Лекция 14. Криптография и стеганография

1. Функции хеширования.
2. Принципы использования криптографического интерфейса ОС Windows.
3. Компьютерная стеганография и ее применение.

Хеширование

Процесс преобразования исходного текста M произвольной длины в хеш-значение (хеш-код, дайджест, образ или просто хеш) H(M) фиксированной длины.

Требования к функциям хеширования

- постоянство длины хеш-значения независимо от длины исходного текста

$\forall M\ \mathrm{Length}[H(M)]=\mathrm{const}$

- полная определенность (для двух одинаковых исходных текстов должно получаться одно и то же хеш-значение)

$\forall M_1=M_2\ H(M_1)=H(M_2)$

- необратимость (невозможность восстановления исходного текста по его хеш-значению)

$\neg \exists H^{-1}\ H^{-1}(M)=M$

- стойкость к «взлому» (практическая невозможность подобрать другой исходный текст для известного хеш-значения)

$\neg \exists M' \neq M\ H(M')=H(M)$

Применение хеширования при защите информации

- Хранение многоразовых паролей пользователей компьютерных систем.
- Генерация одноразовых паролей и откликов на случайные запросы службы аутентификации (протоколы S/Key, CHAP).
- Генерация сеансовых ключей из паролей.
- При вычислении и проверке ЭЦП.
- Для обеспечении целостности информации (конструкция $HMAC_K(M)=H[(K \oplus opad) \,||\, H[(K \oplus ipad) \,||\, M]]$, где K – секретный ключ, ipad и opad – константы).

ЭП и функции хеширования

На функции хеширования, используемые в системах ЭП, налагаются дополнительные условия:

- чувствительность к любым изменениям в документе (вставкам, удалением, перестановкам, заменам фрагментов и отдельных символов);
- минимальность вероятности того, что хеш-значения двух разных документов, независимо от их длин, совпадут.

Способы построения функций хеширования

- На основе односторонней функции f:

$M=M_1M_2 \ldots M_i \ldots M_n$

$H_i=f(M_i, H_{i-1})$ (H_0 – константа)

$H(M)=H_n$

- На основе функции блочного шифрования E:

$M=M_1M_2 \ldots M_i \ldots M_n$

$H_i=E_{Mi}(H_{i-1})$ (H_0 – константа)

$H(M)=H_n$

Функции хеширования

- MD2, MD4, MD5 (Message Digest) – получают хеш-значение длиной 128 бит и используются в системе ЭП RSA;
- SHA (Secure Hash Algorithm) – получает хеш-значение длиной 160 (192, 256, 384 или 512) бит и используется в системе ЭП DSS;

Функции хеширования

- ГОСТ Р 34.11-2012 – получает хеш-значение длиной 256 или 512 бит и используется в российских стандартах ЭП;
- RIPEMD (Race Integrity Primitives Evaluation Message Digest) – получает хеш-значение длиной 128 или 160 бит (две модификации).

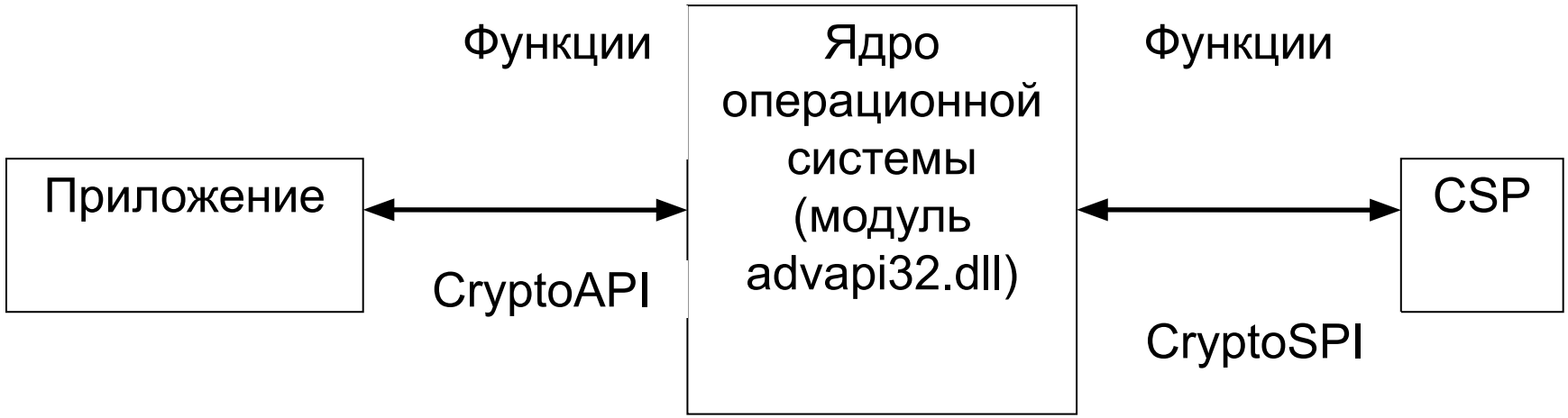
Преимущества использования криптографических библиотек

