## Методы обеспечения информационной безопасности автоматизированных систем (основные понятия, угрозы).

Основные понятия:

* **Политика безопасности** (Security Policy). Совокупность норм и правил, обеспечивающих эффективную защиту системы обработки информации от заданного множества угроз безопасности.
* **Модель безопасности** (Security Model). Формальное представление политики безопасности
* **Дискреционное, или произвольное управление доступом** (Discretionery Access Control). Управление доступом, осуществляемое на основании заданного администратором множества разрешенных отношений доступа (например, в виде "троек" – <объект, субъект, тип доступа>).
* **Мандатное, или нормативное, управление доступом** (Mandatory Access Control). Управление доступом основанное на совокупности правил предоставления доступа, определенных на множестве атрибутов безопасности субъектов и объектов, например, в зависимости от грифа секретности информации и уровня доступа пользователя.
* **Ядро безопасности** (Trusted Computing Base (ТСВ)). Совокупность аппаратных, программных и специальных компонент ВС, реализующих функции защиты и обеспечения безопасности.
* **Идентификация** (Identification). Процесс распознания сущностей путем присвоения им уникальных меток (идентификаторов).
* **Аутентификация** (Autentification). Проверка подлинности предъявленных идентификаторов сущностей.
* **Адекватность** (Assurance). Показатель реально обеспечиваемого уровня безопасности, отражающий степень эффективности и надежности реализованных средств защиты и их соответствия поставленным задачам (в основном политике безопасности).
* **Квалификационный анализ, квалификация уровня безопасности** (Evaluation). Анализ системы с целью определения уровня ее защищенности и соответствия требованиям безопасности на основе критериев стандарта безопасности.
* **Прямое взаимодействие** (Trusted Path). Принцип организации информационного взаимодействие (как правило, между пользователем и системой), гарантирующий, что передаваемая информация не подвергается перехвату или искажению.

Угрозы:

**• Угроза безопасности вычислительной системе** - воздействия на систему, которые прямо или косвенно могут нанести ущерб ее безопасности.

**Виды угроз**

• Конфиденциальности;

• Целостности;

• Доступности.

**Угрозы безопасности КС**

Цель защиты АС – противодействие угрозам безопасности

## Методы обеспечения информационной безопасности автоматизированных систем (методы взлома, защита от взлома).

**Методы взлома АС**

• Атаки на уровне:

– ОС;

– Сетевого ПО;

– СУБД.

**Атаки на уровне ОС**

• Структура ОС сложна -> защитить сложно.

• Эффективны не только сложные виды атак.

• Успех атаки зависит от архитектуры и конфигурации ОС.

• Кража пароля

– подглядывание за пользователем;

– из файла на компьютере;

– записан возле рабочего места;

– кража носителя;

– атака с перебором.

• Сканирование жесткого диска

• Сборка мусора

• Превышение полномочий

– Запуск от имени администратора;

– Подмена системных библиотек;

– Модификация системы защиты ОС.

• Отказ в обслуживании

– Захват ресурсов;

– Бомбардировка запросами;

– Использование ошибок ПО или администрирования.

**Атаки на уровне сетевого ПО**

• Прослушивание сегмента локальной сети

• Перехват сообщений на маршрутизаторе

• Создание ложного маршрутизатора

• Навязывание сообщений

• Отказ в обслуживании

**Защита на уровне сетевого ПО**

• Максимальное ограничение размеров компьютерной сети

• Изоляция сети от внешнего мира

• Шифрование сетевых сообщений

• Электронная цифровая подпись сетевых сообщений

• Использование межсетевых экранов

**Атаки на уровне СУБД**

• СУБД, содержащая ошибки в программном обеспечении;

• Грубые ошибки при определении политики безопасности.

**ЗАЩИТА АВТОМАТИЗИРОВАННЫХ СИСТЕМ ОТ ВЗЛОМА**

**Защита АС от взлома**

• постоянно повышайте квалификацию в области защиты компьютерных систем;

• руководствуйтесь принципом разумной достаточности (чем мощнее защита, тем больше ресурсов компьютерной системы она требует);

• храните в секрете информацию о принципах действия защитных механизмов АС; • постарайтесь максимально уменьшить размеры АС и без крайней необходимости не подключайте ее к Internet;

• перед покупкой нового ПО поищите информацию о нем в сети Internet;

• размещайте серверы в охраняемых помещениях, не подключайте к ним клавиатуру и дисплеи, чтобы доступ к ним осуществлялся только через сеть;

• Сообщения, нуждающиеся в защите и передаваемые по незащищенным каналам связи, должны шифроваться;

• При стыковке защищенной сети с незащищенной все сообщения должны проходить через межсетевые экраны;

• не пренебрегайте возможностями аудита (интервал просмотра журнала аудита не должен превышать одних суток);

• если окажется, что число событий в журнале аудита велико, изучите новые записи, так как не исключено, что КС подверглась атаке взломщика;

• регулярно проводите проверку целостности программного обеспечения АС, проверяйте АС на наличие в ней программных закладок;

• регистрируйте все изменения в политике безопасности в обычном бумажном журнале (регулярная проверка поможет обнаружить присутствие программной закладки);

• пользуйтесь защищенными ОС;

• создайте несколько ловушек для взломщиков;

• регулярно тестируйте КС с помощью специальных программ оценки степени защищенности КС.

## Методы обеспечения информационной безопасности автоматизированных систем (защита от программных закладок).

**Защита от программных закладок**

• Предотвращение внедрения программной закладки;

• Обнаружение внедренной программной закладки;

• Удаление программной закладки.

**Предотвращение внедрения закладок**

• Изолированная среда:

– BIOS и операционная система не содержат программных закладок;

– гарантированно установлена неизменность BIOS и операционной системы в данном сеансе работы компьютера;

– на компьютере не запускалось и не запускается никаких других программ, не проверенных на закладки;

– исключен запуск проверенных программ вне изолированного компьютера.

**Обнаружение внедренной программной закладки**

• Обнаружение признаков присутствия в системе:

– качественные и визуальные признаки (обнаруживаются пользователем);

– обнаруживаемые средствами диагностики.

**Удаление программной закладки**

• Определяется методом внедрения.

• Программно-аппаратная закладка – перепрограммировать ПЗУ;

• Загрузочная, драйверная, прикладная, замаскированная, закладка-имитатор – произвести замену на соответствующее ПО от доверенных источников.

• Исполняемая - убрать код закладки из исходного кода программного модуля и откомпилировать модуль заново.

## Политика безопасности. Модель КС.

**Политика безопасности**

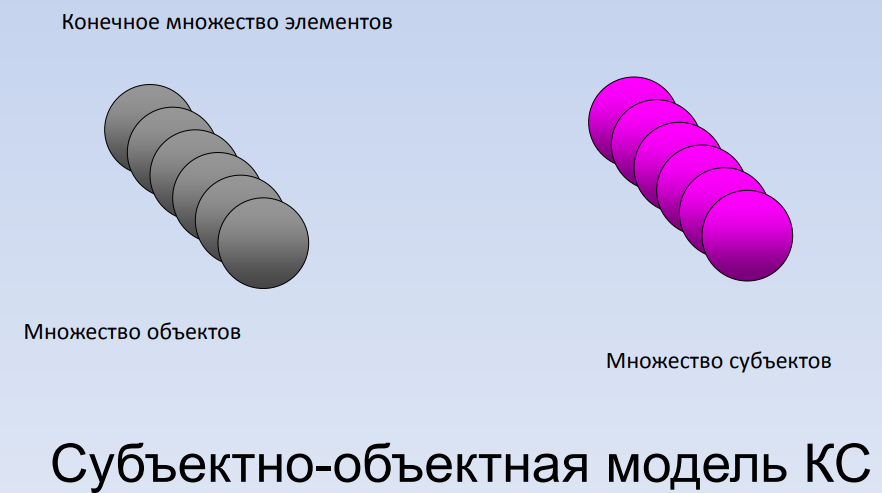
• Политика безопасности (Security Policy). Совокупность норм и правил, обеспечивающих эффективную защиту системы обработки информации от заданного множества угроз безопасности.

