



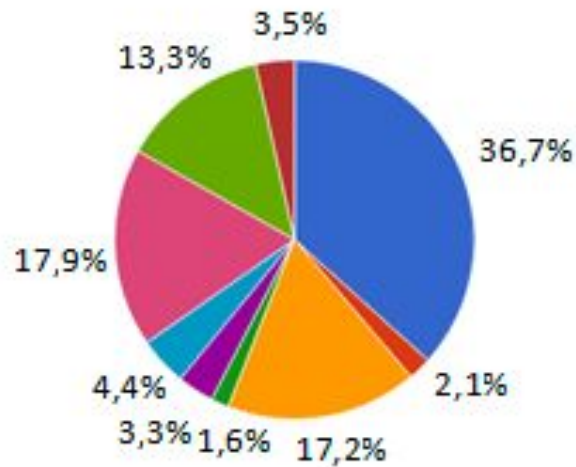
**«ПРОТОКОЛИРОВАНИЕ И АУДИТ,
ШИФРОВАНИЕ, КОНТРОЛЬ
ЦЕЛОСТНОСТИ»**

Каналы утечки информации

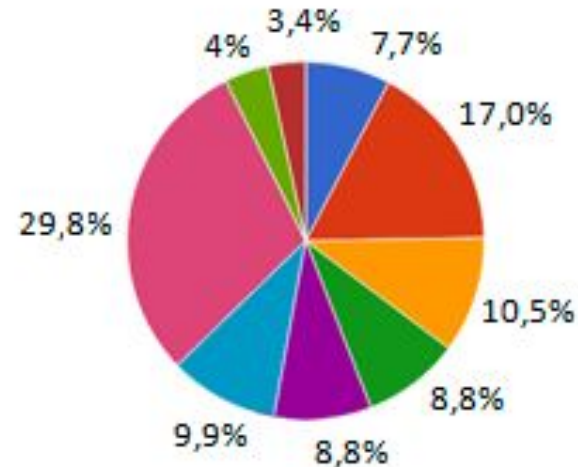


Каналы утечки информации

Умышленные



Случайные



- | | | |
|---------------------|-------------------|--------------------------|
| Не определено | Съемные носители | Бумажные документы |
| Ноутбуки, смартфоны | Веб, интранет | Носители резервных копий |
| ПК, серверы | Электронная почта | Другие |

Сервисы безопасности

- идентификация и аутентификация;
- управление доступом;
- протоколирование и аудит;
- шифрование;
- контроль целостности;
- экранирование;
- анализ защищенности;
- обеспечение отказоустойчивости;
- обеспечение безопасного восстановления;
- туннелирование;
- управление.

Протоколирование и аудит.

Основные понятия.

Под **протоколированием** понимается сбор и накопление информации о **событиях**, происходящих в информационной системе. У каждого **сервиса** свой набор возможных событий, но в любом случае их можно разделить на внешние (вызванные действиями других сервисов), внутренние (вызванные действиями самого сервиса) и клиентские (вызванные действиями пользователей и администраторов).

Аудит – это анализ накопленной информации, проводимый оперативно, в реальном времени или периодически (например, раз в день). Оперативный аудит с автоматическим реагированием на выявленные нештатные ситуации называется активным.

Реализация протоколирования и аудита решает следующие задачи:

- обеспечение **подотчетности** пользователей и администраторов;
- обеспечение возможности реконструкции последовательности событий;
- обнаружение попыток нарушений информационной безопасности;
- предоставление информации для выявления и анализа проблем.

Что необходимо протоколировать

следующие события:

- вход в систему (успешный или нет);
- выход из системы;
- обращение к удаленной системе;
- операции с файлами (открыть, закрыть, переименовать, удалить);
- смена привилегий или иных атрибутов безопасности (режима доступа, уровня благонадежности пользователя и т.п.).

При протоколировании события рекомендуется записывать, по крайней мере, следующую информацию:

- дата и время события;
- уникальный идентификатор пользователя – инициатора действия;
- тип события;

Активный аудит. Основные понятия

Под **подозрительной активностью** понимается поведение пользователя или компонента информационной системы, являющееся злоумышленным (в соответствии с заранее определенной политикой безопасности) или нетипичным (согласно принятым критериям).

Задача **активного аудита** – оперативно выявлять подозрительную активность и предоставлять средства для автоматического реагирования на нее.

Применительно к средствам активного аудита различают **ошибки**:

первого рода : пропуск атак

второго рода: ложные тревоги.

Нежелательность ошибок первого рода очевидна; ошибки второго рода не менее неприятны, поскольку отвлекают администратора безопасности от действительно важных дел, косвенно способствуя пропуску атак.