- Меньше риск создания уязвимостей в системах защиты информации за счет уменьшения вероятности внесения ошибок в программные реализации даже стойких криптографических алгоритмов.
- Отсутствие необходимости внесения изменений в прикладные программы при замене одной криптографической библиотеки другой.

Криптографический интерфейс приложений ОС Windows (CryptoAPI)

- Набор констант, типов данных и функций, предназначенных для выполнения операций шифрования, расшифрования, получения и проверки ЭП, генерации, хранения и распределения ключей шифрования.
- Эти услуги для приложений предоставляют криптопровайдеры (Cryptographic Service Provider, CSP) – динамически компонуемые библиотеки (DLL), экспортирующие единый набор объектов, определяемый интерфейсом CryptoAPI.

Архитектура криптографической подсистемы Windows



Принципы взаимодействия между приложением и CSP

1. приложение не имеет прямого доступа к изготовлению и хранению ключей шифрования (нет риска их потери из-за ошибок в приложении);
2. приложение не определяет деталей выполнения криптографических операций, а лишь указывает на требуемые от CSP действия (например, зашифровать по заданному алгоритму данные и получить для них ЭП);
3. приложение не обрабатывает данных, по которым проводится аутентификация пользователя (владельца секретных ключей), а предоставляет это CSP.

Криптопровайдер

Характеризуется своим присвоенным производителем именем (строкой символов) и типом (именованной целочисленной константой), определяющим поддерживаемые этим провайдером криптографические алгоритмы и их характеристики (атрибуты криптопровайдера).

Основные атрибуты CSP

- обязательно поддерживаемый алгоритм ЭП (всегда единственный);
- длина ключей ЭП;
- формат ЭП;
- форматы блобов, в которых открытый и закрытый ключи асимметричного шифрования экспортируются из CSP (с возможностью его последующего импорта в CSP);
- поддерживаемые функции хеширования.

Дополнительные атрибуты CSP

- возможно поддерживаемый алгоритм обмена сеансовыми ключами (всегда единственный);
- возможно поддерживаемые алгоритмы симметричного шифрования;
- схема генерации сеансового ключа из хеш-значения парольной фразы;
- длины сеансовых ключей;
- формат блоба сеансового ключа при его экспорте из CSP (с возможностью его последующего импорта в CSP);
- режимы симметричного шифрования, принятые по умолчанию (например, режим CBC).

Экспорт и импорт ключей

- Закрытый ключ в блобе зашифрован симметричным алгоритмом на ключе, выводимым из парольной фразы.
- Сеансовый ключ в блобе зашифрован асимметричным алгоритмом на открытом ключе получателя (владельца) зашифрованного этим сеансовым ключом сообщения (для расшифрования сеансового ключа потребуется доступ к закрытому ключу получателя или владельца).

Контейнеры ключей

- Для каждого зарегистрированного у него пользователя или конкретного приложения CSP хранит контейнер ключей асимметричного шифрования, который может включать в себя две пары ключей – открытый и секретный ключи для обмена сеансовыми ключами, а также открытый и секретный ключи для ЭП.
- Ключи симметричного шифрования (сеансовые ключи) не сохраняются CSP и об их сохранении (или правильной повторной генерации) должно позаботиться приложение.

