический университет)»**

Г.В.Кузнецова

**Учебно-методические материалы и
задание к контрольным работам по дисциплине**

Защита информации

**Укрупненная группа направлений подготовки (специальностей)
09.00.00 – Информатика и вычислительная техника**

Квалификация (степень) выпускника
Бакалавр

Форма обучения
заочная

Факультет **информационных технологий и управления**
Кафедра **систем автоматизированного проектирования и управления**

**Санкт-Петербург
2015**

Оглавление

Задание на выполнение работ по дисциплине "Защита информации"

Предлагаемые работы используется для проведения контроля усвоения учебного материала по дисциплине. В течение семестра проводится комплексная проверка знаний, которая включает три самостоятельных модуля.

Задание на выполнение работ по дисциплине включает три самостоятельных модуля

1. Аналитическое исследование

Задание подразумевает поиск информации в открытых источниках (в том числе Internet), ее аналитический обзор и сравнительный анализ результатов, и представление в виде отчета в тестовом формате или в виде мультимедийной презентации (в электронном и бумажном варианте).

2. Тестовое задание

Подготовьтесь к коллоквиуму по теме "Криптография". Ответьте на поставленные вопросы.

3. Практическое задание предполагает разработку программного продукта по заданной теме. Необходимо наличие исполняемого модуля, исходных данных и руководства пользователя, описывающего алгоритм работы программы, условия использования, принятые допущения и ограничения (в электронном и бумажном варианте).

Выбор варианта задания осуществляется по следующим правилам

1. Аналитическое исследование

Номер варианта задания определяется первой буквой фамилии студента по таблице

	Номер варианта									
	1	2	3	4	5	6	7	8	9	10
Первая буква фамилии	А л х	Б М ц	В Н Ш	Г О щ	Д П э	Е Р ю	Ж С я	З т	И у	К ф

2. Тестовое задание: для всех - ответьте на поставленные вопросы.

3**. Практическое задание**: номер вариант работы по выбору студента

Для подготовки отчетов и выполнения заданий рекомендуется ознакомить со учебно-методическими материалами и литературой.

Задание на выполнение работ

1. Аналитическое исследование_____Методы и средства обеспечения безопасности

Задание подразумевает поиск информации в открытых источниках (в том числе Internet), ее аналитический обзор и сравнительный анализ результатов, и представление в виде отчета в тестовом формате или в виде мультимедийной презентации (в электронном и бумажном варианте).

1. Политика безопасности организации. Правовой, организационный и технический аспект.
2. Обеспечение безопасности сайтов. Цель и сущность, объекты охраны, методы и средства.
3. Обеспечение безопасности базы данных. Угрозы, особенности, методы и средства защиты.
4. Биометрические средства идентификации.
5. Хэш-функции: понятие, принцип функционирования, свойства, особенности использование. Сравнительный анализ.
6. Электронная подпись: понятие, принцип функционирования, свойства, особенности использование. Инфраструктура открытых ключей.
7. Обеспечение безопасности при удаленном доступе к ресурсам.
8. Безопасность VPN.
9. Обеспечение целостности информации
10. Обеспечение безопасности сети организации

2. Тестовое задания Криптография

Подготовьтесь к коллоквиуму по теме "Криптография". Ответьте на поставленные вопросы.

1) Найдите десятичный эквивалент двоичного числа 01001101:__

2) Найдите двоичный эквивалент числа 100:_____

3) Потоковые шифры могут обрабатывать тексты:
   Посимвольно
   По битового
   По байтово
   По блочно

4) Алгоритмы шифрования бывают:
   Симметричные
   Ассиметричные
   Смешанные

5) При длине ключа N, размер ключевого пространства определяется по формуле:
   N!
   2^N
   2^N+1

6) Алгоритм RSA основан на следующем математическом обосновании:
   - проблема факторизации больших чисел
   - проблема дискретного логарифма
   - нахождение точек на эллиптической кривой

7) В каких алгоритмах шифрования используются более длинные ключи, для обеспечения их одинаковой криптостойкости:
   - в симметричных
   - в ассиметричных

8) Правило Кирхгоффа говорит о:
   - безопасности информационных систем
   - ключевой информации при шифровании
   - наличии слабых мест в информационной системе

9) Ключевой информацией ГОСТа 28147-89 являются
   - заблица замен
   - синхропосылка ГПЧ
   - ключ 256 бит
   - 8 ключей по 256 бит
   - размер регистра сдвига
   - количество проходов основного шага криптопреобразования

10) Синхропосылка это:
- стартовой число ГПЧ
- средство контроля целостности сообщения
- средство подтверждения авторства текста

11) Имитовставка используется для:
- контроля целостности
- проверки авторства
- является элементом цифровой подписи

12) Отметьте свойства хеш-функций, необходимые для ее криптографического использования:
- Однонаправленность
- Сжатие
- Стойкость к коллизиям
- Стойкость к нахождению первого прообраза
- Стойкость к нахождению второго прообраза

13) Сопоставьте режим шифрования и его особенности:

Простая замена		Одинаковые блоки исходного текста дают одинаковые блоки закрытого текста
Гаммирование		Для одинаковых блоков шифруемой информации необходимы различные синхропосылки
Гаммирование с обратной связью		Работает с зацеплением блоков и обеспечивает распространение ошибок