Функциональные компоненты и архитектура

В составе средств активного аудита можно выделить следующие функциональные компоненты:

- компоненты генерации регистрационной информации. Они находятся на стыке между средствами активного аудита и контролируемыми объектами;

- компоненты хранения сгенерированной регистрационной информации;

компоненты извлечения регистрационной информации (**сенсоры**).

Система архитектуры

- Средства активного аудита строятся в архитектуре менеджер/агент. Основными агентскими компонентами являются сенсоры. Анализ, принятие решений – функции менеджеров. Очевидно, между менеджерами и агентами должны быть сформированы доверенные каналы.

Шифрование

- **Процедура шифрования** – преобразование открытого текста сообщения в закрытый.
- Современные средства шифрования используют известные алгоритмы шифрования. Для обеспечения конфиденциальности преобразованного сообщения используются специальные параметры преобразования – ключи.

Шифрование

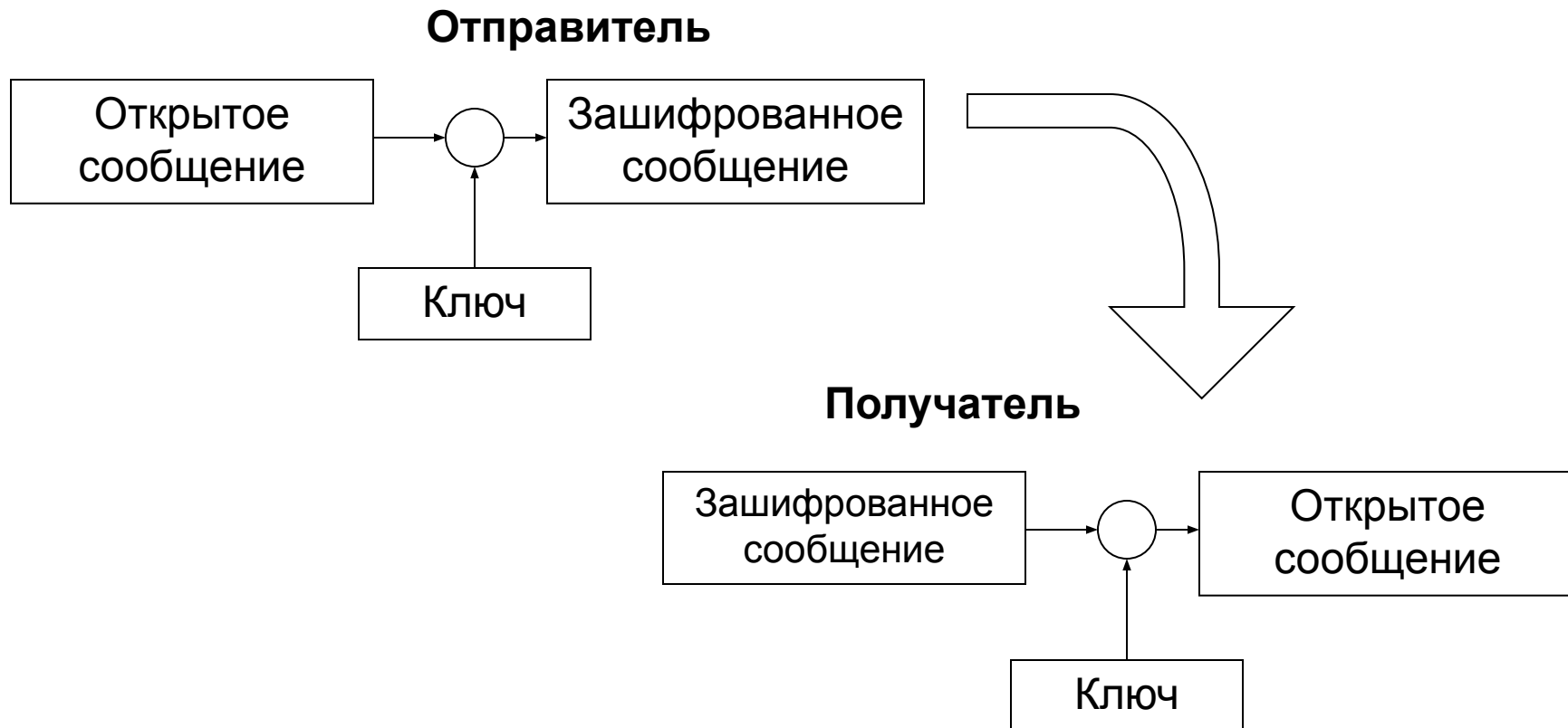
- Криптографические преобразования используются при реализации следующих сервисов безопасности:
 - Собственно шифрование (обеспечение конфиденциальности данных);
 - Контроль целостности;
 - Аутентификация.

