

Вопросы к экзамену по дисциплине «Защита информации»
для студентов 4 курса групп ИВТ-41 и ИВТ-42

1. Информационная безопасность (ИБ) – основные понятия и определения, дайте определения следующим понятиям: информация, безопасность, субъекты и объекты безопасности, угроза, уязвимость, атака.

В широком смысле понятие ИБ определяется как ИБ человека, общества и государства. В узком смысле – это безопасность самой информации и каналов ее приема-передачи, а также организация защиты от применения информационного оружия.

Информация – сведения, воспринимаемые субъектом в форме знаний.

Информация в современном обществе – это ресурс.

Безопасность – защищенность ресурса от определенного множества угроз.

*Явление – это состояние защищенности ресурса.

*Процесс – деятельность по защите ресурса от заданного множества угроз.

Субъекты информационных отношений: владелец информационного ресурса, потребитель информационного ресурса.

Угроза – потенциально возможное действие или событие, которое может привести к нарушению безопасности – нанести неприемлемый ущерб субъектам информационных отношений

Уязвимость – свойство информационной системы, предоставляющее возможность реализации угроз безопасности обрабатываемой в ней информации (потенциально слабое место, ошибка в проектировании или реализации системы, которая может привести к неожиданным и нежелательным последствиям нарушения ИБ).

Атака – попытка покушения на безопасность системы. Действие, которое может нарушить безопасность системы.

2. Информационная безопасность. Дайте определения составляющим информационной безопасности: конфиденциальность (Confidentiality), целостность (Integrity), доступность (Availability).

Составляющие ИБ:

Конфиденциальность – свойство информации, которое заключается в неизвестности информации третьим лицам (т.е. тем, кто не имеет доступа к информации на законном основании)

Целостность – свойство информации, которое заключается в непротиворечивости информации, в невозможности несанкционированной модификации информации.

Виды:

* Статическая – неизменность информации в процессе хранения

* Динамическая – корректное выполнение транзакции, корректная передача информации по каналам связи.

Доступность – возможность для законных пользователей за приемлемое время получить доступ к информации, информационной услуге.

3. Информационная безопасность. Дайте определения составляющим информационной безопасности: идентификация и аутентификация (Identification Authentication), неотказуемость (Non-repudiation), подотчетность (Accountability).

Идентификация – процесс, с помощью которого можно отличить один

объект/субъект от другого. Это присвоение уникальных меток объектам/субъектам с целью отличить один от другого

Аутентификация – определение подлинности субъекта/объекта.

Неотказуемость – невозможность отрицать факт совершения действия, участия субъекта в транзакции.

Подотчетность – фиксация всех событий в ИС(?), имеющих смысл с точки зрения безопасности.

Цель: обеспечить базу фактов для проведения расследования возможных инцидентов, связанных с нарушением безопасности.

4. Перечислите виды угроз информационной безопасности и дайте краткое их определение.

Виды угроз:

Угроза конфиденциальности – потенциально возможное действие или событие, которое может привести к нарушению конфиденциальности информации.

Угроза целостности – потенциально возможное действие или событие, которое может привести к нарушению целостности информации (повреждение, уничтожение).

Угроза доступности – потенциально возможное действие или событие, в результате которого система может быть недоступна для законных пользователей в течение длительного периода времени (превышающего установленную норму задержки).

Угрозы могут быть так же преднамеренными и непреднамеренными, пассивными и активными, естественными и искусственными и др.

5. Проведите классификацию угроз ИБ по природе возникновения и по степени преднамеренности.

По природе возникновения:

1. Естественные (угрозы, вызванные воздействиями на ИС и ее компоненты объективных физических процессов или стихийных природных явлений, независимых от человека).
2. Искусственные (угрозы информационной безопасности ИС, вызванные деятельностью человека).

По степени преднамеренности:

1. Непреднамеренные (ошибки программно-аппаратных средств ИС, неумышленная порча носителей информации, ввод ошибочных данных, неумышленное повреждение каналов связи, пересылка данных по ошибочному адресу абонента (устройства) и т.д.).
2. Преднамеренные (умышленные действия - действия злоумышленника для хищения информации).

6. Проведите классификацию угроз ИБ по источнику угроз и положению источника угроз.

По источнику угроз:

1. Природа (стихийные бедствия, магнитные бури, радиоактивное излучение и т.п.).
2. Человек (внедрение агентов в число персонала системы, вербовка персонала или отдельных пользователей, разглашение, передача или утрата атрибутов разграничения доступа).

3. Санкционированные программные/аппаратные средства (запуск технологических программ, способных при некомпетентном использовании вызывать потерю работоспособности системы (зависания или за цикливания) или необратимые изменения в системе (форматирование или реструктуризацию носителей информации, удаление данных и т. п.), возникновение отказа в работе операционной системы).
4. Несанкционированные программные/аппаратные средства (заражение компьютера вирусами с деструктивными функциями, нелегальное внедрение и использование неучтенных программ с последующим необоснованным расходом ресурсов (загрузка процессора, захват оперативной памяти и памяти на внешних носителях)).

По положению источника угроз:

1. Вне контролируемой зоны (дистанционная фото- и видеосъемка, перехват данных, передаваемых по каналам связи, и их анализ с целью выяснения протоколов обмена, правил вхождения в связь и авторизации пользователя и последующих попыток их имитации для проникновения в систему, перехват побочных электромагнитных, акустических и других излучений устройств и линий связи).
2. В пределах зоны (применение подслушивающих устройств, отключение или вывод из строя подсистем обеспечения функционирования вычислительных систем (электропитания, охлаждения и вентиляции, линий связи и т.д.)).

7. Проведите классификацию угроз ИБ по степени зависимости от активности ИС и степени воздействия на ИС.

По степени зависимости от активности ИС:

1. Независимо от активности (вскрытие шифров, хищение носителей информации (магнитных дисков, лент, микросхем памяти, запоминающих устройств и компьютерных систем)).
2. Только в процессе активности – *угрозы, которые могут проявляться только в процессе автоматизированной обработки данных* (угроза выполнения и распространения вируса).

По степени взаимодействия на ИС:

1. Пассивные – *при реализации ничего не меняют в структуре и содержании ИС* (угроза копирования секретных данных).
2. Активные – *при воздействии вносят изменения в структуру и содержание ИС* (угроза умышленной модификации информации, внедрение аппаратных спецвложений, программных "закладок" и "вирусов" ("троянских коней" и "жучков"), действия по дезорганизации функционирования системы (изменение режимов работы устройств или программ, забастовка, саботаж персонала и т.д.)).

8. Проведите классификацию угроз ИБ по этапам доступа пользователей к ресурсам и по способу доступа к ресурсам ИС.

По этапам доступа пользователей к ресурсам:

1. Угрозы, проявляющиеся на этапе доступа к ресурсам (например, угрозы несанкционированного доступа в ИС).
2. Угрозы, проявляющиеся после разрешения доступа к ресурсам (угрозы несанкционированного или некорректного использования ресурсов ИС).

По способу доступа к ресурсам ИС:

1. Стандартный путь получения доступа к ресурсам (незаконное получение паролей и других реквизитов разграничения доступа с последующей маскировкой под зарегистрированного пользователя; несанкционированное использование терминалов пользователей, имеющих уникальные физические характеристики, такие как номер рабочей станции в сети, физический адрес, адрес в системе связи, аппаратный блок кодирования и т.п.)
2. Нестандартный путь получения доступа к ресурсам (вход в систему в обход средств защиты; угроза доступа к ресурсам ИС путем использования недокументированных возможностей ИС).

Основы информационной безопасности. Часть 1: Виды угроз [Основы информационной безопасности. Часть 1: Виды угроз / Блог компании VPS.house / Хабр](#)

9. Этапы развития информационной безопасности. Расскажите о развитии информационной безопасности до 1946 года

I этап — до 1816 года — характеризуется использованием естественно возникавших средств информационных коммуникаций. В этот период основная задача информационной безопасности заключалась в защите сведений о событиях, фактах, имуществе, местонахождении и других данных, имеющих для человека лично или сообщества, к которому он принадлежал, жизненное значение.

II этап — начиная с 1816 года — связан с началом использования искусственно создаваемых технических средств электро- и радиосвязи. Для обеспечения скрытности и помехозащищенности радиосвязи необходимо было использовать опыт первого периода информационной безопасности на более высоком технологическом уровне, а именно применение помехоустойчивого кодирования сообщения (сигнала) с последующим декодированием принятого сообщения (сигнала).

III этап — начиная с 1935 года — связан с появлением радиолокационных и гидроакустических средств. Основным способом обеспечения информационной безопасности в этот период было сочетание организационных и технических мер, направленных на повышение защищенности радиолокационных средств от воздействия на их приемные устройства активными маскирующими и пассивными имитирующими радиоэлектронными помехами.

10. Этапы развития информационной безопасности. Расскажите о развитии информационной безопасности начиная с 1946 года.

IV этап — начиная с 1946 года — связан с изобретением и внедрением в практическую деятельность электронно-вычислительных машин (компьютеров). Задачи информационной безопасности решались, в основном, методами и способами ограничения физического доступа к оборудованию средств добывания, переработки и передачи информации.

V этап — начиная с 1965 года — обусловлен созданием и развитием локальных информационно-коммуникационных сетей. Задачи информационной безопасности также решались, в основном, методами и способами физической защиты средств добывания, переработки и передачи информации, объединенных в локальную сеть

путём администрирования и управления доступом к сетевым ресурсам.

VI этап — начиная с 1973 года — связан с использованием сверхмобильных коммуникационных устройств с широким спектром задач. Угрозы информационной безопасности стали гораздо серьезнее. Образовались сообщества людей — хакеров, ставящих своей целью нанесение ущерба информационной безопасности отдельных пользователей, организаций и целых стран. Информационный ресурс стал важнейшим ресурсом государства, а обеспечение его безопасности — важнейшей и обязательной составляющей национальной безопасности. Формируется информационное право — новая отрасль международной правовой системы.

VII этап — начиная с 1985 года — связан с созданием и развитием глобальных информационно-коммуникационных сетей с использованием космических средств обеспечения. Можно предположить что очередной этап развития информационной безопасности, будет связан с широким использованием сверхмобильных коммуникационных устройств с широким спектром задач и глобальным охватом в пространстве и времени, обеспечиваемым космическими информационно-коммуникационными системами. Для решения задач информационной безопасности на этом этапе необходимо создание макросистемы информационной безопасности человечества под эгидой ведущих международных форумов.

Лекция 4 по ИБ. Этапы развития информационной безопасности (**9-10 вопросы**) <http://uskov.info/lektsii-po-informatsionnoj-bezopasnosti/lektsiya-4-po-ib-e-tapy-razvitiya-ib/>

11-13. Защита информации от копирования.

Средства защиты бывают формальные и неформальные:

Неформальные средства защиты — это всевозможные правила, морально-этические средства, нормативные средства, законы и т.д.

Формальные – это специальные технические средства и программное обеспечение

Разграничение помогает разграничить зоны ответственности при создании систем информационной безопасности, так как при общем руководстве защитой административный персонал реализует нормативные способы, а IT специалисты технические.

Основы информационной безопасности предполагают разграничение полномочий не только в части использования информации, но и в части работы с её охраной. Подобное разграничение полномочий требует и нескольких уровней контроля.

11. Защита информации от копирования. Расскажите об организационных, юридических и физических мерах.