14) Электронная подпись позволяет подтвердить
- авторство сообщения
- целостность сообщения
- наличие защифрованного сообщения

15) Хеш - функции используются для:
- проверки целостности сообщений
- формирования цифровой подписи
- шифрования информации

16) Какие методы могут использоваться для идентификации пользователя:
- биометрические характеристики
- логин и пароль
- ключевая информация на внешнем носителе

17) К видам резервного копирования относятся:
- инкрементное
- полное
- архивное

18) Показателями криптостойкости являются:
- размер ключевого пространства
- среднее время, необходимое для криптоанализа
- время хранения ключевой информации

3. Практическое задание Разработка программного продукта

Практическое задание предполагает разработку программного продукта по заданной теме. Для представления работы к защите необходимо наличие исполняемого модуля, исходных данных и руководства пользователя, описывающего алгоритм работы программы, условия использования, принятые допущения и ограничения (в электронном и бумажном варианте).

В рамках задания предлагается разработать программный продукт, реализующий один из криптографических алгоритмов, использующихся в системах защиты информации.

1. Реализовать алгоритм криптографического закрытия информации методом гаммирования, в качестве гаммы использовать случайную последовательность, получаемую с помощью генератора случайных чисел.
2. Реализовать алгоритм криптографического закрытия информации методом перестановок, ключевая информация должна быть варьируемой пользователем через соответствующий интерфейс.
3. Реализовать алгоритм криптографического закрытия информации методом стеганографии (например, прячем текст в картинке).
4. Реализовать алгоритм криптографического закрытия информации методом "поворотной решетки", размер решетки должен задаваться пользователей (с ограничением по максимальному размеру).
5. Реализовать алгоритм криптографического закрытия информации методом последовательно подстановки и последующей перестановки.
6. Шифрование с открытым ключом. Реализовать алгоритм ассиметричного шифрования RSA (с ограничениями по длине ключа).
7. Шифрование с открытым ключом. Реализовать алгоритм ассиметричного шифрования Диффи – Хеллмана.
8. Для исходного текста произвольной длины вычислить хеш-значение по алгоритму SHA-1(самостоятельно реализовать алгоритм)
9. Для исходного текста произвольной длины вычислить хеш-значения с использованием нескольких хеш-функций.
10. Разработать программный продукт по самостоятельно выбранной теме в области обеспечения безопасности.

Учебно-методические материалы. Основы криптографии

Проблема защиты информации путем ее преобразования, исключающего ее прочтение посторонним лицом, волновала человеческий ум с давних времен. История криптографии -ровесница истории человеческого языка. Более того, первоначально письменность сама по себе была криптографической системой, так как в древних обществах ею владели только избранные. Священные книги Древнего Египта, Древней Индии тому примеры.

С широким распространением письменности криптография стала формироваться как самостоятельная наука. Первые криптосистемы встречаются уже в начале нашей эры. Так. Цезарь в своей переписке использовал уже более менее систематический шифр, получивший его имя.

Бурное развитие криптографические системы получили в годы первой и второй мировых войн (машина Энигма). Начиная с послевоенного времени и по нынешний день, появление вычислительных средств ускорило разработку и совершенствование криптографических методов.

Почему проблема использования криптографических методов в информационных системах (ИС) стала в настоящий момент особо актуальна?

С одной стороны, расширилось использование компьютерных сетей, в частности глобальной сети Интернет, по которым передаются большие объемы информации государственного, военного, коммерческого и частного характера, не допускающего возможность доступа к ней посторонних лиц. С другой стороны, появление новых мощных компьютеров, технологий сетевых и нейронных вычислений сделало возможным дискредитацию криптографических систем еще недавно считавшихся практически не раскрываемыми.

Проблемой защиты информации путем ее преобразования занимается *криптология (kryplos* - тайный, logos - наука). Криптология разделяется на два направления - *криптографию* и *криптоанализ.* Цели этих направлений прямо противоположны.

Криптография занимается поиском и исследованием математических методов преобразования информации.

Сфера интересов *криптоанализа* - исследование возможности расшифровывания информации без знания ключей.

Основные направления использования криптографических методов - передача конфиденциальной информации по каналам связи (например, электронная почта), установление подлинности передаваемых сообщений, хранение информации (документов, баз данных) на носителях в зашифрованном виде.

Аутентификация, целостность и неоспоримость

Помогая сохранить содержание сообщения в тайне, криптография может быть использована, чтобы дополнительно обеспечить решение следующих задач:

Аутентификация. Получателю сообщения требуется убедиться, что оно исходит от конкретного отправителя.