Системы криптографической защиты информации

- Задача средств криптографической защиты информации — преобразование информационных объектов с помощью некоторого обратимого математического алгоритма.
- Процесс **шифрования** использует в качестве входных параметров объект – открытый текст и объект – ключ, а результат преобразования — объект – зашифрованный текст. При **дешифровании** выполняется обратный процесс.
- Криптографическому методу в ИС соответствует некоторый специальный алгоритм. При выполнении данного алгоритма используется уникальное числовое значение – **ключ**.
- Знание ключа позволяет выполнить обратное преобразование и получить открытое сообщения.
- Стойкость криптографической системы определяется используемыми алгоритмами и степенью секретности ключа.

Криптографические средства защиты данных

- Для обеспечения защиты информации в распределенных информационных системах активно применяются криптографические средства защиты информации.
- Сущность криптографических методов заключается в следующем:



Использование средств криптографической защиты для предотвращения угроз ИБ

- **Обеспечение конфиденциальности данных.** Использование криптографических алгоритмов позволяет предотвратить утечку информации. Отсутствие ключа у «злоумышленника» не позволяет раскрыть зашифрованную информацию;
- **Обеспечение целостности данных.** Использование алгоритмов несимметричного шифрования и хэширования делает возможным создание способа контроля целостности информации.
- **Электронная цифровая подпись.** Позволяет решить задачу отказа от информации.
- **Обеспечение аутентификации.** Криптографические методы используются в различных схемах аутентификации в распределенных системах (Kerberos, S/Key и др.).

Требования к системам криптографической защиты

■ *Криптографические требования*

- Эффективность применения злоумышленником определяется **средней долей дешифрованной информации**, являющейся средним значением отношения количества дешифрованной информации к общему количеству шифрованной информации, подлежащей дешифрованию, и трудоемкостью дешифрования единицы информации, измеряемой Q числом элементарных опробований.
- Под **элементарными опробованиями** понимается операция над двумя n -разрядными двоичными числами. При реализации алгоритма дешифрования может быть использован гипотетический вычислитель, объем памяти которого не превышает M двоичных разрядов. За одно обращение к памяти может быть записано по некоторому адресу или извлечено не более n бит информации. Обращение к памяти по трудоемкости приравнивается к элементарному опробованию.
- За единицу информации принимается общий объем информации обработанной на одном средстве криптографической защиты в течении единицы времени. Атака злоумышленника является успешной, если объем полученной открытой информации больше некоторого заданного объема V .

Требования к системам криптографической защиты

- ***Требования надежности.***
- Средства защиты должны обеспечивать заданный уровень надежности применяемых криптографических преобразований информации, определяемый значением допустимой вероятности неисправностей или сбоев, приводящих к получению злоумышленником дополнительной информации о криптографических преобразованиях.
- Регламентные работы (ремонт и сервисное обслуживание) средств криптографической защиты не должно приводить к ухудшению свойств средств в части параметров надежности.

Требования к системам криптографической защиты

- ***Требование по защите от несанкционированного доступа для средств криптографической информации в составе информационных систем.***
- В автоматизированных информационных системах, для которых реализованы программные или аппаратные средства криптографической защиты информации, при хранении и обработке информации должны быть предусмотрены следующие основные механизмы защиты:
 - идентификация и аутентификация пользователей и субъектов доступа;
 - управление доступом;
 - обеспечения целостности;
 - регистрация и учет.

Требования к системам криптографической защиты

- **Требования к средствам разработки, изготовления и функционирования средств криптографической защиты информации.**
- Аппаратные и программные средства, на которых ведется разработка систем криптографической защиты информации, не должны содержать явных или скрытых функциональных возможностей, позволяющих:
 - модифицировать или изменять алгоритм работы средств защиты информации в процессе их разработки, изготовления и эксплуатации;
 - модифицировать или изменять информационные или управляющие потоки, связанные с функционированием средств;
 - осуществлять доступ посторонних лиц к ключам идентификационной и аутентификационной информации;
 - получать доступ к конфиденциальной информации средств криптографической защиты информации.