Хранение контейнеров ключей

- На жестком диске компьютера (например, в разделе реестра HKEY_CURRENT_USER, где хранят ключи криптопровайдеры, распространяемые вместе с ОС Windows).
- На защищенном от несанкционированного доступа устройстве (например, смарт-карте), подключаемым к компьютеру при выполнении криптографических операций.

Доступ к контейнеру ключей из прикладной программы

- Создание нового контейнера ключей (или открытие существующего) и получение его дескриптора выполняются с помощью функции CryptoAPI CryptAcquireContext, которая должна вызываться в программе до любой из других функций CryptoAPI.

Версии CryptoAPI

- 1.0 – содержит базовый набор функций для выполнения всех необходимых криптографических операций.
- 2.0 – содержит дополнительные функции для работы с сертификатами и поддержки инфраструктуры открытых ключей (требуется подключение библиотеки crypt32.dll).
- CAPICOM – содержит набор многокомпонентных объектов для выполнения криптографических операций в сценариях и апплетах (требуется библиотека capicom.dll).

Примеры использования CryptoAPI

- Шифрующая файловая система Windows (EFS). Используются экспорт и импорт случайного сеансового ключа шифрования файла.
- Пакет программ Microsoft Office. Используется генерация сеансового ключа шифрования документа из парольной фразы.