Описание политики безопасности включает:

• Множество возможных операций над объектами.

• Для каждой пары "субъект-объект" назначение множества разрешенных операций, являющегося подмножеством всего множества возможных операций.

**Модель компьютерной системы**

****

**Примеры субъектов и объектов**

объект

* Файл на внешнем носителе
* Область оперативной памяти
* Запись в базе данных

субъект

* Исполняемый файл, загруженный в оперативную память и которому передано управление

## Замкнутая программная среда.

**ЗАМКНУТАЯ ПРОГРАММНАЯ СРЕДА**

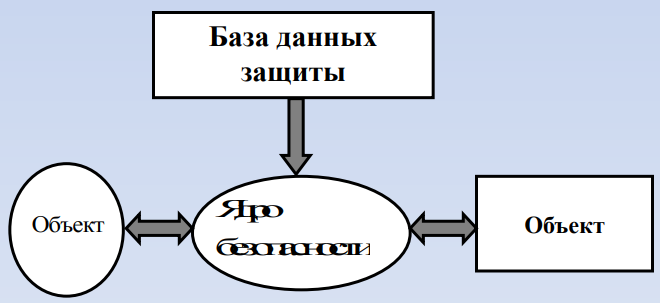
• Определение 12. АС называется замкнутой по порождению субъектов, если в ней действует МБС, разрешающий порождение только фиксированного конечного подмножества субъектов для любых объектов-источников, рассматриваемых для фиксированной декомпозиции компьютерной системы на субъекты и объекты

Определение 13. Множество субъектов АС называется изолированным, если в ней действует МБС и субъекты из порождаемого множества корректны относительно друг друга и МБС.

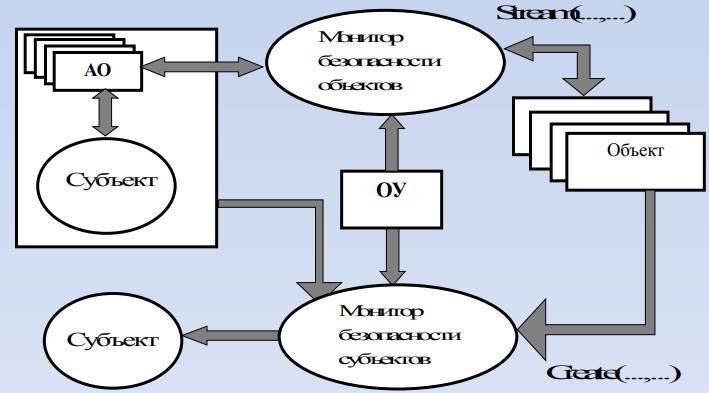
Определение 14. Операция порождения субъекта Create (Sk , Om)-> Si называется порождением с контролем неизменности объекта, если для любого момента времени t>t0 , в который активизирована операция порождения Create, порождение субъекта Si возможно только при тождественности объектов Om[t0] и Om[t].

• Утверждение 2 (базовая теорема ИПС). Если в момент времени t0 в изолированной КС действует только порождение субъектов с контролем неизменности объекта и существуют потоки от любого субъекта к любому объекту, не противоречащие условию корректности субъектов, то в любой момент времени t>t0 КС также остается изолированной

**Классическая модель ядра безопасности**

****

**Ядро безопасности с учетом контроля порождения субъектов**

****

## Формирование и поддержка изолированной программной среды.

**Формирование и поддержка ИПС**

• Предположим, что в КС работают N субъектов- пользователей, каждый i-й из которых характеризуется некоторой персональной информацией Ki, не известной другим пользователям и хранящейся на некотором материальном носителе.

• Существует выделенный субъект - администратор системы, который знает все Кi.

• Администратор КС присваивает i-му пользователю полномочия, заключающиеся в возможности исполнения им только заданного подмножества программ, Тi={Pi1, Pi2, ... Pit}.

**Защитный механизм должен**

• В некоторый начальный момент времени требовать у субъекта предъявления аутентифицирующей информации и по ней однозначно определять субъекта и его полномочия Тi; • В течении всего времени работы пользователя i должен обеспечивать выполнение программ только из подмножества Ti;

• Пользователь не должен иметь возможности изменить подмножество Тi и/или исключить из дальнейшей работы защитный механизм и его отдельные части.

**Виды НСД**

• Непосредственный НСД (Злоумышленник, используя некоторое ПО пытается непосредственно осуществить операции чтения или записи (изменения) интересующей его информации.)

• Опосредованный НСД (Обусловлен общностью ресурсов пользователей и заключается во влиянии на работу другого пользователя через используемые им программы (после предварительного изменения их содержания или их состава злоумышленником).)

**ИПС**

• При выполнении всех условий программная среда называется изолированной.

• От системы защиты требуется:

1. Невозможность запуска программ помимо контролируемых ИПС событий.

2. Отсутствие в базовом ПО возможностей влиять на среду функционирования уже запущенных программ (требование невозможности редактирования оперативной памяти).

## Реализация ИПС с использованием механизма расширения BIOS

**Условия**

• 2 этапа - этап установки ИПС и этап эксплуатации ИПС.

• N пользователей, каждый i-й имеет персональную информацию Ki, не известную другим пользователям и хранящуюся на материальном носителе (например, устройстве Touch Memory).

• Администратор КС - знает все Кi и проводит этап установки.

• Пользователи участвуют в этапе эксплуатации.

**Установка ИПС**

1. В ПЭВМ устанавливается плата, реализует:

• чтение Ki,

• идентификацию пользователя i по Ki,

• чтение множества доступных для выполнения пользователем i задач Pi1, Pi2, ..., Pim, и информации Mi1, Mi2, ... Mim, фиксирующей целостность файлов Fi1, ..., Fim каждой задачи.

1. Администратор определяет для пользователя i набор задач и соответствующих задачам исполняемых файлов {Pit, Fit}, t=1,...,mi; i=1, ..., N, где mi - число разрешенных к запуску задач для i-го пользователя.
2. Администратор формирует (и заносит на носитель) или считывает с носителя для i-го пользователя его Ki и вычисляет значения для последующего контроля целостности Mir=f(Ki, Fir, Pir), где f - функция фиксации целостности (хэш- функция).
3. Администратор проделывает действия 2 и 3 для всех N пользователей.
4. Администратор устанавливает в ПС модуль активизации ИПС и фиксирует его целостность. Фиксируется также целостность файлов ОС Fос.

**Эксплуатация ИПС**

1. Включение питания и активизация расширенного BIOS:

а) идентификация пользователя по его Ki (при успехе п.

б); б) проверка целостности всех включенных в ЭВМ BIOS (при положительном исходе п. в);

в) чтение по секторам файлов ОС и проверка их целостности;

г) чтение как файлов Рипс (с помощью функций ОС) и проверка целостности;

д) активизация сетевого ПО;

е) активизация процесса контроля Pипс;

ж) запуск избранной задачи i-го пользователя.

2. Работа в ИПС. Запуск процесса Рs сопровождается проверками:

а) принадлежит ли Fs к множеству разрешенных для i (Ti), если да, то п. б), иначе запуск игнорируется;

б) совпадает ли G=f(Ki,Fs,Ps) c M=f(Ki,Fs,Ps), вычисленной администратором;

в) при положительном исходе проверки б) задача запускается.

## UEFI. Принципы работы

**Что такое UEFI?**

• UEFI (Unified Extensible Firmware Interface — единый расширяемый интерфейс встроенного ПО)

• Интерфейс между операционной системой и микропрограммами, управляющими низкоуровневыми функциями оборудования

• Назначение: корректно инициализировать оборудование при включении системы и передать управление загрузчику операционной системы

• Цель UEFI заключается в определении стандартного способа взаимодействия операционной системы со встроенным ПО платформы во время процесса загрузки.