Способы шифрования

- Различают два основных способа шифрования:
 - **Симметричное шифрование**, иначе шифрование с закрытым ключом;
 - **Ассиметричное шифрование**, иначе шифрование с открытым ключом;

Шифрование с секретным КЛЮЧОМ

- При симметричном шифровании процесс зашифровывания и расшифровывания использует некоторый *секретный ключ*.
- При симметричном шифровании реализуются два типа алгоритмов:
 - Поточное шифрование (побитовое)
 - Блочное шифрование (при шифровании текст предварительно разбивается на блоки, как правило не менее 64 бит)

Шифрование с секретным КЛЮЧОМ

- Выделяют следующие общие принципы построения шифров:
 - электронная кодовая книга (режим простой замены);
 - сцепление блоков шифра (режим гаммирования с обратной связью);
 - обратная связь по шифротексту;
 - обратная связь по выходу (режим гаммирования).

Шифрование с секретным КЛЮЧОМ

- Стандарт шифрования DES.
 - Алгоритм шифрования представляет собой блочный шифр, использующий подстановки, перестановки и сложения по модулю 2, с длиной блока 64 бита и длиной ключа 56 бит.
 - Подстановки и перестановки, используемые в DES фиксированы.

Алгоритм шифрования DES

- Основные этапы алгоритма шифрования
 - К блоку входного текста применяется фиксированная перестановка IP
 - Для каждого цикла (всего 16) выполняется операция зашифровывания:
 - 64 битный блок разбивается на две половины (левую x'' и правую x') по 32 бита
 - Правая половина x' разбивается на 8 тетрад по 4 бита. Каждая тетрада по циклическому закону дополняется крайними битами из соседних тетрад до 6-битного слова
 - Полученный 48-битный блок суммируется по модулю 2 с 48 битами ключа, биты которого выбираются на каждом цикле специальным образом из 56 бит, а затем разбиваются на 8 блоков по 6 бит

Алгоритм шифрования DES (продолжение)

- Каждый из полученных на предыдущем шаге блоков поступает на вход функции фиксированного S-блока, которая выполняет нелинейную замену наборов 6-битных блоков тетрадами
- Полученные 32 бита подвергаются фиксированной перестановке, результатом которой является полублок $F_i(x')$
- Компоненты правого зашифрованного полублока $F_i(x')$ суммируются по модулю 2 с компонентами левого полублока x'' и меняются местами, т.е. блок $(x'', F_i(x'))$ преобразуется в блок $(x'' + F_i(x'), x'')$
- К блоку текста, полученному после всех 16 циклов, применяется обратная перестановка IP^{-1}
- Результатом является выходной зашифрованный текст

Симметричное шифрование

- В процессе шифрования и дешифрования используется один и тот же параметр – секретный ключ, известный обеим сторонам
- Примеры симметричного шифрования:
 - ГОСТ 28147-89
 - DES
 - Blow Fish
 - IDEA
- Достоинство симметричного шифрования
 - Скорость выполнения преобразований
- Недостаток симметричного шифрования
 - Известен получателю и отправителю, что создает проблемы при распространении ключей и доказательстве подлинности сообщения

Симметричное шифрование

| Алгоритм | Размер ключа | Длина блока | Число циклов | Основные операции |
|---------------|------------------|-------------|--------------|--|
| DES | 56 | 64 | 16 | Перестановка, подстановка, \oplus |
| FEAL | 64, 128 | 64 | ≤ 4 | Сложение по модулю 2^8 , циклический сдвиг, \oplus |
| IDEA | 128 | 64 | 8 | Умножение по модулю $2^{16}+1$, сложение по модулю 2^{16} , \oplus |
| ГОСТ 28147-89 | 256 | 64 | 32 | Сложение по модулю 2^{32} , подстановка, циклический сдвиг, \oplus |
| RC5 | $8t, t \leq 255$ | 32, 64, 128 | ≤ 255 | Сложение по модулю 2^W , ($W=1/2$ длины блока), циклический сдвиг, \oplus |
| Blowfish | ≤ 448 | 64 | 16 | Сложение по модулю 2^{32} , подстановка, \oplus |

Несимметричное шифрование

- В несимметричных алгоритмах шифрования ключи зашифровывания и расшифровывания всегда разные (хотя и связанные между собой).
- Ключ зашифровывания является несекретным (открытым), ключ расшифровывания – секретным.

Несимметричное шифрование

- Алгоритм шифрования RSA (предложен Р.Ривестом, Э. Шамиром и Л.Адлманом) включает в себя:
 - Пусть заданы два простых числа p и q и пусть $n=pq$, $\phi(n)=(p-1)(q-1)$. Пусть число e , такое что числа e и $\phi(n)$ взаимно простые, а d – мультипликативно обратное к нему, то есть $ed \equiv 1 \pmod{\phi(n)}$. Числа e и d называются открытым и закрытым показателями соответственно. Открытым ключом является пара (n, e) секретным ключом – d . Множители p и q должны сохраняться в секрете.
 - Таким образом безопасность системы RSA основана на трудности задачи разложения на простые множители.