Организационные – препятствия при входе в область данных (идентификация):

- Полное резервное копирование и восстановление систем
- Резервное копирование отдельных файлов и папок
- Автоматическое восстановление
- Резервное копирование на виртуальную машину или сетевой диск
- Центральная консоль управления задачами

Неформальные средства защиты группируются на **нормативные**,

административные и морально-этические. На первом уровне защиты находятся нормативные средства, регламентирующие ИБ в качестве процесса в деятельности организации. Нормативные средства обеспечения ИБ представлены законодательными актами и нормативно распорядительными документами, которые действуют на уровне организации. В мировой практике при разработке нормативных средств ориентируются на стандарты защиты, основным из которых является документ ISO IEC 27000-2016. Данный стандарт предполагает опробованные методики, необходимые для внедрения ИБ. Авторы этих методик считают, что основа информационной безопасности заключается в системности и последовательной реализации всех этапов от разработки до постконтроля. Для получения сертификата, который подтверждает соответствия стандартам по обеспечению ИБ необходимо внедрить все рекомендуемые методики в полном объеме. Если нет необходимости получать сертификат в качестве базы для разработки собственных систем ИБ, допускается принять любую из более ранних версий стандарта или российских ГОСТов, имеющих рекомендательный характер.

Юридические:

По итогам изучения стандарта разрабатываются два документа, которые касаются безопасности информации. Основной, но менее формальный документ – это концепция ИБ предприятия, которая определяет меры и способы внедрения системы ИБ для информационных систем организации. Второй документ, который обязаны исполнять все сотрудники компании – это положение об ИБ, утверждаемое на уровне совета директоров или исполнительного органа. Кроме положения на уровне компании должны быть разработаны перечни сведений, составляющих коммерческую тайну, приложения к трудовым договорам, закрепляющие ответственность за разглашение конфиденциальных данных, а также другие стандарты и методики. Внутренние нормы и правила должны содержать механизмы реализации и меры ответственности. Чаще всего эти меры носят дисциплинарный характер.

Физические меры

Физические меры обеспечивают ограничение физического доступа к компьютеру, линии связи, телекоммуникационному оборудованию и контроль доступа.

- охрану периметра, территории, помещений
- визуальное и видеонаблюдение
- опознавание людей и грузов
- идентификацию техники
- сигнализацию и блокировку
- ограничение физического доступа в помещения.

К **организационным и административным** мерам по защите информационной безопасности относятся:

- архитектурно-планировочные решения, позволяющие защитить переговорные комнаты и кабинеты руководства от прослушивания

- установление различных уровней доступа к информации
- организация охраны и пропускного режима
- сертификация деятельности компании по международным стандартам
- сертификация отдельных аппаратно-программных комплексов
- аттестация субъектов и объектов на соответствие необходимым требованиям безопасности
- получение лицензий, необходимых для работы с защищенными массивами информации
- оформление системы запросов на допуск к интернету, внешней электронной почте и другим ресурсам
- получение электронной цифровой подписи для усиления безопасности финансовой и другой информации, которую передают государственным органам по каналам электронной связи.

А также:

- разработку политики безопасности применительно к конкретной информационной системе (какие профили, какие пароли, какие атрибуты, какие права доступа);
- разработку средств управления безопасностью (кто, когда и в каком порядке изменяет политику безопасности);
- распределение ответственности за безопасность (кто и за что отвечает при нарушении политики безопасности);
- обучение персонала безопасной работе и периодический контроль за деятельностью сотрудников;
- контроль за соблюдением установленной политики безопасности;
- разработку мер безопасности на случай природных или техногенных катастроф и террористических актов.

12. Защита информации от копирования. Расскажите об аппаратных мерах защиты информации.

Аппаратные средства защиты информации – это любые электрические, электронные, оптические, лазерные и другие устройства, которые встраиваются в информационные и телекоммуникационные системы: специальные компьютеры, системы контроля сотрудников, защиты серверов и корпоративных сетей. Они препятствуют доступу к информации, в том числе с помощью ее маскировки.

К аппаратным средствам относятся:

- генераторы шума
- сетевые фильтры
- сканирующие радиоприемники и множество других устройств, «перекрывающих» потенциальные каналы утечки информации или позволяющих их обнаружить

Наибольшее распространение получают следующие:

- специальные регистры для хранения реквизитов защиты: паролей, идентифицирующих кодов, грифов или уровней секретности
- устройства измерения индивидуальных характеристик человека (голоса, отпечатков) с целью его идентификации

- схемы прерывания передачи информации в линии связи с целью периодической проверки адреса выдачи данных
- устройства для шифрования информации (криптографические методы)
- модули доверенной загрузки компьютера

Аппаратные меры защиты от копирования:

- наличие аппаратных ключей;
- запись информации в неиспользуемых секторах;
- проверка расположения и содержимого «сбойных» секторов;
- проверка скорости чтения отдельных секторов;
- техподдержка (косвенная защита).

Все средства, гарантирующие безопасность информации, должны использоваться в совокупности после предварительной оценки ценности информации и сравнении ее со стоимостью ресурсов, затраченных на охрану. Поэтому предложения по использованию средств защиты должны формулироваться уже на этапе разработки систем, а утверждение должно производиться на том уровне управления, который отвечает за утверждение бюджетов.

13. Защита информации от копирования. Расскажите о программных мерах защиты информации.

Программные средства – это пользовательские и системные программы, предназначенные для решения частных и комплексных задач, связанных с обеспечением ИБ. Такие комплексные системы могут служить для предотвращения утечки информации, ее переформатирования и перенаправления информационных потоков, а также обеспечивают защиту от инцидентов в сфере ИБ.

Программные средства защиты информации — это специальные программы и программные комплексы, предназначенные для защиты информации в информационной системе.

Программные средства включают программы для идентификации пользователей, контроля доступа, удаления остаточной (рабочей) информации типа временных файлов, тестового контроля системы защиты и другие. Преимущества программных средств – универсальность, гибкость, надежность, простота установки, способность к модификации и развитию.

К программным средствам защиты программного обеспечения относятся:

- встроенные средства защиты информации – это средства, реализующие авторизацию и аутентификацию пользователей (вход в систему с использованием пароля), разграничение прав доступа, защиту ПО от копирования, корректность ввода данных в соответствии с заданным форматом и так далее
- специализированные программные средства защиты информации от несанкционированного доступа:
 - защита папок и файлов на компьютере
 - контроль за выполнением пользователями правил безопасности при работе с компьютером
 - выявление и пресечение попыток несанкционированного доступа к конфиденциальным данным
 - наблюдение за действиями, происходящими на контролируемом компьютере, работающем автономно или в локальной вычислительной сети
- антивирусные программы
- межсетевые экраны
- прокси-серверы и VPN

Программные средства защиты ПО от копирования:

- установка программ;
- регистрационный код;
- защита CD, DVD;
- невозможность работы без диска;
- сканирование сети;
- информация об устройстве
- учетные записи
- проверка серийных номеров в интернете.

Все средства, гарантирующие безопасность информации, должны использоваться в совокупности после предварительной оценки ценности информации и сравнении ее со стоимостью ресурсов, затраченных на охрану. Поэтому предложения по использованию средств защиты должны формулироваться уже на этапе разработки систем, а утверждение должно производиться на том уровне управления, который отвечает за утверждение бюджетов.

14. Что такое система защиты программы от копирования?

Защита от копирования заключается в предупреждении возможностей несанкционированного снятия копии с информации, находящейся в оперативной памяти компьютера или на каком-либо диске, в целях злоумышленного ее использования.

Под **системой защиты программ от копирования** понимается система, которая обеспечивает выполнение ею своих функций только при опознании некоторого уникального не поддающегося копированию элемента, называемого ключевым.

Системы защиты от копирования можно разделить на следующие группы:

1. Защита программы с использованием стандартного носителя.
2. Защита при помощи электронных ключей. Принцип действия электронных ключей: ключ присоединяется к определенному интерфейсу компьютера. Далее защищённая программа через специальный драйвер отправляет ему информацию, которая обрабатывается в соответствии с заданным алгоритмом и возвращается обратно.
3. Использование подхода SaaS, то есть переноса кода самих программ в облако и предоставление функционала этих программ как сервиса.
4. Обфускация кода приложения
5. Использование механизмов активации программного обеспечения, когда программа привязывается к железу компьютера, а разработчик генерирует код активации.
6. Отдельные средства защиты непосредственно кода приложения от копирования и использования в других программах.

Примеры 12-13 вопросы (списки с •)

К **основным функциям**, которые выполняют **системы защиты программ от копирования**, относятся в следующем:

1. **Идентификация** – присвоение индивидуального трудно подделываемого признака той среды, из которой будет запускаться защищаемая программа.
2. **Аутентификация** – опознавание той среды, из которой поступает запрос на копирование защищаемой программы.
3. Регистрация санкционированного копирования.
4. Реагирование на попытки несанкционированного копирования.
5. Противодействие изучению алгоритмов работы системы защиты.

Методы реализации способа **идентификации**:

1. Нарушение последовательности секторов флэш-накопителя;
2. Изменение межсекторной дистанции;
3. Форматирование с разным кодом длины 0 или 1.

Трассировка - пошаговое выполнение программы

Реагирование на попытки несанкционированного копирования:

1. Отказ в исполнении запроса;
2. Предупреждение злоумышленника о более серьезных санкциях;
3. Уничтожение защищаемой программы (после первой попытки или после нескольких попыток).

Противодействие изучению алгоритмов работы системы защиты предусмотрено для того, чтобы воспрепятствовать злоумышленнику в изучении структуры и содержания реализованной на дискете системы защиты в целях её преодоления (нейтрализации).

Для обнаружения модифицированного кода традиционно применялись следующие методы:

1. Подсчет контрольных сумм критических участков
2. Использование контрольной суммы всего кода для расшифровки некоторого фрагмента.
3. Многопроходная расшифровка кода с ключом, вычисляемым на основе контрольной суммы всего кода либо критического участка.
4. Использование корректирующих кодов, позволяющих определить местоположение контрольного байта.
5. Контроль времени выполнения критического участка по сравнению с эталонным временем.
6. Контроль относительного времени выполнения участка программы (относительно другого участка).

15. Политика и модели информационной безопасности.

Под **политикой безопасности** понимается совокупность норм и правил, регламентирующих процесс обработки информации, выполнение которых обеспечивает защиту от определенного множества угроз и составляет необходимое условие безопасности системы. Формальное выражение политики безопасности называют моделью безопасности.

Основная цель создания политики безопасности системы и описания ее в виде формальной модели — это определение условий, которым должно подчиняться поведение системы, выработка критерия безопасности и проведение формального доказательства соответствия системы этому критерию при соблюдении установленных правил и ограничений.

Кроме того, формальные модели безопасности позволяют решить еще целый ряд задач, возникающих в ходе проектирования, разработки и сертификации защищенных систем, поэтому их используют не только теоретики информационной безопасности, но и другие категории специалистов, участвующих в процессе создания и эксплуатации защищенных информационных систем (производители, потребители, эксперты).

Модели информационной безопасности:

1. Модель дискреционного доступа (DAC)

В рамках дискреционной модели контролируется доступ субъектов (пользователей или приложений) к объектам (представляющим собой различные информационные ресурсы: файлы, приложения, устройства вывода и т.д.).

Для каждого объекта существует субъект-владелец, который сам определяет тех, кто имеет доступ к объекту, а также разрешенные операции доступа. Основными операциями доступа являются READ (чтение), WRITE (запись) и EXECUTE (выполнение, имеет смысл только для программ). Таким образом, в модели дискреционного доступа для каждой пары субъект-объект устанавливается набор разрешенных операций доступа.

При запросе доступа к объекту, система ищет субъекта в списке прав доступа объекта и разрешает доступ если субъект присутствует в списке и разрешенный тип доступа включает требуемый тип. Иначе доступ не предоставляется.

Классическая система дискреционного контроля доступа является «закрытой» в том смысле, что изначально объект не доступен никому, и в списке прав доступа описывается набор разрешений. Также существуют «открытые» системы, в которых по умолчанию все имеют полный доступ к объектам, а в списке доступа описывается набор ограничений.

Такая модель реализована в операционных системах Windows (см. рис. 1) и Linux.