• До появления UEFI основным механизмом взаимодействия с оборудованием во время процесса загрузки были программные прерывания.

• UEFI позволяет реализовать модульную структуру встроенного ПО, которая предоставляет разработчикам оборудования и систем значительную гибкость в разработке встроенного ПО для более требовательных современных вычислительных сред

**Безопасная загрузка**

• UEFI имеет процесс проверки встроенного ПО, который называется безопасной загрузкой (Secure boot).

• Безопасная загрузка определяет, как встроенное ПО платформы управляет сертификатами безопасности, проверкой встроенного ПО и реализацией интерфейса (протокола) между встроенным ПО и операционной системой.

• Защита от вредоносных загрузчиков.

• До запуска операционной системы разрешен запуск только подписанных и сертифицированных "известных" загрузчиков и кода

• Среда, предшествующая загрузке ОС, уязвима для атак, осуществляемых с помощью передачи функций загрузчика вредоносным загрузчикам.

• Эти загрузчики остаются необнаруженными мерами безопасности операционной системы и антивирусным ПО

**BIOS – UEFI**

• Возможна загрузка вредоносного ПО

• Использование загрузчика с ЭП

**Как работает UEFI**

• При включении питания компьютера запускается процесс выполнения кода, который настраивает процессор, память и периферийные устройства, подготавливая их к выполнению операционной системы.

• Этот процесс выполняется одинаково для всех платформ независимо от архитектур, на которых они основаны (x86, ARM и т. д.).

• Проверяется сигнатура кода встроенного ПО, присутствующего на периферийных устройствах, таких как сетевые карты, запоминающие устройства или видеоадаптеры.

• Этот код на устройствах, называемый дополнительными ПЗУ, продолжает процедуру настройки, обеспечивая подготовку периферийных устройств к передаче управления ОС.

• Встроенное ПО проверяет внедренную сигнатуру

• Если эта сигнатура присутствует в БД сигнатур, то этот модуль получает разрешение на выполнение.

• БД определяет, может ли продолжиться процесс загрузки

• БД с разрешенными загрузчиками содержит хэши надежного ПО и загрузчиков ОС.

• Другая БД содержит хэши вредоносных программ.

## Персональное средство аутентификации eToken

* Электронные ключи eToken могут использоваться в любых приложениях для замены парольной защиты на более надежную двухфакторную аутентификацию.
* Например, если для аутентификации пользователю необходимо предоставить USBключ и ввести пароль, то злоумышленник не сможет получить доступ к данным, так как ему нужно не только подсмотреть пароль, но и предъявить физическое устройство, кража которого быстро обнаружима.

Особенности eToken:

* Строгая двухфакторная аутентификация пользователей при доступе к защищенным ресурсам (компьютерам, сетям, приложениям)
* Аппаратное выполнение криптографических операций в доверенной среде (в микросхеме ключа: генерация ключей шифрования, симметричное и асимметричное шифрование, вычисление хэш-функции, выработка электронной подписи)
* Безопасное хранение критически важных данных – криптографических ключей, профилей пользователей, настроек приложений, цифровых сертификатов и пр. в энергонезависимой памяти ключа
* Сертифицированные версии для защиты информации в AC до класса защищенности 1Г включительно и для защиты персональных данных в ИСПДн до 1 класса включительно

## eToken API

Для использования eToken в разрабатываемом ПО существует 3 вида API:

* PKCS#11
* CAPI
* SAPI PKCS#11

PKCS#11 (Public-Key Cryptography Standard - Стандарт криптографии с открытым ключом) используется для работы с асимметричными криптографическими алгоритмами, был разработан RSA Laboratories и включает в себя как зависимые от алгоритма, так и не зависимые стандарты разработки. Это промышленный стандарт, который задает интерфейс для криптографических устройств, таких как смарт-карты и карты PCMCIA. Этот стандарт определяет API (application programming interface - интерфейс программирования приложений), называемый Cryptoki (Cryptographic Token Interface), для устройств, физических или виртуальных, содержащих криптографическую информацию (ключи или данные) и выполняющих криптографические функции. Этот API используется на многих платформах и обладает достаточными возможностями для большинства приложений, связанных с безопасностью. Компания Аладдин рекомендует использовать PKCS#11 в качестве основного API для программирования eToken.

CAPI (CryptoAPI) разработан компанией Microsoft в качестве части операционных систем Microsoft Windows. Предполагалось, что он будет использоваться для разработки приложений на платформе MS Windows. CAPI позволяет одновременно использовать несколько криптопровайдеров (cryptographic service providers, CSP) на одном компьютере или в одном приложении. Также он позволяет ассоциировать конкретный криптопровайдер с конкретной смарт-картой, так чтобы приложения вызывали корректный криптопровайдер при работе с криптографией. ОС Windows содержит много вспомогательных функций, позволяющих упростить код при работе с криптографией или сложными структурами данных (например сертификатами).

SAPI (Supplementary API – дополнительный API) был реализован в версии eToken RTE 3.60 для того, чтобы убрать необходимость в использовании низкоуровневых функций при работе с eToken. Он давал доступ к специфичным для eToken возможностям, не рассматриваемым в стандарте PKCS#11. Этот функционал теперь доступен в PKCS#11 API. Выбор API зависит от конкретных нужд разработчика.

Вот несколько ключевых отличий:

* PKCS#11 позволяет управлять несколькими eToken одновременно. В CAPI нет понятия физического токена (необходимо использовать специальные техники).
* PKCS#11 позволяет ожидать уведомлений о подключении/удалении eToken. CAPI не обладает такой возможностью (возможность есть через Win32 API).
* PKCS#11 позволяет хранить ключи RSA, сертификаты и данные на eToken. CAPI позволяет хранить только ключи RSA и соответствующие им сертификаты.
* В PKCS#11 нет вспомогательных функций для работы с сертификатами. Обработка и проверка сертификатов X.509 может быть довольно сложной задачей. Однако, работая в ОС Windows, имеется возможность использовать функции Win32 даже работая с PKCS#11.
* PKCS#11 - это API, а не архитектура. Если приложению требуется работать с несколькими провайдерами, оно само должно определить как взаимодействовать с ними. CAPI является частью ОС Windows и поэтому, как только новый провайдер установлен в систему, он автоматически становится доступен для всех приложений.
* В CAPI имеется множество вспомогательных функций. Они могут помочь программисту сконцентрироваться на логике работы приложения, не сталкиваясь с низкоуровневыми проблемами.

## Безопасное взаимодействие в КС. Процедуры идентификации и аутентификации.

Объекты, аутентифицирующие пользователя

* Внешний аутентифицирующий объект, не принадлежащий АС
* Внутренний аутентифицирующий объект, принадлежащий АС, в который переносится информация из внешнего объекта
* Дополнительно: субъект переноса информации от внешнего к внутреннему объекту (пример - драйвер клавиатуры).

Содержание аутентифицирующего объекта

* Di (логин) – неизменяемый идентификатор i-го пользователя, который является аналогом имени и используется для идентификации пользователя
* Ki (пароль) – аутентифицирующая информация пользователя, которая может изменяться и служит для аутентификации.

Примеры

* Touch Memory (ТМ) имеет 8 байт неперезаписываемого неповторяющегося серийного номера, который однозначно характеризует кокретную ТМ и некоторый объект перезаписываемой памяти, который может содержать Ki
* Пластиковые карты - выделяется неизменяемая информация первичной персонализации пользователя, соответствующая IDi , и объект в файловой структуре карты, содержащий Ki
* Аналогично: eToken.

Требования

* Внутренний аутентифицирующий объект не должен существовать в КС длительное время (большее времени работы конкретного пользователя).
* Для постоянного хранения необходимо использовать некую преобразованную информацию от первичной (хэш).