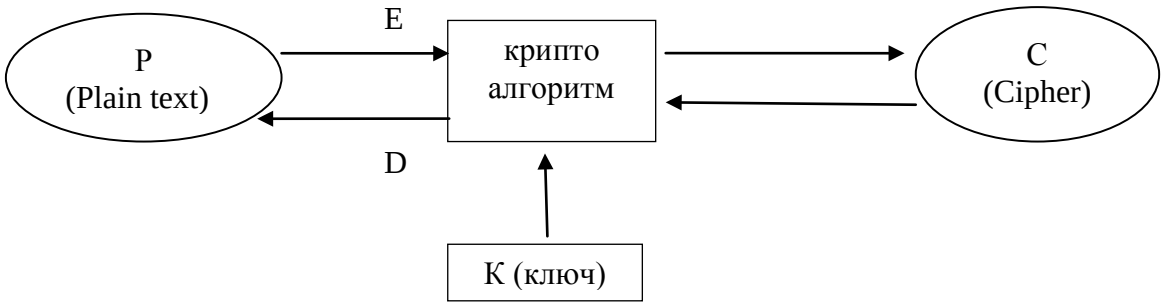
Целостность. Получатель сообщения в состоянии проверить, были ли внесены какие-нибудь изменения в полученное сообщение в ходе его передачи.

Неоспоримость. Отправитель сообщения должен быть лишен возможности впоследствии отрицать авторство сообщения.

Перечисленные задачи часто приходится решать на практике для организации взаимодействия людей при помощи компьютеров и компьютерных сетей. Подобные же задачи возникают и в случае личностного человеческого общения: часто требуется проверить, а действительно ли ваш собеседник тот, за кого он себя выдает, и подлинны ли предъявленные им документы. Будь то паспорт, водительское удостоверение или страховой полис. Вот почему в обыденной жизни не обойтись без аутентификации, проверки целостности и доказательства неоспоримости, а значит и без криптографии.

ТЕРМИНОЛОГИЯ

Шифрование – это преобразовательный процесс, при котором исходный текст (открытый текст) с помощью ключа заменяется шифрованным текстом. Дешифрование – это обратный процесс.



Обозначим открытый текст буквой P (от английского слова plaintext), это может быть текстовый файл, битовое изображение, оцифрованный звук, что угодно. Единственное ограничение связано с тем, что, поскольку предметом изложения является компьютерная криптография, под P понимаются исключительно двоичные данные. Шифртекст обозначается буквой C (от английского слова ciphertext) и также представляет собой двоичные данные. *Ключ* - информация, необходимая для беспрепятственного шифрования и дешифрования текстов. Ключ должен выбираться среди значений, принадлежащих множеству, которое называется ключевым пространством.
Пространство ключей *K* - это набор возможных значений ключа.

Размер ключевого пространства определяется как 2^N, где N –длина ключа в битах.

И функция шифрования E, и функция расшифрования D зависят от ключа.

$E_k(P)=C$

$D_k(C)=P$

Некоторые алгоритмы шифрования используют различные ключи для шифрования и расшифрования. Это означает, что ключ шифрования Кш отличается от ключа расшифрования Кр,

После зашифрования преобразованный открытый текст может быть передан по каналам компьютерной сети или сохранен в памяти компьютера.

Криптографический алгоритм, также называемый *шифром* или *алгоритмом шифрования,* представляет собой математическую функцию, используемую для шифрования и расшифрования. Если быть более точным, таких функций две: одна применяется для шифрования, а другая — для расшифрования.

Когда надежность криптографического алгоритма обеспечивается за счет сохранения в тайне сути самого алгоритма, такой алгоритм шифрования называется *ограниченным*. Ограниченные алгоритмы представляют значительный интерес с точки зрения **истории** криптографии, однако совершенно непригодны при современных требованиях, предъявляемых к шифрованию. Ведь в этом случае каждая группа пользователей, желающих обмениваться секретными сообщениями, должна обзавестись своим оригинальным алгоритмом шифрования.

Все современные криптосистемы построены по **правилу Кирхгоффа**: секретность сообщения определяется только секретностью ключевой информации. Т.е. подразумевается, что сами алгоритмы преобразований всем известны

В качестве информации, подлежащей шифрованию и дешифрованию, будут рассматриваться *тексты,* построенные на некотором *алфавите.* Под этими терминами понимается следующее.

Алфавит - конечное множество используемых для кодирования информации знаков.

Текст - упорядоченный набор из элементов алфавита.

В качестве примеров алфавитов, используемых в современных ИС можно привести следующие:

- алфавит Z_{33} - 32 буквы русского алфавита и пробел;
- алфавит Z_{256} - символы, входящие в стандартные коды ASCII и КОИ-8;
- бинарный алфавит – $Z_2 = \{0,1\}$;
- восьмеричный алфавит или шестнадцатеричный алфавит

Под *криптосистемой* понимается алгоритм шифрования + множество всевозможных ключей +открытых и шифрованных текстов.

Термины *распределение ключей* и *управление ключами* относятся к процессам системы обработки информации, содержанием которых является составление и распределение ключей между пользователями.

Электронной (цифровой) подписью называется присоединяемое к тексту его криптографическое преобразование, которое позволяет при получении текста другим пользователем проверить авторство и подлинность сообщения.

Криптостойкостью называется характеристика шифра, определяющая его стойкость к дешифрованию без знания ключа (т.е. криптоанализу). Имеется несколько показателей криптостойкости, среди которых:

- количество всех возможных ключей (размер ключевого пространства);
- среднее время, необходимое для криптоанализа.

Эффективность шифрования с целью защиты информации зависит от сохранения тайны ключа и криптостойкости шифра.