В частности, в Linux для каждого файла (все ресурсы в ОС Linux представимы в виде файлов, в том числе устройства ввода-вывода) устанавливаются разрешения доступа для трех категорий субъектов: владелец файла, члены той же группы, что и владелец, и все остальные пользователи. Для каждой из этих категорий устанавливаются права на чтение (r), запись (w) и выполнение (x). Набор прав доступа объекта может быть представлен в виде символьной строки. Например, запись «гwxг-хг--» означает, что владелец файла может делать с ним все, что угодно; члены его группы могут читать и исполнять файл, но не могут записывать, а прочим пользователям доступно только чтение.

Недостаток модели DAC заключается в том, что субъект, имеющий право на чтение информации может передать ее другим субъектам, которые этого права не имеют, без уведомления владельца объекта. Таким образом, нет гарантии, что информация не станет доступна субъектам, не имеющим к ней доступа. Кроме того, не во всех АИС каждому объекту можно назначить владельца (во многих случаях данные принадлежат не отдельным субъектам, а всей системе).

2. Модель безопасности Белла—ЛаПадулы

Одна из наиболее известных моделей безопасности — модель Белла-ЛаПадулы (модель мандатного управления доступом). В ней определено множество понятий, связанных с контролем доступа; даются определения субъекта, объекта и операции доступа, а также математический аппарат для их описания. Эта модель в основном известна двумя основными правилами безопасности: одно относится к чтению, а другое — к записи данных.

Пусть в системе имеются данные (файлы) двух видов: секретные и несекретные,

а пользователи этой системы также относятся к двум категориям: с уровнем допуска к несекретным данным (несекретные) и с уровнем допуска к секретным данным (секретные).

1. Свойство простой безопасности: несекретный пользователь (или процесс, запущенный от его имени) не может читать данные из секретного файла.

2. *-свойство: пользователь с уровнем доступа к секретным данным не может записывать данные в несекретный файл. Это правило менее очевидно, но не менее важно. Действительно, если пользователь с уровнем доступа к секретным данным скопирует эти данные в обычный файл (по ошибке или злему умыслу), они станут доступны любому «несекретному» пользователю. Кроме того, в системе могут быть установлены ограничения на операции с секретными файлами (например, запрет копировать эти файлы на другой компьютер, отправлять их по электронной почте и т.д.). Второе правило безопасности гарантирует, что эти файлы (или даже просто содержащиеся в них данные) никогда не станут несекретными и не «обойдут» эти ограничения. Таким образом, вирус, например, не сможет похитить конфиденциальные данные.

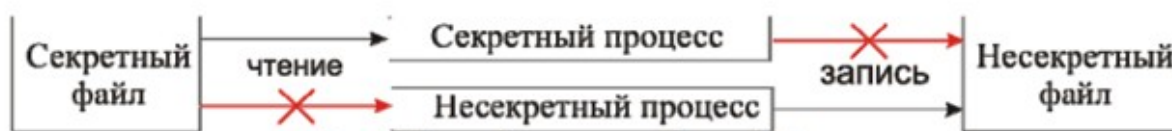


Рис. 2. Модель безопасности Белла-ЛаПадулы.

Рассмотренные правила легко распространить на случай, когда в системе необходимо иметь более двух уровней доступа — например, различаются несекретные, конфиденциальные, секретные и совершенно секретные данные. Тогда пользователь с уровнем допуска к секретным данным может читать несекретные, конфиденциальные и секретные документы, а создавать — только секретные и совершенно секретные.

Общее правило звучит так: пользователи могут читать только документы, уровень секретности которых не превышает их допуска, и не могут создавать документы ниже уровня своего допуска. То есть теоретически пользователи могут создавать документы, прочесть которые они не имеют права.

Модель Белла-ЛаПадулы стала первой значительной моделью политики безопасности, применимой для компьютеров, и до сих пор в измененном виде применяется в военной отрасли. Модель полностью формализована математически. Основной упор в модели делается на конфиденциальность, но кроме неё фактически больше ничего не представлено. Кроме того, в модели игнорируется проблема изменения классификации: предполагается, что все сведения относятся к соответствующему уровню секретности, который остается неизменным. Наконец, бывают случаи, когда пользователи должны работать с данными, которые они не имеют права увидеть. «Сведения о том, что самолет несет груз из некоторого количества бомб, возможно, имеют более высокий уровень секретности, чем уровень доступа диспетчера, но диспетчеру тем не менее необходимо знать вес груза.» [1]

3. Ролевая модель контроля доступа (RBAC)

Ролевой метод управления доступом контролирует доступ пользователей к информации на основе типов их активностей в системе (ролей). Под ролью понимается совокупность действий и обязанностей, связанных с определенным видом деятельности. Примеры ролей: администратор базы данных, менеджер, начальник отдела.

В ролевой модели с каждым объектом сопоставлен набор разрешенных операций доступа для каждой роли (а не для каждого пользователя). В свою очередь, каждому пользователю сопоставлены роли, которые он может выполнять. В некоторых системах пользователю разрешается выполнять несколько ролей одновременно, в других есть ограничение на одну или несколько не противоречащих друг другу ролей в каждый момент времени.

Для формального определения модели RBAC используются следующие соглашения:

S = субъект — человек или автоматизированный агент.

R = роль — рабочая функция или название, определяется на уровне авторизации.

P = разрешения — утверждения режима доступа к ресурсу.

SE = сессия — Соответствие между S, R и/или P.

SA = назначение субъекта (Subject Assignment). $SA \subseteq S \times R$. При этом субъекты назначаются связям ролей и субъектов в отношении «многие ко многим» (один субъект может иметь несколько ролей, а одну роль могут иметь несколько субъектов).

PA = назначение разрешения (Permission Assignment). $PA \subseteq P \times R$. При этом разрешения назначаются связям ролей в отношении «многие ко многим».

RH = частично упорядоченная иерархия ролей (Role Hierarchy). $RH \subseteq R \times R$.

На возможность наследования разрешений от противоположных ролей накладывается ограничительная норма, которая позволяет достичь надлежащего разделения режимов. Например, одному и тому же лицу может быть не позволено создать учетную запись для кого-то, а затем авторизоваться под этой учетной записью.

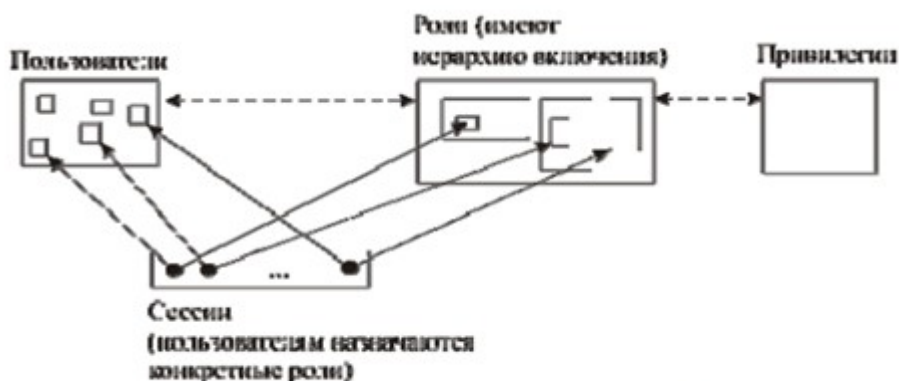


Рис. 3. Схема ролевой модели контроля доступа (RBAC)

Основные достоинства ролевой модели:

1. Простота администрирования. В отличие от модели DAC нет необходимости прописывать разрешения для каждой пары «объект-пользователь». Вместо этого прописываются разрешения для пар «объект-роль» и определяются роли каждого пользователя. При изменении области ответственности пользователя, у него просто изменяются роли. Иерархия ролей (когда роль наряду со своими собственными привилегиями может наследовать привилегии других ролей) также упрощает процесс администрирования.

2. Принцип наименьшей привилегии. Ролевая модель позволяет пользователю регистрироваться в системе ролью, минимально необходимой для выполнения требуемых задач. Запрещение полномочий, не требуемых для выполнения текущей задачи, не позволяет обойти политику безопасности системы.

3. Разделение обязанностей.

RBAC широко используется для управления пользовательскими привилегиями в пределах единой системы или приложения. Список таких систем включает в себя Microsoft Active Directory, SELinux, FreeBSD, Solaris, СУБД Oracle, PostgreSQL 8.1, SAP R/3 и множество других, эффективно применяющих RBAC.

С помощью RBAC могут быть смоделированы дискреционные и мандатные системы управления доступом.

16. Разграничение прав доступа. Идентификация и аутентификация пользователей. Объекты и субъекты доступа.

Права доступа — совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, ее носителям, процессам и другим ресурсам), установленных правовыми документами или собственником, владельцем информации.

Права доступа определяют набор действий (например, чтение, запись, выполнение), разрешенных для выполнения субъектами (например, пользователями системы) над объектами данных.

Разграничение прав доступа - предоставление субъектам различных прав доступа к объектам. Система разграничения прав доступа является главным средством защиты от несанкционированного доступа к информации или порче системы.

Доступ к файлам является частным случаем доступа к разделяемым ресурсам.

Идентификация — присвоение какому-либо объекту или субъекту уникального образа, имени или числа. Установление подлинности (аутентификация) заключается в проверке, является ли проверяемый объект (субъект) тем, за кого себя выдает.

Аутентификация — процедура, проверяющая, имеет ли пользователь с предъявленным идентификатором право на доступ к ресурсу, то есть проверка соответствия имени входа и пароля.

Аутентификация - процедура проверки подлинности заявленного

пользователя. Эта проверка позволяет достоверно убедиться, что пользователь является именно тем, кем себя объявляет. При проведении аутентификации проверяющая сторона убеждается в подлинности проверяемой стороны.

Конечная цель идентификации и установления подлинности объекта в вычислительной система – допуск его к информации ограниченного пользования в случае положительного исхода проверки или отказ в допуске в случае отрицательного исхода проверки.

Объектами идентификации и аутентификации в вычислительной системе могут быть:

- Человек (оператор, пользователь, должностное лицо)
- Техническое средство
- Документы
- Носители информации
- Информация на экране монитора и т.д.

Установление подлинности объекта может производиться человеком, аппаратным устройством, программой, вычислительной системой и т.ж.

В вычислительных системах применение указанных методов в целях защиты информации при ее обмене предполагает конфиденциальность образов имен объектов.

Объект доступа – разделяемый ресурс, над которым субъекты выполняют операции.

Субъекты доступа – пользователи, выполняющие действия с объектами.

17. Дайте краткие характеристики подходов к определению прав доступа – избирательного и принудительного.

Существует 2 основных подхода к определению прав доступа:

Избирательный доступ - для каждого объекта сам владелец может определить допустимые операции с объектами. Этот подход называется также **произвольным доступом**, т.к. позволяет администратору и владельцам объектов менять права доступа произвольным образом по их желанию. Между пользователями и группами пользователей в системах с избирательным доступом нет жестких иерархических отношений, т.е. взаимоотношений, которые определены по умолчанию и которые нельзя изменить. Исключение делается только для администратора, по умолчанию наделенного всеми правами.

Мандатный доступ (принудительный) - система наделяет пользователя определенными правами по отношению к каждому разделяемому ресурсу в зависимости от того, к какой группе пользователь отнесен. От имени системы выступает администратор, а владельцы объектов лишены возможностей управлять доступом к ним по своему усмотрению. Все группы пользователей в такой системе образуют строгую иерархию, причем каждая группа пользуется всеми правами группы более низкого уровня иерархии, к которым добавляются права данного уровня. Членам какой-либо группы не разрешается предоставлять свои права членам групп более низких

уровней иерархии. Мандатные системы доступа считаются более надежными, но менее гибкими, обычно они применяются в специализированных вычислительных системах с повышенными требованиями к защите информации. В универсальных ОС общего назначения используются как правило избирательные методы доступа.