Аутентификация на основе пароля

* Часто пароли выбираются из множества осмысленных слов
* Необходимы жесткие требования к паролям
* Для повышения надежности защиты необходимо добавить аппаратную аутентификацию (добавляется 2-ой фактор)

## Аутентификация до загрузки ОС

Возможна в 2 вариантах:

* На уровне расширений BIOS
* На уровне загрузчика ОС

Уровень расширений Bios

* В ЭВМ на платформе Intel, первичная активизация вычислительных ресурсов компьютера производится кодом процессора, хранящемся в основном Bios.
* При включении питания код основного Bios "проецируется" в область памяти F000 и управление передается на точку входа, определенную производителем Bios
* Код Bios производит тестирование оборудования, инициализацию векторов прерываний, активизацию видеосистемы и др., зависящие от специфики Bios
* В состав кода Bios входит типовая процедура поиска так называемых расширений Bios (Bios Extention).
* Управление передается в ходе процедуры поиска расширений
* С учетом того, что объем расширения Bios не может быть очень большим, то на этом уровне может быть реализован достаточно небольшой объем значимых для безопасности функций

Функции, реализуемые в расширениях

* Идентификация и аутентификация пользователя (возможно, с использованием специфического аппаратного носителя;
* Запрет несанкционированной загрузки ОС с избранных носителей (например с CD-ROM);
* Контроль неизменности или целостности аппаратной или программной компоненты ЭВМ.

Уровень загрузчиков ОС

Возможно решение задач:

* “Ранняя" идентификация и аутентификация пользователей (при отсутствии аппаратных средств защиты)
* Защита от несанкционированной загрузки ОС
* Получение специального вида загрузочных носителей.

“Ранняя" идентификация и аутентификация

* Не всегда процедуры идентификации и аутентификации удается реализовать на этапе инициализации аппаратной компоненты компьютера (в частности, невозможно реализовать указанные процедуры в расширении Bios).
* Удается выполнить идентификацию и аутентификацию на ранней стадии сеанса работы пользователя (Программно).

Защита от несанкционированной загрузки ОС

* Используют тонкости обработки загрузки с внешних носителей либо преобразуют (например, шифруют) информацию на несъемных носителях компьютера.
* В первом случае загрузка с внешних носителей операционной системы невозможна физически
* Во втором – даже при успешной загрузке с несанкционированной копии ОС информация недоступна.

Решение

* В общем случае - программирование модифицированного загрузчика (или загрузчиков) операционной системы.

Необходимые операции

* Заместить первичный код загрузчика собственным фрагментом;
* Сохранить исходный код загрузочного сектора (в случае необходимости его выполнения);
* С учетом необходимости размещения первичного загрузчика по тому же адресу, что и модифицированного, обеспечить корректное перемещение модифицированного загрузчика в другую область памяти без потери управления.

## Контроль и управление доступом. Диспетчер доступа.

Контроль и управление доступом

* Основная задача - ограничение операций, выполняемых зарегистрированными пользователями в системе.
* Два основных механизма управления доступом – дискреционный (произвольный) и мандатный (нормативный).

**Произвольное управление доступом**

* Основа - матрица прав доступа
* Строки - субъекты (пользователи, процессы)
* Столбцы – объекты (файлы, каталоги и т.п.)
* В ячейках - права доступа субъектов к объектам

Варианты реализации

* Списки прав доступа (Access Control List – ACL; С каждым объектом ассоциируется список пользователей с указанием их прав доступа к объекту; При принятии решения о доступе, соответствующий объекту доступа ACL проверяется на наличие прав, ассоциированных с идентификатором пользователя, запрашивающего доступ, или его группы.)
* Биты доступа ( Привязаны к объектам. Указывают права доступа для трех категорий пользователей (ОС Unix): все пользователи (world), члены группы владельца (group) и владелец (owner). Может изменять только владелец объекта и администратор.)
* “Парольная” защита. (Пользователь использует отдельный пароль для доступа к каждому объекту в системе; Неудобства, связанные с запоминанием паролей.)

**Нормативное управление доступом**

* Полностью запрещает передачу прав доступа между пользователями. Полномочное – нет.
* Позволяет решить проблему "троянских коней”
* Модели Белла-Лападула и Биба.
* Объектам задается метка секретности, субъектам – уровень доступа.
* Запрещается запись в объекты более низкого уровня и чтение из объектов более высокого уровня, чем уровень доступа субъекта.

Примеры:

* ОС UTS MLS (доработанная Linux)
* Информация о метке секретности объекта содержится в битах доступа файла
* Одновременно работает произвольный доступ (поле все пользователи)

## Назначение, функции, принцип работы ПАК «Аккорд».

**Аккорд-АМДЗ**

• Аппаратный модуль доверенной загрузки - обеспечивает доверенную загрузку ОС вне зависимости от ее типа для аутентифицированного пользователя

**Компоненты**

Аппаратные компоненты:

• Контроллер;

• Контактное устройство;

• Идентификатор;

Программные компоненты:

• BIOS контроллера комплекса Аккорд-АМДЗ;

• Firmware, в котором реализованы функции АМДЗ

**Защитные функции комплекса**

• идентификацию пользователя по уникальному Идентификатору;

• аутентификацию с учетом необходимой длины пароля и времени его жизни;

• аппаратный (до загрузки ОС) контроль целостности технических средств СВТ, программ и данных на жестком диске (в том числе системных областей диска и модулей программной части комплекса);

• ограничение времени доступа субъекта к СВТ в соответствии с установленным режимом работы пользователей;

• блокировку несанкционированной загрузки СВТ с отчуждаемых носителей (FDD, CD-ROM, ZIP-drive, USB-disk и др.);

2) процедур блокирования экрана и клавиатуры по команде пользователя или по истечению установленного интервала «неактивности» пользователя;

3) Разграничение доступа к локальным и сетевым ресурсам

4) Управление процедурами ввода/вывода на отчуждаемые носители информации. Для каждого пользователя контролируется список разрешённых USB-устройств и SD карт в соответствии с их уникальными идентификационными номерами;

5) Контроль доступа к любому устройству, или классу устройств, доступных в «Диспетчере устройств» Windows, в том числе последовательных и параллельных портов, устройств PCMCI, IEEE 1394, WiFi, Bluetooth и пр;

6) Гарантированная очистка оперативной памяти и остаточной информации на жестких дисках и внешних носителях;

7) Контроль вывода на печать документов из любых программ, автоматическая маркировка печатных листов специальными пометками, грифами и т.д. Процесс печати протоколируется в отдельном журнале (создаётся учетная карточка документа);

8) регистрация контролируемых событий, в том числе несанкционированных действий пользователей, в системном журнале, доступ к которому предоставляется только Администратору БИ;

9) Защиты от НСД систем терминального доступа;

10) Контроль целостности критичных с точки зрения информационной безопасности программ и данных.

• Статический список (проверка выполняется однократно в начале сеанса, а далее с периодичностью, заданной администратором)

• Динамический список, проверка по которому выполняется при каждой загрузке контролируемого файла в оперативную память.

11) Создание изолированной программной среды за счет использования защитных механизмов комплекса;

12) Встраивание или совместное использование других средств защиты информации, в том числе криптографических;

## Назначение, функции, принцип работы ПАК “Соболь”

**Что такое ПАК «Соболь»?**

• ПАК «Соболь» – это программно-аппаратное средство защиты информации от несанкционированного доступа, выполняющее роль аппаратно-программного модуля доверенной загрузки (АПМДЗ).

**Для чего предназначен ПАК «Соболь»?**

• защиты конфиденциальной информации, персональных данных, гос. тайны (гриф «Совершенно Секретно»).

• предотвращения доступа неавторизованных пользователей к информации, обрабатываемой на компьютере.

• информирования администратора комплекса о всех важных событиях ИБ.

• предоставления случайных чисел прикладному ПО.

**Применение**

• Применятся для защиты автономного компьютера, а также рабочей станции или сервера, входящих в состав локальной вычислительной сети (ЛВС).