ТРЕБОВАНИЯ К КРИПТОСИСТЕМАМ

Процесс криптографического преобразования может осуществляться как программно, так и аппаратно. Аппаратная реализация более дорогая, но её отличают высокая

производительность, простота использования, защищенность. Программная реализация более практичная и допускает более гибкое использование.

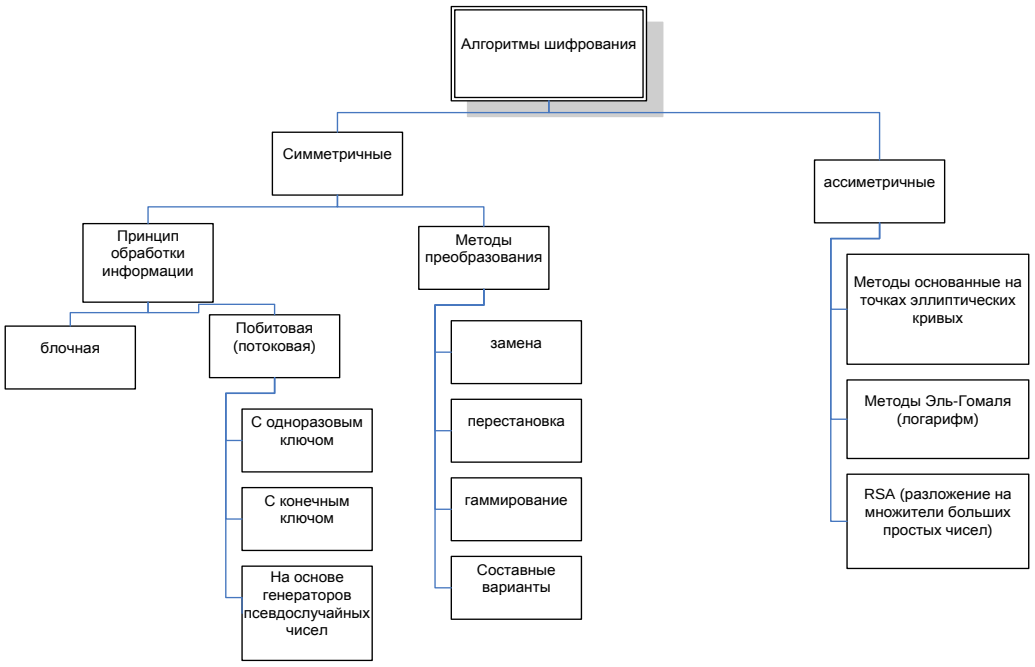
Общепринятые требования к построению криптографических систем:

1. знание алгоритма не должно влиять на надёжность защиты;
2. шифрованное сообщение должно поддаваться чтению только при знании ключа;
3. незначительное изменение ключа должно приводить к значительным изменениям в зашифрованном сообщении;
4. не должно быть простых, легко устанавливаемых зависимостей между ключами, последовательно используемыми в процессе шифрования;
5. число операций необходимых для расшифровывания информации путём перебора всех ключей должно имеет строгую нижнюю оценку, и выходить за пределы возможностей современных компьютеров, даже с использованием сетевых вычислений;
6. структурные элементы алгоритма шифрования должны быть;
7. дополнительные биты, вводимы при шифровании, должны быть скрыты в шифрованном сообщении;
8. алгоритм должен допускать как программную, так и аппаратную реализацию, при этом изменение длины ключа не должно приводить к качественному ухудшению самого алгоритма.

КЛАССИФИКАЦИЯ АЛГОРИТМОВ ШИФРОВАНИЯ

Алгоритмы шифрования

1. Симметричные алгоритмы (с одним ключом) – для шифрования и дешифрования используется один и тоже ключ.
2. Асимметричные алгоритмы – формируется пара открытый и закрытый элементы ключа. Открытый ключ доступен всем абонентам, закрытый – только у владельца.



В *симметричных криптосистемах* и для шифрования, и для дешифрования используется *один и тот же ключ.* Этот ключ должен храниться в тайне и передаваться способом, исключающим его перехват. Надежность одноключевого алгоритма определяется выбором ключа, поскольку его знание дает возможность злоумышленнику без помех расшифровывать все перехваченные сообщения.

Симметричные алгоритмы шифрования бывают двух видов. Одни из них обрабатывают открытый текст побитно. Они называются *потоковыми алгоритмами,* или *потоковыми шифрами, при этом* каждый бит исходной информации шифруется независимо от других с помощью гаммирования.

Согласно другим, открытый текст разбивается на блоки состоящие из нескольких бит. Такие алгоритмы называются *блочными,* или *блочными шифрами.* В современных компьютерных Алгоритмах блочного шифрования обычно длина блока составляет не менее 64 бита.

Симметричные алгоритмы используют различные методы преобразования данных.

Шифры замены

Шифром замены называется алгоритм шифрования, который производит замену буквы открытого текста на какой-то символ шифрованного текста. Получатель сообщения расшифровывает его путем обратной замены.

В классической криптографии различают 4 разновидности шифров замены:

- *Простая замена,* или *одно-алфавитный шифр.* Каждая буква открытого текста заменяется на один и тот же символ шифртекста. (шифр Цезаря – циклический сдвиг букв алфавита)

  А → Г
  Б → Д
  В → Е
  .....