18. Дискреционная модель доступа. Матрица доступа. Элементы матрицы доступа. Листы возможностей и листы контроля доступа. Маркер доступа. Дескриптор защиты.

Модели управления доступом регламентируют доступ субъектов к объектам. Наиболее распространена так называемая **дискреционная (произвольная) модель**, в которой обычные пользователи могут принимать участие в определении функций политики и присвоении атрибутов безопасности. Среди дискреционных моделей классической считается модель Харрисона-Руззо-Ульмана – в ней система защиты представлена в виде набора множеств, элементами которых являются составные части системы защиты: субъекты, объекты, уровни доступа, операции и т. п.

С концептуальной точки зрения текущее состояние прав доступа при дискреционном управлении описывается **матрицей**, в **строках** которой перечислены **субъекты**, в **столбцах** – **объекты**, а в **ячейках** – **операции**, которые субъект может выполнить над объектом.

Операции зависят от объектов. Например, для файлов это операции чтения, записи, выполнения, изменения атрибутов, а для принтера – операции печати и управления

Объект	O ₁	O ₂	O ₃	O ₄ (Printer)
Субъект				
S ₁	read			
S ₂				Print
S ₃		Read	Execute	
S ₄	read write		read write	

Элементы матрицы доступа могут содержать указатели на специальные **процедуры**, которые должны выполняться при обращении субъекта к объекту.

Решение о доступе в этом случае осуществляется на основании результатов выполнения процедур:

- На анализе предыдущих доступов к другим объектам
- На динамике состояния системы, т. е. права доступа субъекта зависят от текущих прав доступа других субъектов
- На значении определенных переменных, например, на значениях таймера.

Варианты задания матрицы доступа:

1. **листы возможностей**, когда для каждого субъекта S_i создается файл всех объектов, к которому имеет доступ данный субъект;

2. **листы контроля доступа**, когда для каждого объекта O_i создается список всех субъектов, имеющих право доступа к этому объекту.

Дескриптор защиты и маркер доступа

В ОС Windows все типы объектов защищены одинаковым образом. С каждым объектом связан дескриптор защиты. **Дескриптор защиты** описывается структурой типа SECURITY_DESCRIPTION и инициализируется функцией InitializeSecurityDescription.

Связь объекта с дескриптором происходит в момент создания объекта. Например, один из аргументов функции CreateFile – указатель на структуру SECURITY_DESCRIPTION, которая содержит указатель на дескриптор защиты.

Дескриптор защиты содержит SID владельца объекта, SID групп для данного объекта и два указателя на списки DACL (Discretionary ACL) и SACL (System ACL) контроля доступа. DACL и SACL содержат разрешающие и запрещающие доступ списки пользователей и групп, а также списки пользователей, чьи попытки доступа к данному объекту подлежат аудиту.

Так же как и объекты, субъекты должны иметь отличительные признаки – контекст пользователя, для того, чтобы система могла контролировать их действия. Сведения о контексте пользователя хранятся в **маркере доступа**.

Маркер доступа — объект ОС Windows, содержит информацию по безопасности сеанса и идентифицирует пользователя, группу пользователей и пользовательские привилегии.

Маркер доступа используется Windows, когда процесс пытается взаимодействовать с объектами, дескрипторы безопасности которых требуют контроль доступа.

При интерактивном входе в систему пользователь обычно вводит свое имя и пароль. Система (процедура Winlogon) по имени находит соответствующую учетную запись, извлекает из нее необходимую информацию о пользователе, формирует список привилегий, ассоциированных с пользователем и его группами и все это объединяет в структуру данных, которая называется **маркером доступа**.

Вслед за оболочкой (Windows Explorer) все процессы (и потоки процесса), запускаемые пользователем, наследуют этот маркер. Когда один процесс создает другой при помощи функции CreateProcess, дочернему процессу передается дубликат маркера, который, таким образом, распространяется по системе.

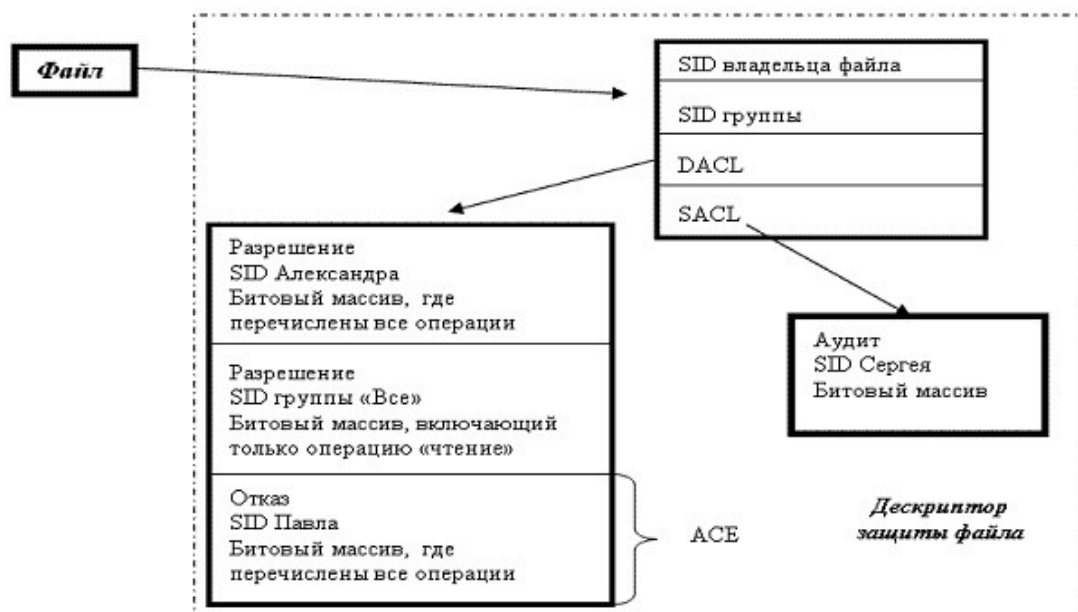
Основные компоненты **маркера доступа**

SID пользователя	SID1...SIDn Идентификаторы групп пользователя	DACL по умолчанию	Привилегии	Другие параметры
---------------------	---	----------------------	------------	---------------------

Включая в маркер информацию о защите, в частности, DACL, Windows упрощает создание объектов со стандартными атрибутами защиты. Если процесс не позаботится о том, чтобы явным образом указать атрибуты безопасности объекта, на основании списка DACL, присутствующие в его

маркере, будут сформированы права доступа к объекту по умолчанию.

Настройку стандартной защиты можно осуществить при помощи функции SetTokenInformation. При этом, поскольку объекты в Windows отличаются большим разнообразием, в списке DACL «по умолчанию» можно указать только так называемые базовые права доступа, из которых система будет формировать стандартные права доступа в зависимости от вида создаваемого объекта.



Кроме списка DACL дескриптор защиты включает также список SACL, который имеет такую же структуру, что и DACL, то есть состоит из таких же ACE записей, только вместо операций, регламентирующих доступ к объекту, в нем перечислены операции, подлежащие аудиту. В примере операции с файлом процессов, запускаемых Сергеем, описанные в соответствующем битовом массиве будут регистрироваться в системном журнале.

19-21. Мандатная модель доступа.

Многоуровневые модели предполагают формализацию процедуры назначения прав доступа посредством использования так называемых меток конфиденциальности или мандатов, назначаемых субъектам или объектам доступа.

19. Мандатная модель доступа. Монитор обращения. Требования к мандатному механизму.

Монитор обращений контролирует допустимость выполнения субъектами (пользователями) определенных операций над объектами (пассивными сущностями).

Монитор проверяет каждое обращение пользователя к программам или данным на предмет согласованности с набором действий, допустимых для пользователя.

Монитор обращений должен обладать тремя качествами:

- Изолированность - необходимо предупредить возможность отслеживания работы монитора.
- Полнота - монитор должен вызываться при каждом обращении, не должно быть способов обойти его.

- Верифицируемость - монитор должен быть компактным, чтобы его можно было проанализировать и протестировать, будучи уверенным в полноте тестирования.

Требования к мандатному механизму:

- все субъекты и объекты компьютерной системы должны быть однозначно идентифицированы;
- имеется линейно упорядоченный набор меток конфиденциальности и соответствующих им уровней (степеней) допуска (нулевая метка или степень соответствуют общедоступному объекту и степени допуска к работе только с общедоступными объектами);
- каждому объекту компьютерной системы присвоена метка конфиденциальности;
- каждому субъекту компьютерной системы присваивается степень допуска;
- в процессе своего существования каждый субъект имеет свой уровень конфиденциальности, равный максимуму из меток конфиденциальности объектов, к которым данный субъект получил доступ;
- в компьютерной системе существует привилегированный пользователь, имеющий полномочия на удаление любого объекта системы;
- понизить метку конфиденциальности объекта может только субъект, имеющий доступ к данному объекту и обладающий специальной привилегией;
- право на чтение информации из объекта получает только тот субъект, чья степень допуска не меньше метки конфиденциальности данного объекта (правило «не читать выше»);
- право на запись информации в объект получает только тот субъект, чей уровень конфиденциальности не больше метки конфиденциальности данного объекта (правило «не записывать ниже»).

20. Мандатная модель доступа. Мандатный принцип контроля доступа

Данные могут передаваться субъектам, если выполняются правила:

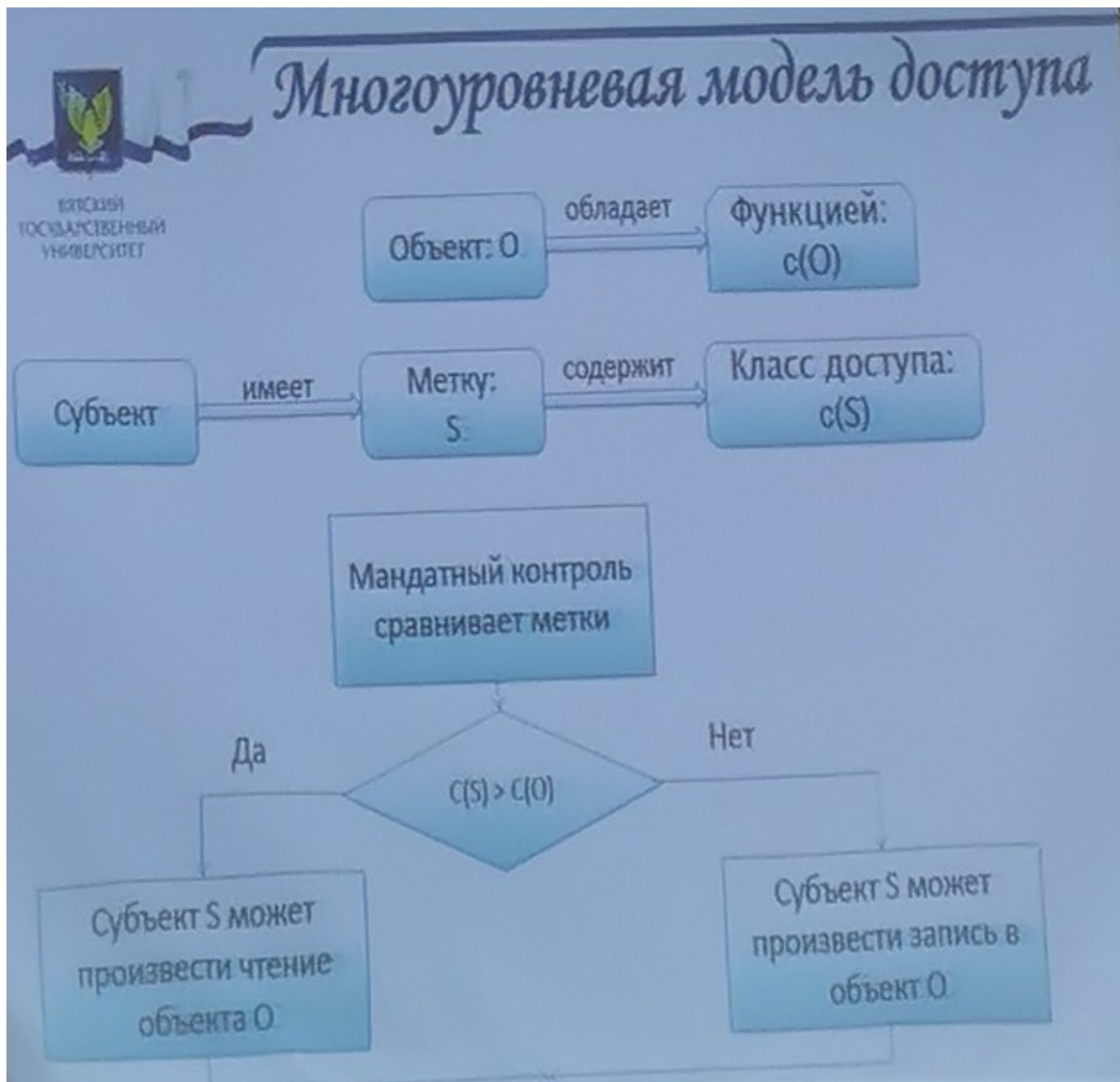
- Данные могут передавать субъектом самому себе (если X не меньше X).
- Данные могут передаваться от субъекта A к субъекту C , если они могут передаваться от субъекта A к субъекту B и от B к C .
- Если X не больше Y и Y не больше X , то $X=Y$

Эти правила представляют свойства рефлексивности, транзитивности и асимметричности.