**Возможности ПАК «Соболь»**

• Идентификация и аутентификация пользователей

В качестве персональных идентификаторов пользователей могут применяться:

iButton

– eToken PRO и eToken PRO (Java)

– Rutoken

– iKey 2032

– Смарт-карты eToken PRO

• Блокировка загрузки ОС со съемных носителей

* После успешной загрузки штатной копии ОС доступ к этим устройствам восстанавливается.
* Запрет распространяется на всех пользователей компьютера, за исключением администратора.

• Регистрация попыток доступа к ПЭВМ

Электронный замок «Соболь» осуществляет ведение системного журнала, записи которого хранятся в специальной энергонезависимой памяти:

1. факт входа пользователя и имя пользователя;

2. предъявление незарегистрированного идентификатора;

3. ввод неправильного пароля;

4. превышение числа попыток входа в систему;

5. дата и время регистрация событий НСД.

• Контроль целостности системного реестра Windows

Данная возможность позволяет контролировать неизменность системного реестра Windows, что существенно повышает защищённость рабочих станций от несанкционированных действий внутри операционной системы

• Контроль целостности программной среды

Используемый в комплексе "Соболь" механизм контроля целостности позволяет контролировать неизменность файлов и физических секторов жесткого диска до загрузки операционной системы, а также файловых систем: NTFS, FAT 32, FAT 16, UFS, UFS2, EXT3, EXT2, EXT4 в ОС семейства Linux и MS Windows

• Контроль конфигурации

Возможность контролировать неизменность конфигурации компьютера – PCI-устройств, ACPI, SMBIOS и оперативной памяти.

**Достоинства ПАК «Соболь»**

• Наличие сертификатов ФСБ и ФСТЭК России.

• Защита информации, составляющей государственную тайну.

• Предоставление ресурсов в построении прикладных криптографических приложений.

• Простота в установке, настройке и эксплуатации.

• Поддержка 64-битных ОС Windows (в том числе Windows 8.1 и Windows server 2012 R2) и Linux.

• Поддержка различных типов идентификаторов.

• Гибкий выбор форматов исполнения платы (PCI, PCI-E, Mini PCI-E, Mini PCI-E Half) и вариантов комплектации.

• Возможность программной инициализации комплекса.

• Поддержка файловой системы EXT 4 в ОС семейства Linux.

• Интеграция с другими продуктами «Кода Безопасности».

• Физический датчик случайных чисел.

## Персональные идентификаторы. Виды, назначение, функции.

iButton

* iButton (Touch Memory) — это семейство многофункциональных микроэлектронных устройств, разработанных фирмой Dallas Semiconductor (USA) в настоящее время выпускаемых фирмой Maxim).
* Каждое устройство iButton имеет уникальный номер (ID), записываемый в процессе изготовления.
* Все устройства iButton помещаются в стальной цилиндрический корпус MicroCan, выполнены по жестким стандартам и выдерживают серьезные механические и температурные нагрузки.
* Обмен данными с iButton производится через интерфейс 1-Wire. Информация в этом интерфейсе передается по единственному проводнику. Питание iButton получают из этого же проводника, заряжая внутренний конденсатор в моменты, когда на шине нет обмена данными.
* Скорость обмена достаточна для обеспечения передачи данных в момент касания контактного устройства.

Области применения:

* средства идентификации;
* системы контроля доступа;
* системы компьютерной безопасности;
* температурный мониторинг и др.

ruToken

Программные и аппаратные средства для идентификации и аутентификации пользователей, электронной подписи и безопасного хранения криптографических ключей.

Модели

* Рутокен ЭЦП (Электронный идентификатор с аппаратной реализацией электронной подписи, шифрования и хеширования по ГОСТ. Также имеется поддержка RSA 2048. Существует модификация, Рутокен КП, используемая только с продуктами КриптоТРИ и КриптоПРО Рутокен CSP.)
* Рутокен ЭЦП Flash (+flash память до 64 Гб)
* Рутокен ЭЦП Bluetooth (работает с моб. устр.)
* Рутокен PINPad (Решение класса TrustScreen, позволяющее визуализировать подписываемый документ в доверенной среде непосредственно перед наложением электронной подписи. Документ отображается на экране и, в случае подтверждения пользователем корректности информации, подпись осуществляется непосредственно на самом устройстве.)
* Рутокен S (без ЭЦП)
* Рутокен Lite (ключевой носитель без криптографических функций)
* Рутокен Web (Основное назначение — замена небезопасной аутентификации по связке «логин-пароль» на двухфакторную аппаратную аутентификацию. Решение основано на технологии электронной подписи и позволяет свести к нулю риск кражи или неправомерного использования учетных записей пользователей Web-ресурса.)

## Назначение, функции, принцип работы ключей защиты. Известные модели.

Ключ защиты (электронный ключ, англ. dongle) — аппаратное средство, предназначенное для защиты программного обеспечения (ПО) и данных от копирования, нелегального использования и несанкционированного распространения.

Аппаратные ключи защиты

* Содержат специализированную микросхему, либо защищённый от считывания микроконтроллер, имеющие уникальные для каждого ключа алгоритмы работы.
* Имеют защищённую энергонезависимую память небольшого объёма
* Могут иметь встроенный криптопроцессор (для аппаратной реализации шифрующих алгоритмов), часы реального времени.

Схема работы

1. Ключ присоединяется к определённому интерфейсу компьютера.
2. Защищённая программа через специальный драйвер отправляет ему информацию, которая обрабатывается в соответствии с заданным алгоритмом и возвращается обратно.
3. Если ответ ключа правильный, то программа продолжает свою работу.
4. В противном случае она может выполнять определенные разработчиками действия, например, переключаться в демонстрационный режим, блокируя доступ к определённым функциям.

Форм-факторы

* Чаще всего USB-устройства.
* Также встречаются с LPT- или PCMCIAинтерфейсами.
* Возможно впайка микросхемы на плату на этапе производства техники.

• Sentinel HL (ранее - HASP) • Guardant • SenseLock

## Виды защиты ПО с помощью электронных ключей. Методы взлома.

Виды защиты

* Автоматическая (с помощью специального ПО)
* С помощью API функций.

Автоматическая защита

* Автоматические инструменты (входящие в SDK), позволяющие защитить программу «за несколько кликов мыши».
* Файл приложения «оборачивается» в собственный код разработчика.
* Чаще всего код осуществляет проверку наличия ключа, контроль лицензионной политики, внедряет механизм защиты исполняемого файла от отладки и декомпиляции (например, сжатие исполняемого файла) и др.
* Для использования автоматического инструмента защиты не требуется доступ к исходному коду приложения. Такой механизм защиты не позволяет использовать весь потенциал электронных ключей и реализовать гибкую и индивидуальную защиту.

Защита с помощью API

* Самостоятельная разработка защиты, интегрируя систему защиты в приложение на уровне исходного кода.
* Библиотеки для различных языков.
* API - набор функций, предназначенных для обмена данными между приложением, системным драйвером и самим ключом.
* Операции с ключом через API: поиск, чтение и запись памяти, шифрование и расшифровывание данных при помощи аппаратных алгоритмов, лицензирование сетевого ПО и т. д.
* Обеспечивает высокий уровень защищённости приложений.
* Нейтрализовать защиту, встроенную в приложение, достаточно трудно вследствие её уникальности и «размытости» в теле программы.
* Необходимость изучения и модификации исполняемого кода защищенного приложения для обхода защиты является препятствием к ее взлому.

Обход защиты

Задача злоумышленника — заставить защищённую программу работать в условиях отсутствия легального ключа, подсоединённого к компьютеру. Злоумышленник имеет доступ ко всем открытым интерфейсам, документации, драйверам и может их анализировать на практике с привлечением любых средств.