- *Блочная замена.* Шифрование открытого текста производится блоками. Например, блоку "АБА" может соответствовать "РТК", а блоку "АББ" -"СЛЛ".
- Многоалфавитная замена. Состоит из нескольких шифров простой замены. Например, могут использоваться пять шифров простой замены, а какой из них конкретно применяется для шифрования данной буквы от' крытого текста, — зависит от ее положения в тексте.

Разновидностью шифра замены можно считать код, при этом заменяют не буквы, а слова или фразы, при этом закодировать можно только те слова, которые заранее предусмотрены. (Пример: «ракета» будет «ласточкой», а старт «гнездованием» -> обычно гнездование ласточек начинается в первых числах апреля)

Шифры перестановки – меняется порядок следования букв в тексте

Пример. Записываем шифруемый текст столбиками в матрицу и считываем по строками. При этом ключом (секретной информацией) является размер матрицы.

Л О П Е А
А Ч Р Т
С К И Е
Т А Л Л
↓
ЛОПЕААЧРТСКИЕТАЛЛ

Гаммирование – метод заключается в наложении на исходный текст некоторой псевдослучайной последовательности 0 и 1; которая может генерироваться на основе ключа по определённому правилу, с помощью генератора псевдослучайных чисел или выбирать прозвольно. Чаще всего используется наложение по правилу «исключающего или» или сложение по модулю 2.

Таблица истинности « исключающее или» (XOR).

	0	**1**
0	0	1
1	1	0

Пример:

⊕ 1 1 0 0 1 0 1 0 – текст
1 1 0 1 1 1 0 1 - гамма (ключ)
---
0 0 0 1 0 1 1 1 –шифрованный текст

Гаммирование

В потоковых алгоритмах шифрования, т. е. при шифровании потока данных, каждый бит исходной информации шифруется независимо от других с помощью гаммирования.

Гаммирование - наложение на открытые данные гаммы шифра (случайной или псевдослучайной последовательности единиц и нулей) по определенному правилу. Обычно используется "исключающее ИЛИ", называемое также сложением по модулю 2. Для расшифровывания та же гамма накладывается на зашифрованные данные еще раз.

Гаммирование

- С одноразовым ключом
  При однократном использовании случайной гаммы одного размера с шифруемыми данными взлом кода невозможен (так называемые криптосистемы с одноразовым или бесконечным ключом). В данном случае "бесконечный" означает, что гамма не повторяется.
- С конечным ключом - ключ короче сообщения
- На основе генератора ПСЧ

Понятно, что обмен ключами размером с шифруемую информацию не всегда уместен. Поэтому чаще используют гамму, получаемую с помощью генератора псевдослучайных чисел (ПСЧ). В этом случае ключом является порождающее число генератора (начальное значение, вектор инициализации).

Каждый генератор ПСЧ имеет период $m=2^b$, b- длина ПСЧ в битах, после которого генерируемая последовательность повторяется. Очевидно, что период псевдослучайной гаммы должен превышать длину шифруемой информации.

Генератор ПСЧ

Пусть имеется конгруэнтный генератор псевдослучайных чисел, работающий по некоторому определённому алгоритму. Часто используют следующий алгоритм:

$T_{i+1} := (a*T_i+b) \bmod m$

где T_i- предыдущее, а T_{i+1} - следующее псевдослучайное число,
коэффициенты *a,b,m* постоянны и хорошо известны.

0бычно $m=2^n$, где *n* - разрядность процессора, *a mod 4=1,* а *b* - нечётное. В этом случае последовательность псевдослучайных чисел имеет период m.

Процесс шифрования осуществляется следующим образом: шифруемое сообщение представляется в виде последовательности слов S0,S1... каждое длины *n,* которые складываются по модулю 2 со словами последовательности To, T1, ..., то есть $C_i := S_i \oplus T_i$.
Последовательность To,T1, ..., называется гаммой шифра, ⊕ - логическая операция XOR.

Процесс расшифровывания заключается в том, чтобы ещё раз сложить шифрованную последовательность с той же гаммой шифра: $S_i := C_i \oplus T_i$

Ключом шифра является начальное значение *To,* которое является секретным и должно быть известно только отправителю и получателю шифрованного сообщения.

Если период последовательности псевдослучайных чисел достаточно велик, чтобы гамма шифра была длиннее сообщения, то дешифровать сообщение можно только подбором ключа. При увеличении *n*- разрядности процессора экспоненциально увеличивается криптостойкость шифра.

Данный метод шифрования обладает существенным недостатком. Если известна хотя бы часть исходного сообщения, то всё сообщение может быть легко дешифровано: $T_i := C_i \oplus S_i$.

АССИМЕТРИЧНЫЕ АЛГОРИТМЫ

Всем системам симметричного шифрования присущи следующие основные недостатки.

- принципиальной является надежность канала передачи ключа второму участнику секретных переговоров. Иначе говоря, ключ должен передаваться по секретному каналу.
- к службе генерации ключей предъявляются повышенные требования, обусловленные тем, что для N абонентов при схеме взаимодействия "каждый с каждым" требуется n(n-1)/2 ключей, то есть зависимость числа ключей от числа абонентов является квадратичной.