Многоуровневая модель доступа в современных системах защиты реализуется через мандатный контроль или мандатную политику.

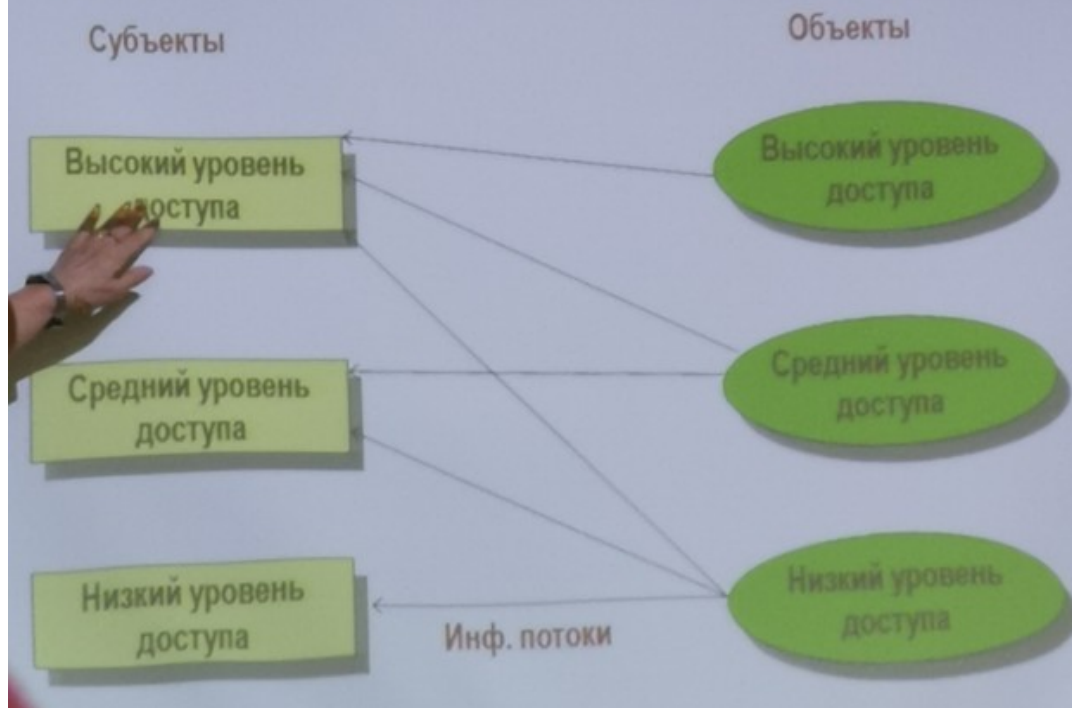
Основу реализации управления доступом составляют:

- Формальное сравнение метки субъекта, запросившего доступ, и метки объекта, к которому запрошен доступ.
- Принятие решений о предоставлении доступа на основе некоторых правил, основу которых составляет противодействие снижению уровня конфиденциальности защищаемой информации.

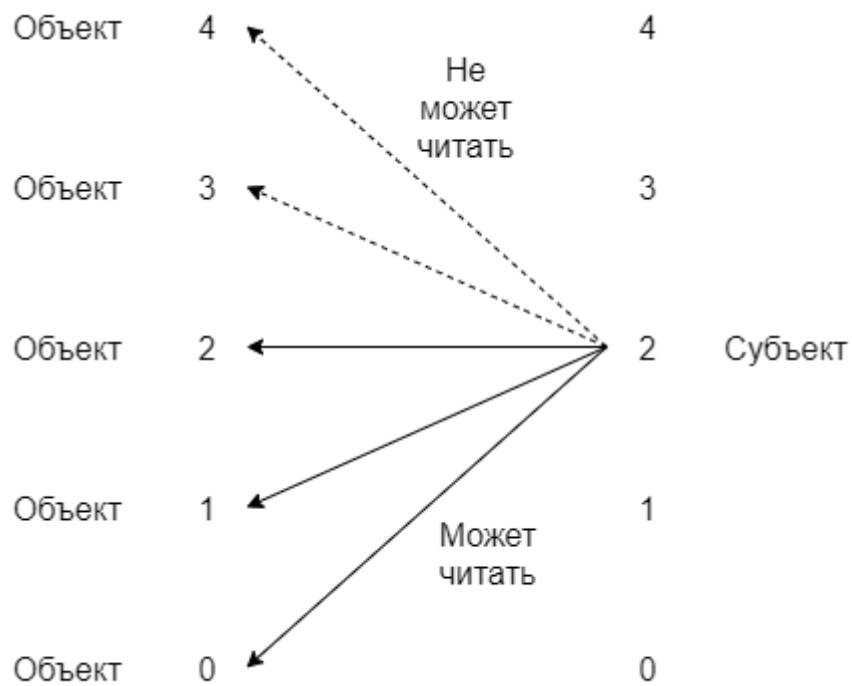


Субъект может читать объект, только если иерархическая классификация в классификационном уровне субъекта не меньше, чем иерархическая классификация в классификационном уровне объекта.

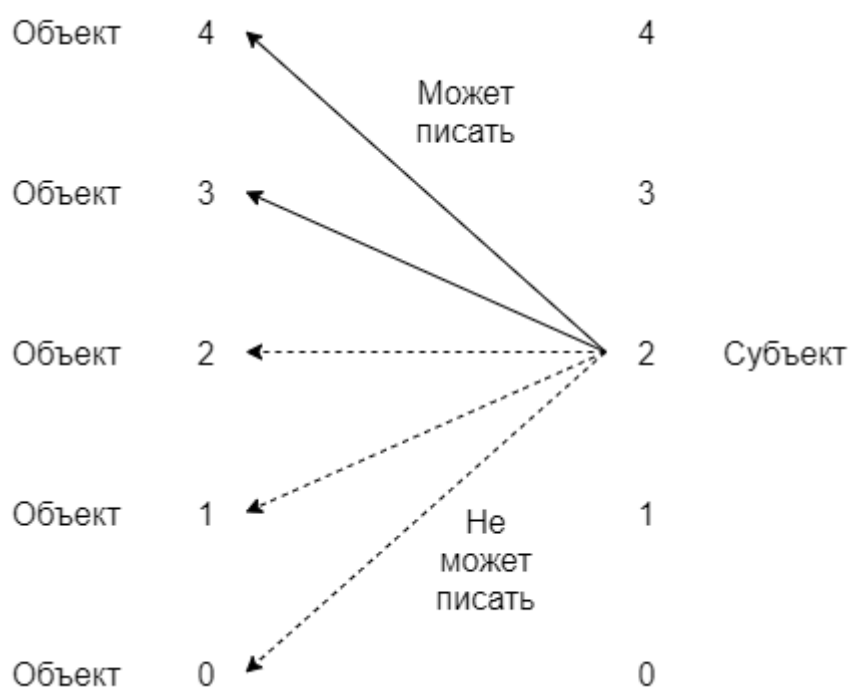
Чтение информации



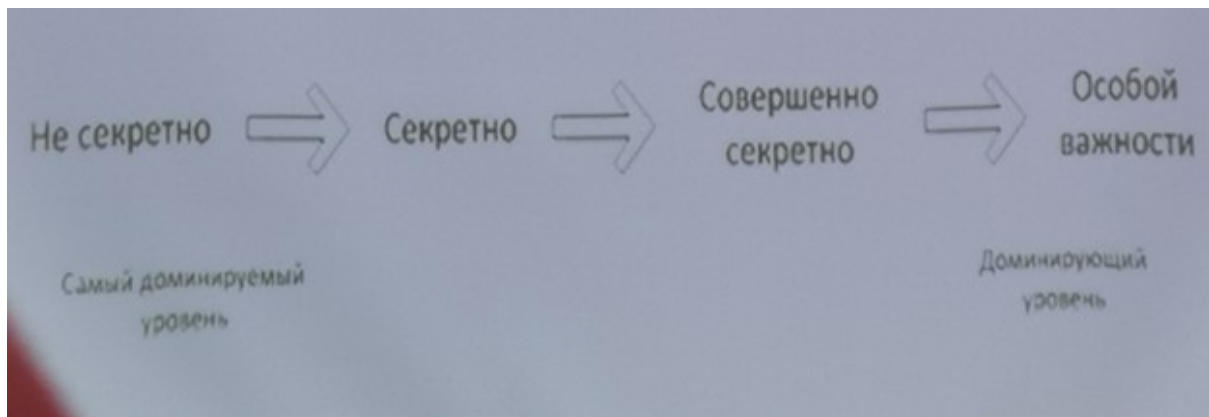
Субъект осуществляет запись в объект, только если классификационный уровень субъекта в иерархической классификации не больше, чем классификационный уровень объекта в иерархической классификации.



Упрощенная схема вышесказанного. 4 - самый высокий уровень



Уровни доступа упорядочиваются по доминированию одного уровня над другим.



21. Мандатная модель доступа. Два правила доступа к защищенным файлам

Контроль доступа основывается на двух правилах:

- Пользователь (Субъект) имеет право читать только те документы, уровень безопасности которых не превышает его собственный уровень безопасности. Это правило обеспечивает защиту информации обрабатываемой более высокоуровневыми пользователями, от доступа со стороны низкоуровневых пользователей.
- Субъект имеет право заносить информацию только в документы, уровень которых выше или равен уровню субъекта.

Второе правило исключает возможность перемещения информации с более секретных уровней на менее секретные.

22. Как организован контроль доступа в ОС Windows NT?

Система безопасности следит за выполнением тех или иных действий клиента при помощи двух основных механизмов:

- **Привилегия** – разрешение на выполнение некоторым пользователем некоторого действия в отношении всей системы в целом. (устанавливать время, вход, выход и т.д.)
- **Право** – это разрешение на выполнение некоторым пользователем некоторого действия в отношении некоторого отдельного объекта.

Права назначаются в отношении конкретных объектов, доступ к которым контролируется операционной системой.