При наличии ключа, злоумышленник может:

* перехватывать все обращения к ключу;
* протоколировать и анализировать эти обращения;
* посылать запросы к ключу и получать на них ответы;
* протоколировать и анализировать эти ответы;
* посылать ответы от имени ключа и др.

2 вида обхода защиты:

* Внесение изменений в программный модуль (взлом) (Злоумышленник исследует логику работы программы, анализируя код приложения, с целью выделения блока защиты и отключения его. Осуществляется с помощью отладки (или пошагового исполнения), дизассемблирования, декомпиляции и дампа оперативной памяти. Эти способы анализа исполняемого кода программы чаще всего используются злоумышленниками в комплексе.);
* Эмулирование наличия ключа путем перехвата вызовов библиотеки API для обмена с ключом. (Современные электронные ключи (например, ключи Guardant поколения Sign и современные ключи HASP HL) обеспечивают стойкое шифрование протокола обмена электронный ключ - библиотека API работы с ключом. В результате наиболее уязвимыми местами остаются точки вызовов функций этого API в приложении и логика обработки их результата.)

## КСЗИ Панцирь-К. Серверная и клиентские части, идентификация и аутентификация пользователей

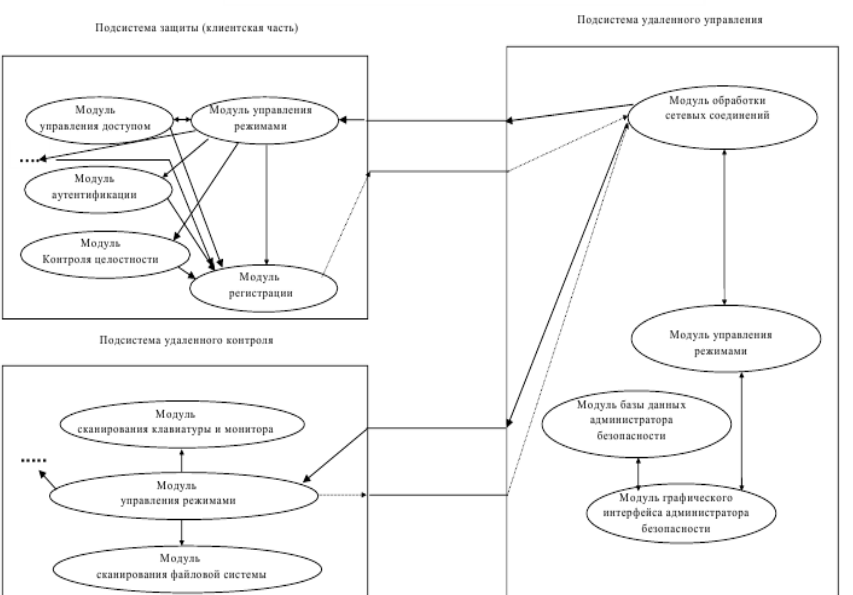
КСЗИ Панцирь-К

Возможности:

* Идентификация и аутентификация: Console, flash, eToken USB, …
* Разграничение и аудит действий пользователей и приложений
* Контроль целостности
* Гарантированное удаление
* Шифрование: 3DES, AES, DES, ГОСТ 28147-89

Общая схема КСЗИ

* Подсистема защиты рабочих станций и информационных серверов (клиент);
* Подсистема удаленного контроля рабочих станций и информационных серверов;
* Подсистема удаленного управления механизмами защиты рабочих станций и серверов (сервер).



Структура клиентской части

* Модуль аутентификации.
* Модуль управления доступом.
* Модуль управления подключением устройств.
* Модуль регистрации.
* Модуль контроля целостности.
* Модуль противодействия ошибкам и закладкам в системном и функциональном программном обеспечении.
* Модуль очистки памяти, кодирования и шифрования данных.
* Другие

Сервер безопасности

• Обеспечивает централизованное управление клиентской частью защиты КСЗИ.

* Модуль обработки сетевых соединений
* Модуль доступа базы данных
* Модуль графического интерфейса
* Модуль управления режимами

Сервер безопасности. Интерфейс

Представление на основе информации о пользователях:

* Отображение иерархической структуры пользователей на предприятии;
* Отображение функциональной принадлежности пользователей к АС;
* Отображение территориального расположения пользователей.

Профили пользователей

* Управление доступом к файловой системе, к разделяемым сетевым ресурсам, к съемным накопителям (дисковод, CD-ROM)
* Управление доступом к реестру ОС
* Управление доступом к Буферу Обмена
* Управление доступом к сетевым ресурсам по протоколу TCP/IP и др.
* Общие настроек клиентской части КСЗИ

## КСЗИ Панцирь-К. Контроль и разграничение доступа

**Разграничение доступа к ФС**

Виды файловых объектов:

• Файлы на жестком диске;

• Удаленные файловые объекты (разделенные в сети);

• Файловые объекты на внешних носителях.

**Основные принципы разграничения доступа к ФС**

• Только администратор может назначать (изменять) права доступа субъекта к объекту, такой сущности, как “Владелец” не существует;

• Две политики контроля доступа к ресурсам – разрешительная и запретительная;

• Права доступа назначаются субъектам, а не присваиваются объектам в качестве их атрибутов;

• Для любого субъекта доступа может быть реализована собственная разграничительная политика;

• Для каждого устанавливаемого типа доступа может быть использована собственная разграничительная политика «Разрешенные для…», либо «Запрещенные для…»;

• Типы доступа к ресурсам - «Чтение», «Запись», «Выполнение»;

• Объект, задается своим полнопутевым именем (логический диск, каталог, подкаталог, файл), либо используются маски;

• Разграничения действуют иерархически, права нижестоящего объекта наследуются у вышестоящего, если для него не заданы собственные права доступа;

• В качестве самостоятельных субъектов доступа в КСЗИ используются две сущности: «ПОЛЬЗОВАТЕЛЬ» и «ПРОЦЕСС». Права для субъекта процесс могут назначаться эксклюзивно и совместно с правами пользователя;

• Субъект доступа «ПРОЦЕСС» задается своим полнопутевым именем. Для задания объекта могут использоваться маски;

• Для прав доступа субъекта «ПРОЦЕСС» используется механизм наследования.

Существуют:

Разграничение доступа пользователей

Запретительная политика

Разграничение доступа процессов

Создание масок:

Процессы и объекты могут задаваться маской:

• \* - любая последовательность символов;

• ? - любой символ;

• [набор символов] - любой символ входящий в набор;

• [!набор символов] или

• [^набор символов] - любой символ не входящий в набор.

**Разграничение доступа к реестру**

Механизм полностью унифицирован с механизмом управления доступом к ФС:

• Разрешительная и запретительная разграничительные политики

• Разграничение для процессов и пользователей.

Отличие – типы доступа “Запись” и “Чтение”

## КСЗИ Панцирь-К. Аудит и дополнительные возможности

**Аудит событий ФС**

• Реализован механизм выборочной регистрации – регистрироваться доступ к объектам, задаваемым администратором безопасности. Для этого необходимо заполнить список контролируемых объектов.

• Все действия администратора безопасности по изменению прав доступа к рассматриваемым объектам защиты фиксируются и записываются в файле fileadm.log в каталоге Filectrl КСЗИ.

**Аудит событий доступа к реестру**

• Выборочный аудит - регистрируется доступ только к заданным объектам реестра ОС.

• Регистрация изменений прав доступа к реестру

**Аудит событий**

• КСЗИ осуществляет регистрацию основных событий при функционировании защищаемого объекта и реализует иерархическую модель аудита событий информационной безопасности системы.

**Уровни аудита**

1. Регистрация событий уровней, реализующих разграничительную политику доступа к ресурсам.
2. Регистрация событий уровня, реализующего контроль активности и корректности функционирования механизмов защиты, реализующих разграничительную политику доступа к ресурсам

**Типы регистрируемых событий**

1 Уровень:

• События, связанные с действиями пользователей по доступу к ресурсам защищаемой системы;

• События, связанные с действиями администратора безопасности по созданию и переназначению прав доступа пользователей к ресурсам защищаемой системы.