Несимметричное шифрование

- Кроме алгоритма RSA часто используемыми алгоритмами несимметричного шифрования являются:
 - Алгоритм Эль-Гамала (использует простое число p , образующую группы g и экспоненту $y=g^x \pmod{p}$)
 - Алгоритм шифрования Мессе-Омуры (использует простое число p , такое что $p-1$ имеет большой простой делитель в качестве открытого ключа, секретный ключ определяется в процессе диалога между приемником и источником)

Ассиметричное шифрование

- В криптографических преобразованиях используется два ключа. Один из них не секретный (открытый) ключ используется для шифрования. Второй, секретный ключ для расшифровывания.
- Примеры несимметричного шифрования:
 - RSA
 - Алгоритм Эль-Гамала
- Недостаток асимметричного шифрования
 - низкое быстродействие алгоритмов (из-за длины ключа и сложности преобразований)
- Достоинства:
 - Применение асимметричных алгоритмов для решения задачи проверки подлинности сообщений, целостности и т.п.

Сравнение симметричных и несимметричных алгоритмов шифрования

- Преимущества симметричных алгоритмов:
 - Скорость выполнения криптографических преобразований
 - Относительная легкость внесения изменений в алгоритм шифрования
- Преимущества несимметричных алгоритмов
 - Секретный ключ известен только получателю информации и первоначальный обмен не требует передачи секретного ключа
 - Применение в системах аутентификации (электронная цифровая подпись)

Проверка подлинности

- Криптографические методы позволяют контролировать целостность сообщений, определять подлинность источников данных, гарантировать невозможность отказа от совершенных действий
- В основе криптографического контроля целостности лежат два понятия:
 - Хэш-функция;
 - Электронная цифровая подпись.

Проверка целостности сообщений

- Контроль целостности потока сообщений помогает обнаружить их повтор, задержку, переупорядочивание или утрату. Для контроля целостности сообщений можно использовать хэш-функцию.
- **Хэш-функция** – преобразование преобразующее строку произвольной длины в строку фиксированной длины и удовлетворяющее следующим свойствам:
 - Для каждого значения $H(M)$ невозможно найти аргумент M – стойкость в смысле обращения;
 - Для данного аргумента M невозможно найти аргумент M' , что $H(M) = H(M')$ – стойкость в смысле возникновения коллизий.
- Хэш-функция используется:
 - Для создания сжатого образа сообщения, применяемого в ЭЦП;
 - Для защиты пароля;
 - Для построения кода аутентификации сообщений.

Контроль подлинности

- **Электронная цифровая подпись** выполняет роль обычной подписи в электронных документах для подтверждения подлинности сообщений – данные присоединяются к передаваемому сообщению, подтверждая подлинность отправителя сообщения.
- При разработке механизма цифровой подписи возникает три задачи:
 - создание подписи таким образом, чтобы ее невозможно было подделать;
 - возможность проверки того, что подпись действительно принадлежит указанному владельцу.
 - предотвращение отказа от подписи.

Алгоритм формирования электронной цифровой подписи

- При формировании цифровой подписи по классической схеме отправитель:
 - Применяет к исходному тексту хэш-функцию;
 - Дополняет хэш-образ до длины, требуемой в алгоритме создания ЭЦП;
 - Вычисляет ЭЦП по хэш-образу с использованием секретного ключа создания подписи.
- Получатель, получив подписанное сообщение, отделяет цифровую подпись от основного текста и выполняет проверку:
 - Применяет к тексту полученного сообщения хэш-функцию;
 - Дополняет хэш-образ до требуемой длины;
 - Проверяет соответствие хэш-образа сообщения полученной цифровой подписи с использованием открытого ключа проверки подписи.

Примеры алгоритмов формирования хэш-функции и ЭЦП

- В качестве распространенных алгоритмов хэширования можно указать:
 - MD5;
 - SHA;
 - ГОСТ Р34.11-94;
- Алгоритмы формирования электронной цифровой подписи:
 - RSA;
 - DSA;
 - ГОСТ Р34.10-94

Выбор алгоритмов аутентификации

- При выборе протоколов аутентификации, необходимо определить, какой тип аутентификации требуется – односторонняя или двусторонняя, наличие доверенной стороны и т.д.
- Параметры протокола аутентификации:
 - Тип алгоритма (симметричный, несимметричный);
 - Конкретный вид алгоритма;
 - Режим работы;
 - Процедура управления ключами;
 - Совместимость используемых алгоритмов.

Вопросы???