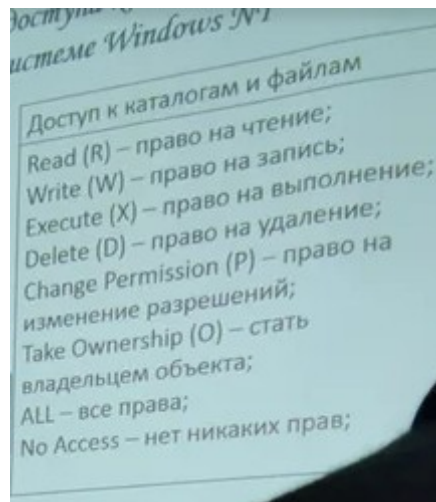
Например, пользователь может обладать правом чтения некоторого файла, но этот же пользователь может не обладать правом записи информации в этот файл. Права могут быть также негативными. Например, право может запрещать пользователю чтение защищаемого объекта (например, именованного канала). Кроме того, правом могут обладать не только отдельные пользователи, но и группы пользователей.

В Windows NT для контроля доступа к ресурсам любого вида имеется общий модуль ядра – **менеджер безопасности**.

В качестве субъектов доступа могут выступать как отдельные пользователи, так и группы пользователей. У каждого объекта доступа существует владелец.

Владелец – субъект, который вправе выполнять с созданным им объектом любые доступные операции.

Администратор – субъект, обладающий всеми правами на все объекты.



Стандартный набор прав

Константа	Значение
DELETE	Право на удаление объекта
READ_CONTROL	Право на чтение дескриптора безопасности объекта, за исключением информации о SACL
SYNCHRONIZE	Право на использование объекта для синхронизации.
WRITE_DAC	Право на изменение DACL объекта в дескрипторе безопасности
WRITE_OWNER	Право на изменение владельца объекта в дескрипторе безопасности

Процесс-оболочка – процесс, создаваемый при входе пользователя в систему, поддерживающие диалог с этим пользователем и запускающий для него другие процессы. Процесс-оболочка получает от пользователя символьное имя и пароль, и находит по ним числовые идентификаторы пользователя и его групп. Для Windows это Windows Explorer (explorer.exe).

В ОС Windows все типы объектов защищены одинаковым образом. С каждым объектом связан дескриптор защиты. **Дескриптор защиты** описывается структурой типа SECURITY_DESCRIPTION и инициализируется функцией InitializeSecurityDescription.

Связь объекта с дескриптором происходит в момент создания объекта. Например, один из аргументов функции CreateFile – указатель на структуру SECURITY_DESCRIPTION, которая содержит указатель на дескриптор защиты.

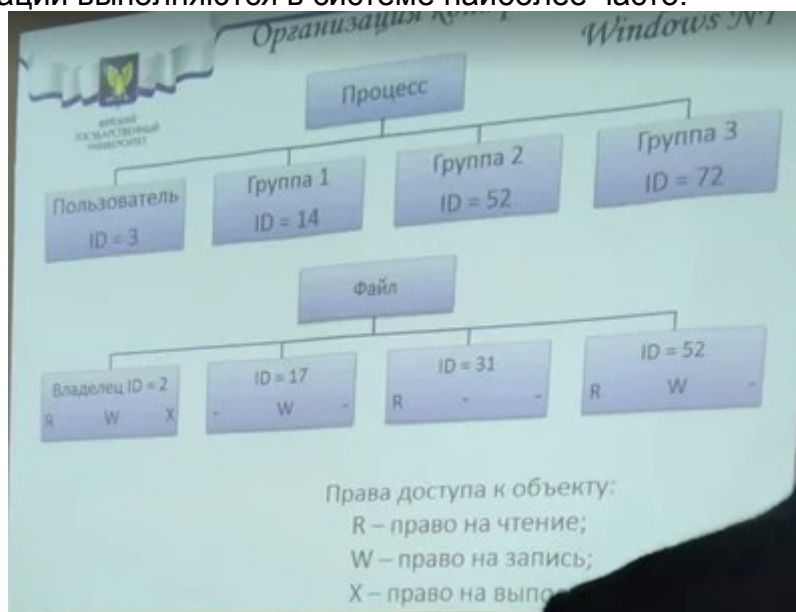
Дескриптор защиты содержит SID владельца объекта, SID групп для данного объекта и два указателя на списки DACL (Discretionary ACL) и SACL (System ACL) контроля доступа. DACL и SACL содержат разрешающие и запрещающие доступ списки пользователей и групп, а также списки пользователей, чьи попытки доступа к данному объекту подлежат аудиту.

ACL (Access Control List) – список, в котором описываются права на

выполнение операций пользователями и группами пользователей по отношению к этому файлу или каталогу.

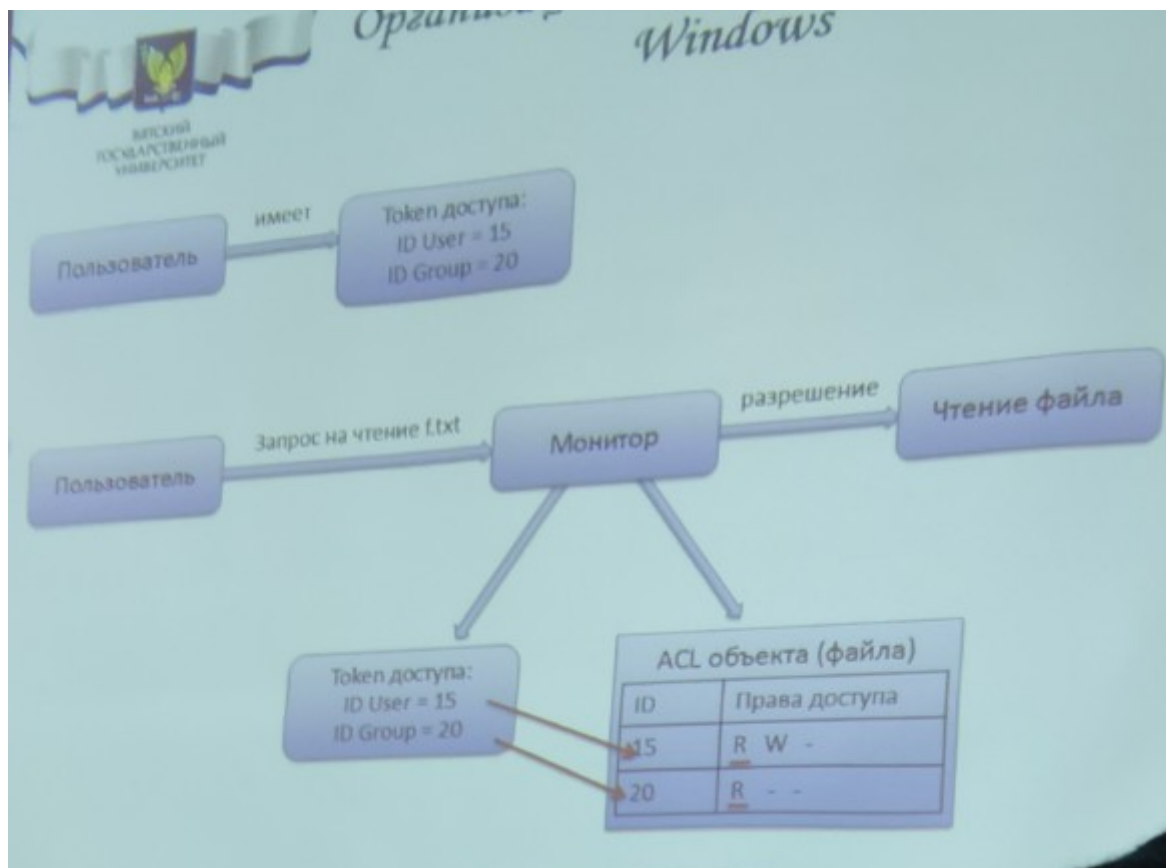
Список управления доступом хранится на диске в соответствующей области. Список ACL состоит из элементов управления доступом (ACE). ACE могут описывать как разрешенные операции, так и запрещенные. При этом, каждый элемент соответствует одному идентификатору субъекта.

В Windows NT однозначно определены правила, по которым вновь создаваемому объекту назначается список ACL. Если исполняемый процесс во время создания объекта явно задает все права доступа к вновь создаваемому объекту, то система безопасности приписывает этот ACL объекту. Если же программа не снабжает объект списком ACL, а объект имеет имя, то применяется принцип наследования разрешений. Система безопасности просматривает ACL того каталога объектов, в котором хранится имя нового объекта. Некоторые из кодов ACL каталога объектов могут быть помечены как наследуемые. Это означает, что они могут быть приписаны к новым объектам, создаваемым в этом каталоге. В том случае, когда процесс не задал явно список ACL для создаваемого объекта, и объект каталог не имеет наследуемых элементов ACL, используется список ACL по умолчанию из токена доступа процесса. Наследование разрешений употребляется наиболее часто при создании нового объекта, и оно особенно эффективно при создании файлов, так как эти операции выполняются в системе наиболее часто.



Монитор безопасности – процесс, работающий в привилегированном режиме и проверяющий права доступа для объектов любого типа. Когда процесс запрашивает доступ к объекту, управление передается монитору безопасности, который сравнивает идентификаторы пользователя и групп пользователей из токена доступа с идентификаторами, хранящимися в ACL объекта. Если какой-либо элемент разрешает текущий запрос, то пользователь получает доступ к объекту.

Централизация функций контроля доступа повышает надёжность ОС.



Для системы безопасности Windows NT характерно большое количество predetermined субъектов доступа. **Права и разрешения, данные группе, автоматически предоставляются всем членам этой группы.**

При интерактивном входе в систему пользователь обычно вводит свое имя и пароль. Система (процедура Winlogon) по имени находит соответствующую учетную запись, извлекает из нее необходимую информацию о пользователе, формирует список привилегий, ассоциированных с пользователем и его группами и все это объединяет в структуру данных, которая называется маркером доступа (см. **вопрос 18**).

Вслед за оболочкой (Windows Explorer) все процессы (и потоки процесса), запускаемые пользователем, наследуют этот маркер. Когда один процесс создает другой при помощи функции CreateProcess, дочернему процессу передается дубликат маркера, который, таким образом, распространяется по системе.

Встроенный администратор в Windows NT, в отличие от супер-пользователя Unix, может не иметь некоторых разрешений на доступ к объекту. Для реализации этой возможности идентификаторы администратора и группы администраторов могут входить в ACL, как и идентификаторы рядовых пользователей. Однако администратор все же имеет возможность выполнить любые операции с любыми объектами, так как он всегда может стать владельцем объекта, а затем уже как владелец получить полный набор разрешений. Однако вернуть владение предыдущему владельцу объекта администратор не может, поэтому пользователь всегда может узнать о том, что с его ресурсом работал администратор.

Для смены в некоторых ситуациях процессом своих идентификаторов в Windows NT используется механизм **олицетворения**. В Windows NT

существуют простые **субъекты** и **субъекты-серверы**. Простой субъект – это процесс, которому не разрешается смена токена доступа и соответственно смена идентификаторов. **Субъект-сервер** – это процесс, который работает в качестве сервера и обслуживает процессы своих клиентов, например, процесс файлового сервера. Поэтому такому процессу разрешается получить токен доступа у процесса клиента, запросившего у сервера выполнение некоторого действия и использовать его при доступе к объектам.

23. Как организован контроль доступа в ОС UNIX?