Для решения вышеперечисленных проблем симметричного шифрования предназначены системы асимметричного шифрования, или шифрование с открытым ключом, которые используют свойства функций с секретом, разработанных Диффи и Хеллманом.

Эти системы характеризуются наличием у каждого абонента двух ключей; открытого и закрытого (секретного). При этом открытый ключ передается всем участникам секретных переговоров. Таким образом, решаются обе проблемы.

Ассиметричные алгоритмы	Особенности построения алгоритма
RSA	Стойкость зависит от сложности факторизации больших целых чисел.
ECC (криптосистема на основе эллиптических кривых)	Использует алгебраическую систему, которая описывается в терминах точек эллиптических кривых.
Эль-Гамаль	Вариант Диффи-Хеллмана, который может быть использован как для шифрования, так и для электронной подписи. В основе - Проблема дискретного логарифма*
DSS (Digital Signature Standard) [≅DSA-Digital Signature Algorithm – Алгоритм Цифровой подписи)]	Стандарт одобрен правительством США. Длина ключа варьируется в пределах от 512 до 1024 бит. DSS предназначен для создания цифровой подписи, но не для закрытия информации. В стандарте DSS найдены некоторые слабые места защиты, вследствие чего он не так широко распространен. В основе лежит - Проблема дискретного логарифма

* Проблема Дискретного Логарифма (Discrete Logarithm Problem – DLP) кратко формулируется так: по заданному простому числу p, основанию g и значению $g_x \pmod p$ найти значение x", причем проблема может быть сформулирована в ограниченной области.

RSA

Первым шифром, разработанным на принципах асимметричного шифрования в 1978 году, является шифр RSA. Шифра RSA назван так по первым буквам фамилий его изобретателей: Рона Райвеста, Ади Шамира и Леонарда Элдемана - основателей компании RSA Data Secutity. RSA - не только самый популярный из асимметричных шифров, но и самый известный шифр.

Математическое обоснование RSA таково: поиск делителей очень большого натурального числа, являющегося произведением двух простых, - крайне трудоемкая процедура. По открытому ключу очень сложно вычислить парный ему личный ключ. Шифр RSA всесторонне изучен и признан стойким при длине ключей не менее 1024 бит.
Предполагается, что с ростом мощности процессоров RSA потеряет стойкость к атаке полным перебором. Однако же увеличение мощности процессоров позволит применить более длинные ключи, что повысит стойкость шифра.

Рассмотрим принцип построения алгоритма

	Алгоритм RSA	пример
1	Случайно выбирается 2 **простых***, очень больших (250-300 десятичных разрядов) числа p и q	p=3, q=11
2	Вычисляются два произведения n=pq, m=(p-l)(q-l)	n=33, m=20
3	Выбирается небольшое целое нечетное E, взаимно простое (не имеющее общих сомножителей) с m	E=7
4	Находится D , такое что (DE)(mod m)=1	(D*7)(mod 20)=1 $\Rightarrow$ D=3
5	Отправитель шифрует сообщение, при необходимости разбивая его на блоки X длиной не более n $C_i = X_i^e \bmod n$	Шифруем сообщение «CAB» с помощью отображения, где A-1, B-2, C-3 $\Rightarrow$ получим (3,1,2) Зашифровываем 1. $3^7 \bmod 33 = 2187 \bmod 33 = 9$ 2. $1^7 \bmod 33 = 1 \bmod 33 = 1$ 3. $2^7 \bmod 33 = 128 \bmod 33 = 29$ $\Rightarrow$ (9,1,29)
6	Расшифровывание происходит по формуле $P_i = C_i^d \bmod n$	(9,1,29) Расшифровываем 1. $9^3 \bmod 33 = 729 \bmod 33 = 3$ 2. $1^3 \bmod 33 = 1 \bmod 33 = 1$ 3. $29^3 \bmod 33 = 24389 \bmod 33 = 2$ $\Rightarrow$ (3,1,2)
	E, n - открытый ключ; D - закрытый.	

* Малая теорема Ферма позволяет определить является ли число простым или составным: для простого числа P и любого целого K, при K<P, справедливо тождество: $K^{P-1} \bmod P = 1$
Пример: P=7, K=2 $\Rightarrow 2^6 \bmod 7 = 1$

Теоретически, зная открытый ключ, можно вычислить значение закрытого ключа, однако необходимым промежуточным действием этого преобразования является нахождение сомножителей p и q, для чего нужно разложить n на сомножители, - эта процедура занимает очень много времени. Именно с вычислительной сложностью этого разложения связана криптостойкость RSA.

Алгоритм Диффи – Хеллмана

Предположим, что всем адресатам известны некоторые два числа g и p, которые не являются секретными и могут быть известны также другим заинтересованным лицам. Адресат, желающий отправить подписанное сообщение, выбирает секретный ключ a (обычно это большие случайные числа) и генерирует открытый ключ, по следующей формуле:

$$A = g^a \bmod p$$

Адресат, желающий получить сообщение, аналогично генерирует свой открытый ключ:

$$B = g^b \bmod p,$$

где b – его секретный ключ. Все открытые ключи находятся в общем доступе. На втором этапе отправитель, зная открытый ключ, генерирует ключ, использующийся для шифровки сообщения:

$$K^* = B^a \bmod p$$

Получатель, зная открытый ключ отправителя, генерирует ключ, использующийся для расшифровки сообщения:

$$K^{**} = A^b \bmod p$$

Очевидно, что $K^{**} = K^* = g^{a \cdot b} \bmod p = K$. Таким образом, и отправитель, и получатель имеет общий секретный ключ K, пригодный для выполнения операций шифрования/дешифрования. Для злоумышленника крайне сложно будет за разумное время определить значение $g^{a \cdot b} \bmod p$ по известным ему $g^a \bmod p$ и $g^b \bmod p$. Иллюстрация данного алгоритма представлена на рисунке.