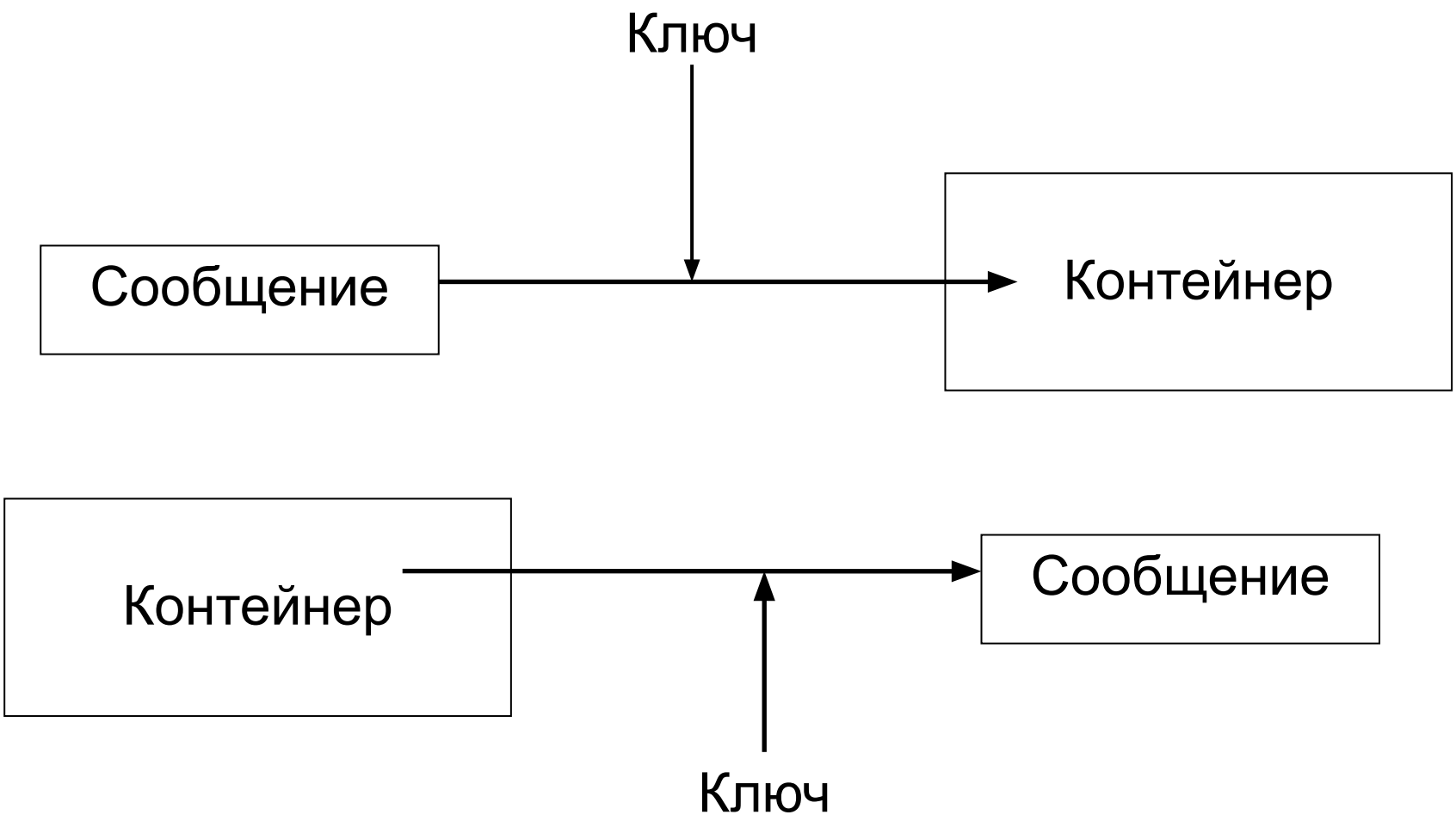
Криптография и стеганография

- Применение методов криптографии позволяет скрыть от непосвященных содержание конфиденциальной информации, но не способно скрыть самого факта ее наличия или передачи.
- Методы *стеганографии* направлены на скрытие самого присутствия конфиденциальной информации.

Основные понятия стеганографии

- Применительно к стеганографии различают *сообщение* (объект, существование и содержание которого должно быть скрыто) и *контейнер* (объект, в котором скрывается сообщение).
- При помещении сообщения в контейнер может использоваться секретный ключ, определяющий порядок помещения сообщения в контейнер. Этот же ключ должен быть задан при извлечении сообщения из контейнера

Скрытие и извлечение сообщения



Принципы компьютерной стеганографии

- обеспечение аутентичности и целостности файла-сообщения;
- открытость методов компьютерной стеганографии;
- сохранение основных свойств файла-контейнера после помещения в него сообщения (после этого файл-контейнер можно открывать, сжимать, восстанавливать без потери качества и изменения содержания информации в контейнере);
- сложность извлечения сообщения из файла контейнера при известности факта скрытия сообщения, но без знания ключа.

Криптография и стеганография

Возможно объединение методов криптографии и стеганографии, при котором сообщение предварительно зашифровывается перед помещением в контейнер.

Применение компьютерной стеганографии

- защита от несанкционированного доступа к конфиденциальной информации;
- преодоление систем сетевого мониторинга и управления сетевыми ресурсами (например, систем промышленного шпионажа, регистрирующих частоту обмена конфиденциальными сообщениями даже при отсутствии возможности их расшифрования);

Применение компьютерной стеганографии

- камуфлирование конфиденциального программного обеспечения (защита от его использования незарегистрированными пользователями путем его скрытия в мультимедийных файлах);
- защита авторских прав создателей (владельцев) электронных документов путем нанесения на файлы с этими документами (фото, аудио и видеоматериалами) специальной метки («водяного знака»), распознаваемого только специальным программным обеспечением.

Методы компьютерной стеганографии

- методы, использующие специальные свойства форматов электронных документов;
- методы, использующие естественную избыточность оцифрованных графических изображений, звука и видеоинформации.

Использование свойств компьютерных форматов

- зарезервированных для дальнейшего применения полей;
- специального форматирования текстовых документов;
- неиспользуемых мест дисковой памяти (например, последних байт и секторов последнего выделенного файлу кластера);
- имитирующих функций для генерации осмысленного текста файла-контейнера, скрывающего сообщения и др.

Недостаток перечисленных методов

Небольшой размер сообщения, которое может быть скрыто в контейнере.

Использование естественной избыточности цифровой мультимедиа информации

Метод последнего значащего бита (Last Significant Bit, LSB). Например, полноцветные графические файлы в формате RGB кодируют каждую точку (пиксель) изображения тремя байтами для представления соответственно красной, зеленой и синей составляющих. Изменение каждого из трех младших битов (для хранения битов скрываемого сообщения) приведет к изменению цветовых характеристик данной точки изображения менее чем на 1%, что абсолютно незаметно для человеческого глаза.

Метод LSB

- Этот метод позволяет скрыть в графическом файле размером 800 килобайт сообщение размером до 100 килобайт.
- Одна секунда оцифрованного звука с частотой дискретизации 44100 герц и уровнем отсчета 8 бит в стереорежиме позволяет скрыть сообщение размером до 10 килобайт.