2 Уровень:

• События, связанные с некорректностью функционирования системы разграничения доступа (ошибки доступа)

**Информация о событии**

• Дата и время;

• Субъект, осуществляющий регистрируемое действие;

• Тип события;

• Результат выполнения и т.д.

**Журналы аудита**

• Журнал регистрации пользователей - регистрация входа и завершения работы пользователей в ОС

• Журнал входа пользователей (пароли) - регистрация всех (правильных и неправильных) попыток входа и смену действующего пароля пользователей в ОС

**Аудит доступа пользователей к ресурсам**

• Журнал аудита доступа к ФС

• Журнал доступа к реестру

• Журнал доступа к сети

• Журнал аудита доступа к принтерам

• Журнал подключения устройств

• Журнал доступа к Буферу Обмена

**Журналы аудита**

• Журнал запуска СЗИ/ОС - запуска сервиса КСЗИ и времени последнего цикла обработки событий • Журнал печати - реализует автоматическую регистрацию печати.

**Ротация журналов аудита**

• Программа ротации позволяет обрезать и очищать журналы, а также делать копии журналов перед (либо после) их очисткой.

• Программа содержит два файла sizelogs.exe и sizelogs.lcmd

**Дополнительные настройки**

• Пароль для снятия блокировки

• Разрешить Буфер Обмена между пользователями

• Не контролировать системное время

• Не использовать повторную регистрацию

• Очистка файла подкачки

**Сохранение и восстановление настроек**

• Тиражирование с помощью сервера

• Копирование настроек вручную

• Файлы настроек - pre.set, filectrl.ini, regctrl.ini, tcpctrl.ini, dirlink.ini, devctrl.ini, impctrl.ini, clipctrl.ini и printer.ini

**Дополнительная защита данных**

• Шифрование данных «на лету»

• Автоматическое гарантированное удаление остаточной информации

• Разграничение прав доступа к защищаемым объектам

• Скрытие защищаемых объектов файловой системы

**Шифрование данных «на лету»**

• Шифрование (расшифрование) данных при их сохранении в локальный или удаленный файловый объект.

• Поддерживаются файловые системы NTFS, FAT32 или FAT16.

• Алгоритмы шифрования: XOR, GOST, DES, 3DES, AES, ГОСТ 28147-89.

• СЗД позволяет подключать СКЗИ «Signal-COM CSP» и СКЗИ «КриптоПро CSP».

**Гарантированное удаление остаточной информации**

• КСЗИ позволяет осуществлять очистку оперативной и внешней памяти. Очистка производится путем записи маскирующей информации в память при ее освобождении (перераспределении).

• Для модифицируемых файловых объектов система позволяет задавать число «проходов очистки» и вид маскирующей информации, записываемой СЗД в файловый объект.

**Варианты автоматического запуска программ очистки оперативной и дисковой памяти**

• По расписанию.

• При входе (идентификации) нового пользователя

• Для реализации функций очистки памяти в состав дистрибутива КСЗИ включены две утилиты “delsec.exe” и “clearam.exe”

**Гарантированное удаление файлов с жёсткого диска**

• Программа FullDel.exe:

• Удаление файлов

• Очистка диска

**Скрытие защищаемых объектов файловой системы**

• Скрытие защищаемого файлового объекта;

• Скремблирование имени файлового объекта;

• Кодирование имени файлового объекта.

**Механизмы контроля целостности**

1. Контроль целостности каталогов и файлов данных (синхронный и асинхронный)

2. Контроль целостности исполняемых файлов (программ перед запуском)

3. Контроль целостности файлов КСЗИ

**Дополнительные механизмы контроля печати**

• Механизм маркировки документов Автоматический ввод реквизитов документов при печати из программы MS Word (другие программы печати в этом режиме средствами КСЗИ запрещены)

• Механизм теневого копирования любых печатных документов Автоматическое копирование документов при печати из любых программ.

## Управление криптографическими ключами

Основной международный стандарт – ISO/IEC 11770 – Key management :

• Управление ключами - совокупность процедур и процессов, сопровождающих жизненный цикл ключей в криптосистеме.

Цель управления ключами – обеспечение безопасности криптографических ключей на всех этапах жизненного цикла безопасности всей криптосистемы.

Секретные ключи - необходимо обеспечить секретность, подлинность, целостность:

* Общие секретные ключи симметричных криптосистем;
* Частные секретные ключи асимметричных криптосистем (закрытые ключи).

Открытые ключи - необходимо обеспечить подлинность, целостность:

• Открытые ключи асимметричных криптосистем, помещаемые в общедоступные справочники.

Жизненный цикл ключей

* Все криптосистемы, за исключением простейших, в которых используемые ключи зафиксированы раз и навсегда, нуждаются в периодической замене ключей.
* Эта замена проводится с помощью определенных процедур и протоколов, в ряде которых используются и протоколы взаимодействия с третьей стороной.
* Последовательность стадий, которые проходят ключи от момента установления до следующей замены, называется жизненным циклом ключей.

1. Регистрация пользователя (обмен первоначальной информацией)
2. Инициализация (подключение программно-аппаратных средств)
3. Генерация ключей
4. Установка ключей
5. Регистрация ключей
6. Обычный режим работы
7. Хранение ключа (на токенах, смарт-картах и т.д.)
8. Замена ключей (при истечении срока, компрометации и т.д.)
9. Архивирование (если нужно)
10. Уничтожение ключей
11. Восстановление ключей
12. Отмена ключей (прекращение работы ключа ДО истечения срока действия)

Транспортировка - самый опасный этап !

* Для секретных ключей симметричных криптосистем главная цель – предотвратить попадание ключи к посторонним лицам: традиционные меры физической защиты, усиленные аппаратными и организационными мерами.
* Для открытых ключей главная цель – обеспечить подлинность и целостность сложная задача, которая решается созданием инфраструктуры открытых ключей.

Инфраструктура открытых ключей (PKI – Public Key Infrastructure) – универсальная модель организованной поддержки криптографических средств защиты информации в крупномасштабных компьютерных системах в соответствии с принятыми в них политиками безопасности, которая реализует управление криптографическими ключами на всех этапах их жизненного цикла, обеспечивая взаимодействие всех средств защиты

## Концепция иерархии ключей, генерация ключей.

Концепция иерархии ключей

* Любая информация об используемых ключах должна быть защищена - храниться в зашифрованном виде.
* Необходимость в хранении и передаче ключей, зашифрованных с помощью других ключей, приводит к концепции иерархии ключей.
* В стандарте ISО 8532 (Ваnking-Кеу Маnаgement) подробно изложен метод главных сеансовых ключей (master/session keys).
* Суть метода состоит в том, что вводится иерархия ключей: главный ключ (ГК), ключ шифрования ключей (КК), ключ шифрования данных (КД)

Иерархия ключей может быть:

* двухуровневой (КК/КД);
* трехуровневой (ГК/КК/КД).

Концепция иерархии ключей

* Нижний уровень: рабочие или сеансовые КД. (Применяются для шифрования данных, персональных идентификационных номеров (РIN) и аутентификации сообщений.)
* Ключи шифрования ключей (КК) (Используют для шифрования ключей с целью защиты при передаче или хранении. Никогда не должны использоваться как сеансовые КД.)
* Верхний уровень - главный ключ или мастер- ключ (ГК) (Шифрует КК, если требуется сохранить его на диске. Обычно в одной системе используется только один мастер-ключ. Для исключения перехвата мастер-ключ распространяется между участниками обмена неэлектронным способом. Значение мастер-ключа сохраняется длительное время (до нескольких недель или месяцев). Мастер-ключ компьютера создается случайным выбором из всех возможных значений ключей и помещается в защищенный от считывания и записи блок криптографической системы)

Сроки действия ключей

Срок действия ключа означает промежуток времени, в течение которого он может быть использован доверенными сторонами.