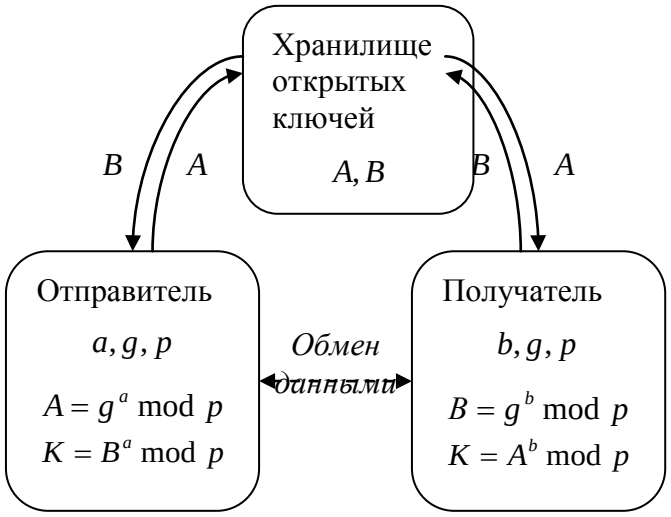


Рисунок – Иллюстрация алгоритма алгоритма Диффи – Хеллмана

ХЕШИРОВАНИЕ

Под **хэш- функцией** понимается процедура получения контрольной характеристики двоичной последовательности, основанная на контрольном суммировании и криптографических преобразованиях.

Отечественный ГОСТ Р 34.11-94.

Хэш- функция применяемая к исходным данным в результате даёт значение, состоящее из фиксированного небольшого числа бит, которое иногда называется **digest**. Digest передаётся вместе с сообщением. Получатель, зная какая хэш-функция была использована, заново вычисляет digest и сравнивает значения, если значения полученного по почте и рассчитанного совпадают, значит, сообщение по дороге никаким изменениям не подвергалось.

Чаще всего хэш -функции является односторонними функциями (Пример с разбитой вазой: т. е из состояния А легко в В, а из В в А не возможно) , однако бывают одностороннии функции с лазейкой (можно собрать, если знаешь что).

Построение Хэш функции - задача математически сложная. Хэш-функция должна удовлетворять ряду требований:

1. По дайджесту должно быть невозможным вычислить исходное сообщение.
2. Должна отсутствовать возможность существования двух разных сообщений, для которых могут быть получены одинаковые дайджесты.

Основные требования к функциям хэширования

1. Сжатие. Функция отображает входное сообщение x произвольной конечной длины в хэш- значение: $y=h(x)$ небольшой фиксированной длины. При этом исходное сообщение называется прообразом.
2. Простота вычисления. Для заданной функции h и сообщения x, $h(x)$ вычисляется не выше, чем с полиномиальной сложностью.

1 и 2 - базовые минимальные требования. Для хэш-функция, используемых в криптографии необходимо выполнение дополнительных требований.

3. Стойкость к вычислению прообраза - невозможность нахождения неизвестного прообраза для любых предварительно заданных хэш значений, то есть для заданной функции h вычислительно невозможно найти прообраз x при известном хеш значении $y=h(x)$для любого y.
4. Стойкость к вычислению второго прообраза- невозможность нахождения любого другого прообраза, который давал бы такое же хэш- значение, как и заданный, то есть для заданной функции h и прообраза x вычислительно невозможно найти другой прообраз $x' \neq x$, для которого выполнялось бы условие: $h(x)=h(x')$
5. Стойкость к коллизиям – невозможность нахождения двух прообразов, для которых вырабатывалось бы одинаковое значение функции h.

Все существующие функции Хэш можно разделить на:

- бесключевые Хэш фунции, зависящие только от сообщения,
- Хэш функции с секретным ключом зависящие как от сообщения так и от секретного ключа.

Однонаправленная hash-функция- функция удовлетворяющая требованиям: 1)2)3)4).
Бесколлизионной hash-функцией –функция удовлетворяющая требованиям 1)-5).

ЦИФРОВАЯ ПОДПИСЬ (DIGITAL SIGNATURE)

Способ проверки целостности содержимого сообщения и подлинности отправителя. Реализуется с помощью ассиметричных шифров и Хэш функций.

Цифровая подпись основывается на обратимости ассиметричных шифров, а так же на взаимосвязи содержимого сообщения, самой подписи и пары ключей. Изменение одного из этих параметров сделает невозможным подтверждение подлинности подписи.

Наиболее известные алгоритмы: DSS, RSA.