Основные права доступа к файлам в Linux и других UNIX-подобных ОС

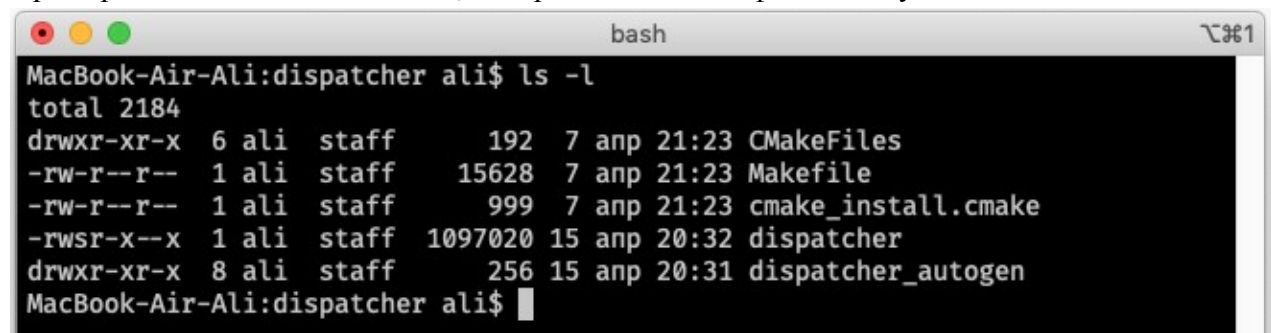
Изначально каждый файл имел три параметра доступа.

- **Чтение (r)** – разрешает получать содержимое файла, но на запись нет. Для каталога позволяет получить список файлов и каталогов, расположенных в нем. Код - 4
- **Запись (w)** – разрешает записывать новые данные в файл или изменять существующие, а также позволяет создавать и изменять файлы и каталоги. Код - 2
- **Выполнение (x)** – вы не можете выполнить программу, если у нее нет флага выполнения. Этот атрибут устанавливается для всех программ и скриптом, именно с помощью него система может понять, что этот файл нужно запускать как программу. Код - 1

Каждый файл имеет три категории пользователей, для которых можно устанавливать различные сочетания прав доступа:

- **Владелец** – набор прав для владельца файла, пользователя, которые его создал или сейчас установлен его владельцем. Обычно владелец имеет все права, чтение, запись и выполнение.
- **Группа** – любая группа пользователей, существующая в системе и привязанная к файлу. Но это может быть только одна группа и обычно это группа владельца, хотя для файла можно назначить и другую группу.
- **Остальные** – все пользователи, кроме владельца и пользователей, входящих в группу файла. Именно с помощью этих набором полномочий устанавливаются права файлов в линукс. Каждый пользователь может получить полный доступ только к файлам, владельцем которых он является или к тем, доступ к которым ему разрешен. Только пользователь root может работать со всеми файлами независимо от их набора и полномочий.

Пример вывода команды `ls -l`, которая показывает флаги доступа



```
MacBook-Air-Ali:dispatcher ali$ ls -l
total 2184
drwxr-xr-x  6 ali  staff    192  7 anp  21:23 CMakeFiles
-rw-r--r--  1 ali  staff  15628  7 anp  21:23 Makefile
-rw-r--r--  1 ali  staff   999  7 anp  21:23 cmake_install.cmake
-rwsr-x--x  1 ali  staff 1097020 15 anp  20:32 dispatcher
drwxr-xr-x  8 ali  staff   256 15 anp  20:31 dispatcher_autogen
MacBook-Air-Ali:dispatcher ali$
```

Например у файла `dispatcher` есть следующие права:

- Для владельца: чтение + запись + исполнение (флаг SUID установлен)

- Для группы: чтение + исполнение
- Для остальных: чтение + исполнение

Владелец: **ali**

Группа: **stuff**

Но со временем такой системы стало не хватать и было добавлено несколько флагов, которые позволяют делать файлы не изменяемыми или же позволяют выполнять их от имени суперпользователя.

Специальные права доступа к файлам в Linux

Для того, чтобы позволить обычным пользователям выполнять программы от имени суперпользователя без знания его пароля была придумана такая вещь, как SUID и SGID биты.

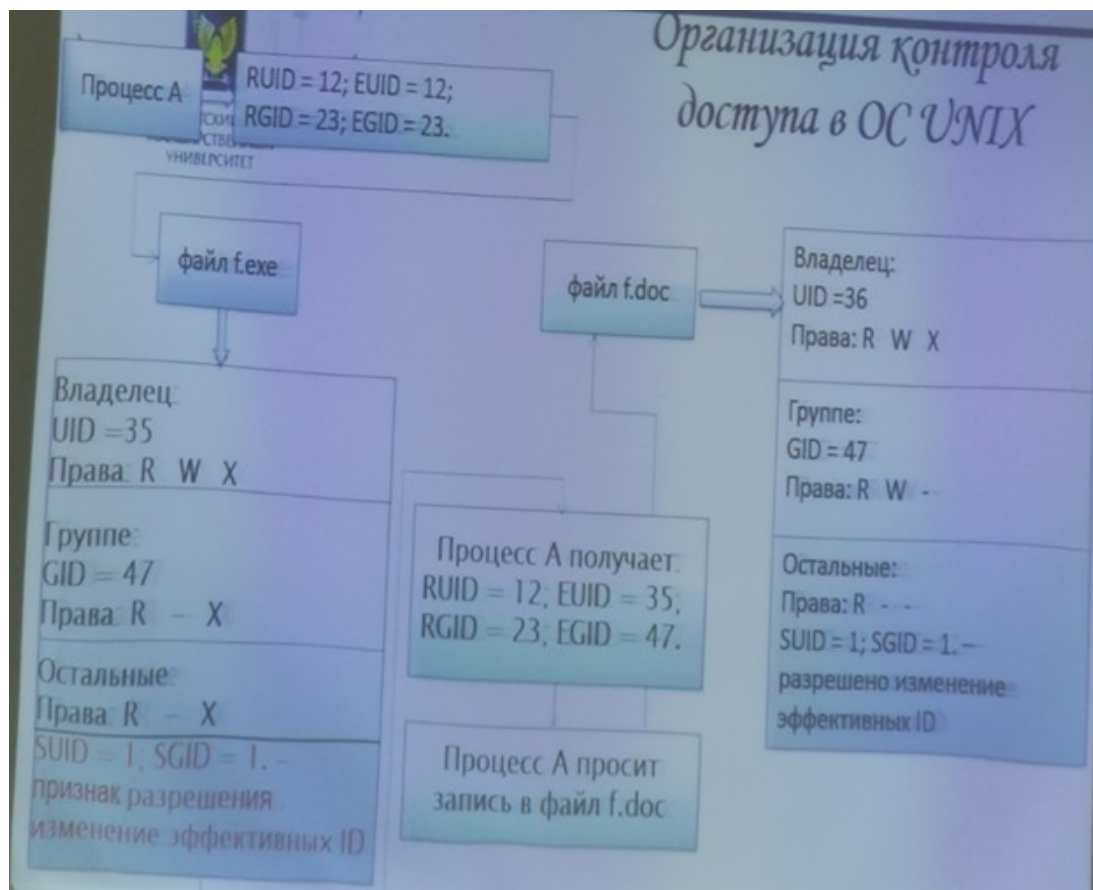
- **SUID** – если этот бит установлен, то при выполнении программы, id пользователя, от которого она запущена заменяется на id владельца файла. Фактически, это позволяет обычным пользователям запускать программы от имени суперпользователя.
- **SGID** – этот флаг работает аналогичным образом, только разница в том, что пользователь считается членом группы, с которой связан файл, а не групп, с которым он действительно принадлежит. Если SGID флаг установлен на каталог, все файлы, созданные в нем, будут связаны с группой каталога, а не пользователя. Такое поведение используется для организации общих папок.
- **Sticky-bit** – этот бит тоже используется для создания общих папок. Если он установлен, то пользователи, которым предоставлены права с помощью этого бита могут только создавать, читать и выполнять файлы, но не могут удалять файлы, принадлежащие другим пользователям.

С каждым процессом UNIX связаны два идентификатора:

- Идентификатор пользователя от имени которого был создан процесс (Real User ID (RUID))
- Идентификатор группы, к которой принадлежит данный пользователь (Real Group ID (RGID))

При проверке прав доступа используется не эти идентификаторы, а **эффективные идентификаторы (Effective User id и Effective Group Id)**

Файл имеет два признака разрешения смены идентификаторов (Set user id in execution SUID) и Set Group ID on Execution (SGID)), которые позволяют смену идентификатора пользователя и группы при выполнении данного файла.



24. Достоинства и недостатки дискреционной и мандатной моделей.

Достоинства **дискреционной модели** — относительно простая реализация (проверка прав доступа субъекта к объекту производится в момент открытия этого объекта в процессе субъекта), хорошая изученность (применяется в Windows и Unix).

Недостатки :

- Статичность разграничения доступа — права доступа к уже открытому субъектом объекту в дальнейшем не изменяются независимо от состояния компьютерной системы.
- Нет проверки на то, не приведет ли разрешение доступа к объекту для некоторого субъекта к нарушению безопасности информации в компьютерной системе. (пример, владелец бд с конфиденциальной информацией, дав разрешение на её чтение другому пользователю, делает этого пользователя фактически владельцем защищаемой информации)
- Не выдерживает атак троянов. Для облегчения работы сисадмина есть автоматическое назначение прав доступа субъектам (может привести к нарушениям прав доступа).

Достоинства **мандатной модели**:

- Самое важное достоинство заключается в том, что пользователь не может полностью управлять доступом к ресурсам, которые он создаёт.
- Такая система запрещает пользователю или процессу, обладающему определённым уровнем доверия, получать доступ к информации, процессам или устройствам более защищённого уровня.

- Многоуровневая модель доступа устойчива к атакам троянским конем;
- Такая модель создана, в основном, для сохранения секретности информации.

Недостатки:

- Отдельно взятые категории одного уровня равнозначны, что приводит в большинстве случаев к избыточности прав доступа для конкретных субъектов в пределах соответствующих уровней.
- Вопросы целостности при помощи этой политики не решаются или решаются как побочный результат защиты секретности.

Различают два основных подхода к определению прав доступа

<https://mylektsii.ru/7-27713.html>

25. Вредоносные программы. Классификация. Способы защиты.

Вредоносное программное обеспечение – это собирательный термин, обозначающий вредоносную программу или код, который может причинить ущерб компьютерной системе.

Вредоносное ПО умышленно создается враждебным, назойливым и агрессивным. Оно стремится проникнуть в систему, нанести урон, частично перехватить контроль над некоторыми процессами или вовсе вывести из строя компьютеры, компьютерные системы, сети, планшетные и мобильные устройства.

Ниже перечислены основные виды вредоносных программ:

- Агенты ботнетов. Ботнетом называется группа зараженных компьютеров, получающих команды от злоумышленника; за прием и исполнение этих команд отвечает соответствующая вредоносная программа.
- Эксплойты — хакерские утилиты, предназначенные для эксплуатации уязвимостей в программном обеспечении.
- Бэкдоры — программы для удаленного подключения к компьютеру и управления им.
- Компьютерные вирусы. Вирусом принято называть программу, которая внедряет свой код в другие приложения, при каждом запуске инфицированного объекта этот код исполняется.
- Руткиты — средства сокрытия вредоносной деятельности (например, другие приложения не смогут обнаружить файлы, принадлежащие нежелательному ПО).
- Сетевые черви — вредоносные программы с самой разной функциональной нагрузкой, которые способны самостоятельно распространяться по компьютерным сетям.
- «Троянские кони» — широкий класс вредоносных объектов разнообразного назначения, которые обычно не имеют собственного механизма распространения. Название произошло от ранней тактики их проникновения — под видом легитимной программы или в качестве скрытого дополнения к ней.
- В особую группу можно выделить вымогатели и шифровальщики (ransomware). Сценарий работы таких вредоносных программ состоит в том, что они каким-либо способом блокируют доступ пользователя к его данным и требуют выкуп за разблокировку.

Ниже перечислены основные и наиболее эффективные меры для повышения безопасности:

- Регулярно обновлять программное обеспечение. Это позволяет исправить ошибки и уязвимости.
- Использовать антивирусов.
- Ограничить физический доступ к компьютеру посторонних лиц.
- Использовать межсетевой экран (аппаратный или программный), контролирующий выход в сеть Интернет с персонального компьютера на основании соответствующих политик.
- Делать резервное копирование важной информации на внешние носители и отключать их от компьютера.
- Не загружать приложения из сомнительных источников, не переходить по странным ссылкам и т.д.