Сроки действия ключей

Выделяют:

* Ключи с длительным сроком действия. К ним относится главный ключ, часто — ключи для шифрования ключей.
* Ключи с коротким сроком действия. К ним относятся ключи для шифрования данных

Генерация случайных чисел

* Нужны для генерации ключей шифрования, должны удовлетворять определенным требованиям.
* Генератор псевдослучайных чисел.
* Аппаратный генератор случайных чисел.

Генератор псевдослучайных чисел — алгоритм, порождающий последовательность чисел, элементы которой почти независимы друг от друга и подчиняются заданному распределению (обычно равномерному).

Достоинства:

* Требуется однократная проверка;
* Многократная воспроизводимость последовательности чисел;
* Мало места в памяти и нет внешних устройств.

Недостатки:

* Запас чисел ограничен периодом последовательности;
* Затраты машинного времени.

Хранение ключей

* Важные ключи нельзя хранить локально -> возможна утечка ключа.

Возможно применение внешних устройств для хранения ключей и выполнения криптографических операций:

* eToken и подобные;
* Аппаратные модули безопасности.

1. Аппаратные модули безопасности (HSM)

Аппаратный модуль безопасности (hardware security module - HSM) – вычислительное устройство, которое обеспечивает безопасность и управляет криптографическими ключами, используемыми для усиленной аутентификации и криптографических функций.

Функции:

• Безопасная генерация ключей шифрования

• Безопасное хранение и управление ключами

• Работа с зашифрованной и конфиденциальной информацией

• Работа с симметричной и асимметричной криптографией.

Применение:

• PKI, центр сертификации

• Банковские операции

• Установление SSL соединений

Безопасный криптопроцессор - это система на кристалле или микропроцессор, предназначенный для проведения криптографических операций и обеспеченный мерами физической защиты, дающими ему некоторую возможность противодействия несанционированному доступу.

В отличие от криптографических процессоров, "доверяющих" шине и выводящих незашифрованные данные на нее, как будто она находится в защищенной среде, безопасный криптопроцессор не выводит незашифрованные данные или незашифрованные программные инструкции в среду, которая не может быть гарантированно защищенной все время.

Особенности:

• Обнаружение подделок и индикация вскрытия.

• Проводящие защитные слои в чипе, мешающие считывать внутренние сигналы.

• Контролируемое исполнение, чтобы предотвратить раскрытие любой секретной информации по временным задержкам.

• Автоматическое обнуление секретов в случае фальсификации.

• Доверенный загрузчик - проверяет подлинность операционной системы перед ее запуском.

• Доверенная операционная система - проверяет подлинность приложений перед их запуском.

• Аппаратные регистры, где реализуется модель с разделением привелегий.

• Может иметь несколько уровней физической защиты в одном чипе криптопроцессоре.

• Чип криптопроцессора может быть помещен аппаратный модуль безопасности наряду с другими процессорами и памятью, где хранятся и обрабатываются зашифрованные данные.

• Любая попытка извлечь его вызовет обнуление ключей в крипточипе.

• Аппаратные модули безопасности также могут быть частью компьютера (например банкомата), который проводит операции внутри запертого сейфа, чтобы предотвратить кражи, замены и подделки.

1. Концепция доверенных сеансов связи.. Комплекс «МАРШ!», «М!&М».

• Клиент ДСС • Сервер ДСС

Режим доверенной среды обеспечивает защищённый ввод данных поставщиками данных, а также защищённое получение данных потребителями

Средство обеспечения доверенного сеанса связи (СОДС) «МАРШ!» предназначено для организации защищённой работы удалённых пользователей недоверенных компьютеров с сервисами доверенной распределенной информационной системы (РИС) через сети передачи данных в рамках доверенного сеанса связи (ДСС).

Клиент ДСС

•Загрузочное USB-устройство с собственным микропроцессором

• Аппаратно разделённые области памяти с однократно назначаемыми атрибутами доступа

• Загрузочная ОС, СКЗИ и набор прикладного ПО в защищённой от записи области памяти

Сервер ДСС

• Доверенный сервер

• Создание и работа защищённых сетевых соединений с Клиентами ДСС

• Переключение Клиентов ДСС на сервисы защищаемых ИР в соответствии с правами пользователей

Преимущества «МАРШ!»

• Необходимый функционал и достаточный уровень защиты доступа пользователя к защищаемым информационным ресурсам

• Мобильность, готовность к работе на произвольном (в т.ч. недоверенном) компьютере

• Отсутствие ограничений на работу пользователя с компьютером вне режима доверенного сеанса связи

• Приемлемая стоимость

**М!&М** • Добавлен модем

Используется мобильная рабочая станция; • Используется РС, для которой политикой безопасности запрещено подключение к сети; • Рабочая станция эксплуатируется в условиях, в которых уровень развития сетевой инфраструктуры не позволяет иметь собственные средства связи.

1. Защищенные микрокомпьютеры «MKT». Назначение, функции.

• Память, в которой размещается ОС, переводится в режим «только чтение» • Этот обеспечивает неизменность ОС аппаратным способом, никакие программные действия злоумышленника не могут нарушить целостность, а, следовательно, доверенность программной среды.

Из доверенной среды нельзя выходить в недоверенную • Для этого применяются отдельные ОС • Взаимовлияние защищенной и незащищенной ОС друг на друга исключено, так как они размещены в физически разделенным банках памяти и ни при каких обстоятельствах не могут быть запущены одновременно.

• MKT –предназначен только для работы в защищенном режиме. • MKTrusT – предназначен для работы в одном из двух режимов на выбор – защищенном или обычном, без ограничений.

MKT-card – терминал, состоящий из стационарной док-станции, к которой подключается периферия, и отчуждаемого мобильного устройства, которое является носителем всей персонифицированной части информационной среды клиентского рабочего места.

TrusTPad – планшетный компьютер, работающий аналогично MKTrusT – с возможностью выбора одного из режимов с помощью переключателя.

TrusTPhone – IP-телефон, также построенный на «гарвардской» архитектуре по логике MKTrusT. Фактически, это TrusTPad функцией IP-телефонии, выполненный в виде стационарного телефона с ЖК-дисплеем и сенсорным номеронабирателем.

1. Защищенные носители «СЕКРЕТ». Виды, назначение, функции.

«Секрет» — это специальный носитель информации, который имеет независимые от компьютера, к которому он будет подключаться, механизмы принятия решения о том, на каких компьютерах с ним можно работать

«Личный секрет»

«Секрет фирмы» — это корпоративное решение, включающее в себя помимо необходимого числа «Секретов» также сервера аутентификации и регистрации, оборудованные средствами защиты информации обеспечивающими адекватный масштабу сети уровень защищенности. ПО сервера аутентификации выполняется на обычном ПК, выделенном для этой цели, поэтому для обеспечения доверенной среды вычислений нужно установить на этот ПК ПАК Аккорд, который будет обеспечивать доверенную загрузку ОС и контролировать доступ к серверу аутентификации.

«Секрет особого назначения» - предназначен для сотрудников, в сферу ответственности которых входит работа с данным, конфиденциальность которых критична, но которые, вместе с тем, должны храниться на служебном носителе и переноситься сотрудником в рамках его должностных обязанностей на различные компьютеры (а не только между зафиксированными администратором системы в списке разрешенных рабочих мест). В аппаратном журнале фиксируются все попытки работы с ним на различных ПК, вне зависимости от того, была ли попытка успешной. • Отредактировать журнал нельзя.

«Идеальный токен» - предназначен для хранения ключевой информации. • Производится контроль подключения только с рабочего места.

«ПАЖ» - средство архивирования журналов событий • Запрещает редактирование журналов. • Данные можно только добавлять (Add Only), и настроить на работу с одним или несколькими конкретными ПК, чтобы журналы не путались. • Можно настроить сбор журналов из разных источников.