Secure Hash Algorithm 1 (SHA-1)

Известным алгоритмом криптографического хеширования является *Secure Hash Algorithm 1* (SHA-1). Для входного сообщения произвольной длины до $2^{64}-1$ бит алгоритм генерирует 160-битное хеш-значение. SHA-1 реализует хеш-функцию, построенную на идее функции сжатия. Входами функции сжатия являются блок сообщения длиной 512 бит и выход предыдущего блока сообщения. Выход представляет собой значение всех хеш-блоков до этого момента. Далее приведено описание алгоритма.

Исходное сообщение разбивается на блоки по 512 бит в каждом. Последний блок дополняется до длины, кратной 512 бит. Сначала добавляется 1, а потом нули, чтобы длина блока стала равной (512 - 64 = 448) бит. В оставшиеся 64 бита записывается длина исходного сообщения в битах. Если последний блок имеет длину более 448, но менее 512 бит, то дополнение выполняется следующим образом: сначала добавляется 1, затем нули вплоть до конца 512-битного блока; после этого создается ещё один 512-битный блок, который заполняется вплоть до 448 бит нулями, после чего в оставшиеся 64 бита записывается длина исходного сообщения в битах. Дополнение последнего блока осуществляется всегда, даже если сообщение уже имеет нужную длину. После этого инициализируются пять 32-битных переменных:

$$\begin{cases} A = a = 0x67452301 \\ B = b = 0xEFCDAB89 \\ C = c = 0x98BADCFE \\ D = d = 0x10325476 \\ E = e = 0xC3D2E1F0 \end{cases} \quad (7)$$

Далее определяются четыре нелинейные операции и четыре константы:

$$\begin{cases} F_t(B,C,D) = (B \wedge C) \vee (\neg B \wedge D),\ K_t = 0x5A827999 \quad для \quad 0 \le t \le 19 \\ F_t(B,C,D) = B \oplus C \oplus D,\ K_t = 0x6ED9EBA1 \quad для \quad 20 \le t \le 39 \\ F_t(B,C,D) = (B \wedge C) \vee (B \wedge D) \vee (C \wedge D),\ K_t = 0x8F1BBCDC \quad для \quad 40 \le t \le 59 \\ F_t(B,C,D) = B \oplus C \oplus D,\ K_t = 0xCA62C1D6 \quad для \quad 60 \le t \le 79 \end{cases} \quad (8)$$

Главный цикл итеративно обрабатывает каждый 512-битный блок. Итерация состоит из четырех этапов по двадцать операций в каждом. Блок сообщения преобразуется из 16 32-битовых слов M_i в 80 32-битовых слов W_j по следующему правилу:

$$\begin{cases} W_t = M_t \quad для \quad 0 \le t \le 15 \\ W_t = (W_{t-3} \oplus W_{t-8} \oplus W_{t-14} \oplus W_{t-16}) <<< 1_t \quad для \quad 16 \le t \le 79 \end{cases} \tag{9}$$

Таким образом, на каждом шаге цикла для $0 \le t \le 79$ производятся следующие вычисления:

$$\begin{cases} temp = (a <<< 5) + F_t(B,C,D) + e + W_t + K_t \\ e = d \\ d = c \\ c = b <<< 30 \\ b = a \\ a = temp \end{cases} \tag{10}$$

Литература и источники информации

Рекомендуемые источники информации для выполнения 2 и 3 модулей задания

1. Баричев С.В. Криптография без секретов. –М.: Наука, 1998, с.120
2. Б.Шнайер. Прикладная криптография
3. Калинцев Ю.К. Криптозащита сообщений в системах связи. Уч.пособие. – М.: МТУСИ, 2000г., 336с.
4. В. В. Ященко — Введение в криптографию
5. Зегжда Д.П., Ивашко А.М. Основы безопасности информационных систем.-М.: Горячая линия – Телеком, 2000. 452с.
6. Защита информации в компьютерных сетях. Практический: учебное пособие / А. Н. Андрончик, В. В. Богданов, Н. А. Домуховский, А. С. Коллеров, Н. И. Синадский, Д. А. Хорьков, М. Ю. Щербаков; под ред. Н. И. Синадского. Екатеринбург : УГТУ-УПИ, 2008. 248 с
7. Организационно-правовое обеспечение информационной безопасности: Учебное пособие для вузов / под ред. Стрельцова А.А. Изд-во "Академия", 2008
8. Мельников, В. П. Информационная безопасность и защита информации : учеб. пособие для вузов / В. П. Мельников, С. А. Клейменов, А. М. Петраков; под ред. С. А. Клейменова. – 5-е изд., стер. – М. : Академия, 2011. – 331 с.
9. Норенков, И. П. Автоматизированные информационные системы : учеб. пособие / И. П. Норенков. – М. : Изд-во МГТУ им. Н.Э. Баумана, 2011. – 342 с.
10. Методы и средства защиты компьютерной информации. Межсетевое экранирование : учеб. пособие / В.А. Мулюха [и др.] – СПб. : Изд-во Политехн. Ун-та, 2010. – 90 с.
11. Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации»
12. ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритмы криптографического преобразования.
13. ГОСТ 34.10-2012. Информационная технология. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма.
14. ГОСТ Р ИСО/МЭК 27001. Информационная технология. Методы и средства обеспечения безопасности. Системы менеджмента информационной безопасности.