26. Криптографическая защита информации. Понятие криптологии, криптографии и криптоанализа. Для решений каких проблем безопасности применяются криптографические методы?

Криптология – наука, занимающаяся методами шифрования и расшифровывания. Криптология состоит из двух частей – криптографии и криптоанализа.

Криптография – наука о способах преобразования (шифрования) информации с целью её защиты от незаконных пользователей. Криптография занимается поиском и исследованием математических методов преобразования информации. (защита, т.е. разработка шифров)

Криптоанализ – наука о методах использования вскрытия шифров. То есть сфера криптоанализа – исследование возможности расшифровки информации без знания ключей (нападение, т. е. атака на шифры.)

Две дисциплины связаны друг с другом, и не бывает хороших криптографов, не владеющих методами криптоанализа: стойкость разработанного шифра можно доказать только с помощью проведения различных атак на шифр.

Криптографические методы могут применяться для решения следующих проблем:

- Конфиденциальности передаваемых/храняемых данных
- Аутентификации
- Целостности храняемых и передаваемых данных
- Обеспечения подлинности документов.

27. Перечислите способы обеспечения конфиденциальность информации между абонентами. Базовые методы преобразования информации в криптологии.

Обеспечить конфиденциальность информации между абонентами в общем случае можно одним из трех способов

- Создать абсолютно надежный, недоступный для других канал связи между абонентами
- Использовать общедоступный канал связи, но скрыть сам факт передачи информации
- Использовать общедоступный канал связи, но передавать по нему информацию в преобразованном виде, причем преобразовать ее надо так, чтобы восстановить ее мог только адресат

Базовых методов преобразования информации, которыми располагает криптография,

немного, среди них:

- Шифрование (симметричное и несимметричное)
- Вычисление хэш-функций
- Генерация электронной цифровой подписи
- Генерация последовательности псевдослучайных чисел

28. Наивная криптография. Алгоритмы, использовавшиеся на первом этапе развития криптографии.

Для наивной криптографии (до нач. XVI века) характерно использование любых (обычно примитивных) способов запутывания противника относительно содержания шифруемых текстов.

1. Примерно в 1900 году до н. э. древние египтяне начали видоизменять и искажать иероглифы, чтобы закодировать определенные сообщения.
2. Шифр Считала известен со времен войны Спарты против Афин в V веке до н. э. Это был цилиндр, на который наматывалась узкая папирусная лента (без просветов и нахлестов), а затем на этой ленте вдоль его оси записывался необходимый для передачи текст. Лента сматывалась с цилиндра и отправлялась адресату, который имея цилиндр точно такого же диаметра, наматывал ленту на него и читал сообщение.
3. Шифр Цезаря — это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом, находящимся на некотором постоянном числе позиций левее или правее него в алфавите. Например, в шифре со сдвигом вправо на 3, А была бы заменена на Г, Б станет Д, и так далее.
4. В криптографии **квадрат Полибия** - оригинальный код простой замены, одна из древнейших систем кодирования, предложенная Полибием.

Шаг 1: Формирование таблицы шифрования

К каждому языку отдельно составляется таблица шифрования с одинаковым (не обязательно) количеством пронумерованных строк и столбцов, параметры которой зависят от его мощности (количества букв в алфавите). Берутся два целых числа, произведение которых ближе всего к количеству букв в языке — получаем нужное число строк и столбцов. Затем вписываем в таблицу все буквы алфавита подряд — по одной в каждую клетку. При нехватке клеток можно вписать в одну две буквы (редко употребляющиеся или схожие по употреблению).

Латинский алфавит

В современном латинском алфавите 26 букв, следовательно таблица должна состоять из 5 строк и 5 столбцов, так как $25=5*5$ наиболее близкое к 26 число. При этом буквы I, J не различаются (J **отождествляется** с буквой I), так как не хватает 1 ячейки:

	1	2	3	4	5
1	A	B	C	D	E

2	F	G	H	I/ J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Русский алфавит

Идею формирования таблицы шифрования проиллюстрируем для **русского языка**. Число букв в русском алфавите отличается от числа букв в греческом алфавите, поэтому размер таблицы выбран другой (квадрат $7*5=36$, поскольку 36 наиболее близкое число к 33):

	1	2	3	4	5	6
1	А	Б	В	Г	Д	Е
2	Ё	Ж	З	И	Й	К
3	Л	М	Н	О	П	Р
4	С	Т	У	Ф	Х	Ц
5	Ч	Ш	Щ	Ъ	Ы	Ь
6	Э	Ю	Я	-	-	-

Шаг 2: Принцип шифрования

Существует несколько методов шифрования с помощью квадрата Полибия. Ниже приведены три из них.

Метод 1

Зашифруем слово «SOMETEXT»:

Для шифрования на квадрате находили букву текста и вставляли в шифровку нижнюю от неё в том же столбце. Если буква была в нижней строке, то брали верхнюю из того же столбца.

Таблица координат								
Буква текста:	S	O	M	E	T	E	X	T

Буква шифротекста :	Х	Т	Р	К	У	К	С	У
--------------------------------	----------	----------	----------	----------	----------	----------	----------	----------

Таким образом после шифрования получаем:

Результат	
До шифрования:	SOMETE ХТ
После шифрования:	ХТRКУКС У

Метод 2

Сообщение преобразуется в координаты по квадрату Полибия, координаты записываются вертикально:

Таблица координат								
Буква:	S	O	M	E	T	E	X	T
Координата вертикальная:	3	4	2	5	4	5	3	4
Координата горизонтальная:	4	3	3	1	4	1	5	4

Затем координаты считывают по строкам:

34 25 45 34 43 31 41 54
(*)

Далее координаты преобразуются в буквы по этому же квадрату:

Таблица координат								
Координата вертикальная:	3	2	4	3	4	3	4	5

Координата горизонтальная:	4	5	5	4	3	1	1	4
Буква:	S	W	Y	S	O	C	D	U

Таким образом после шифрования получаем:

Результат	
До шифрования:	SOMETEX T
После шифрования:	SWYSOCD U

Метод 3

Усложнённый вариант, который заключается в следующем: полученный первичный шифротекст (*) шифруется вторично. При этом он выписывается без разбиения на пары:

3425453443314154

Полученная последовательность цифр сдвигается циклически влево на один шаг (нечётное количество шагов):

4254534433141543

Эта последовательность вновь разбивается в группы по два:

42 54 53 44 33 14 15 43

и по таблице заменяется на окончательный шифротекст:

Таблица координат								
Координата вертикальная:	4	5	5	4	3	1	1	4
Координата горизонтальная:	2	4	3	4	3	4	5	3

Буква:	I	U	P	T	N	Q	V	O
--------	---	---	---	---	---	---	---	---

Таким образом после шифрования получаем:

Результат	
До шифрования:	SOMETE ХТ
После шифрования:	IUPTNQV О

Метод 4 (БЫЛ У КАРАВАЕВОЙ В ЛЕКЦИИ В ЗАДАЧАХ И В БИЛЕТЕ ЛУЧШЕ ЕГО ПИСАТЬ СКОРЕЕ ВСЕГО)

По картинке все понятно

Квадрат Полибия

	1	2	3	4	5	6
1	А	Б	В	Г	Д	Е
2	Ж	З	И	Й	К	Л
3	М	Н	О	П	Р	С
4	Т	У	Ф	Х	Ц	Ч
5	Ш	Щ	Ъ	Ы	Ь	Э
6	Ю	Я				

Пример:

Г Р Е Ц И Я

14 35 16 45 23 62

5. **Атбаш** – простой шифр подстановки для алфавитного письма. Правило шифрования состоит в замене i -й буквы алфавита буквой с номером $n-i+1$, где n – число букв в алфавите.

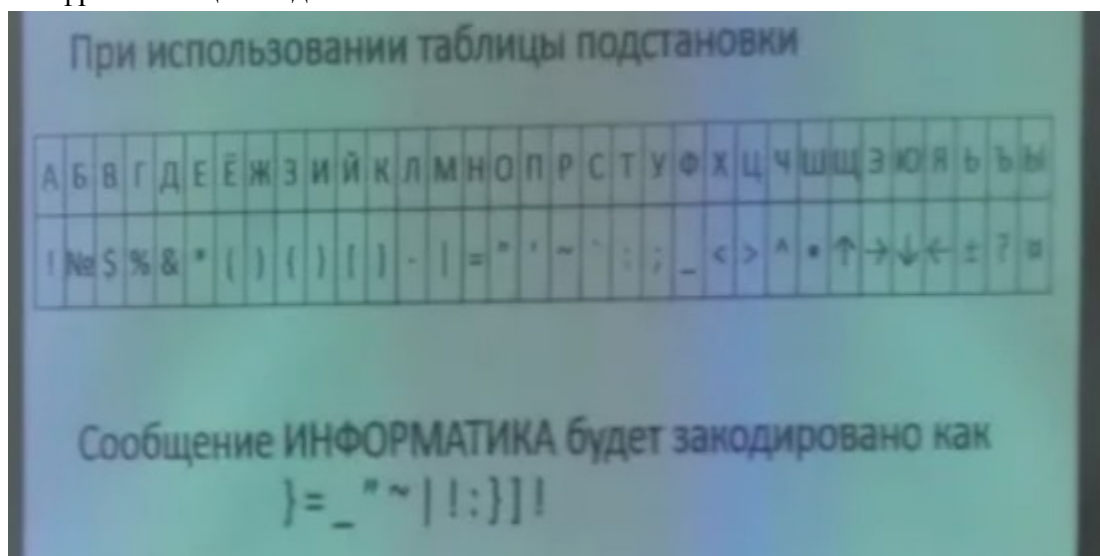
Исходный текст	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Зашифрованный текст	Z	Y	X	W	V	U	T	S	R	Q	P	O	N	M	L	K	J	I	H	G	F	E	D	C	B	A

Исходны й текст	А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
Зашифро ванный текст	я	ю	э	ы	ь	щ	ш	ч	ц	х	ф	у	т	с	р	п	о	н	м	л	к	й	и	з	ж	ё	е	д	г	в	б	а	

29. Формальная криптология. Алгоритмы, использовавшиеся на втором этапе развития криптографии.

Этап формальной криптографии (кон. XV – нач. XX веков) связан с появлением формализованных и относительно стойких к ручному криптоанализу шифров.

1. Шифр с таблицей подстановки



2. Шифр Виженера

https://ru.wikipedia.org/wiki/Шифр_Виженера

3. Шифр Плейфера

https://ru.wikipedia.org/wiki/Шифр_Плейфера

4. Шифр Уитстона

https://ru.wikipedia.org/wiki/Шифр_Уитстона

Презентация по Плейферу и Уитстону

https://drive.google.com/open?id=1HL0HJPzR3xoGNBJQYvdBD_U8ugJhHCH_

5. Роторные криптосистемы.

Многоалфавитная подстановка с помощью роторной машины реализуется вариацией взаимного положения вращающихся роторов, каждый из которых осуществляет «прошитую» в нем подстановку. Шифровальная система очень похожа на шифр Виженера, но с лучшей защитой. (Одной первых машин была Enigma 1917 Эдвард Хеберном, усовершенствована Артуром Кирхом; Sigaba (США), Турех (Великобритания), Red, Orange и Purple2(Япония)

Успешные криптоатаки на роторные системы стали возможны только с появлением ЭВМ в начале 40-х годов

30. Криптоанализ. Методы Касицкого и Фридмана.

Рассмотрим два метода нахождения длины ключевой фразы.

Первый метод был предложен Фридрихом **Касицким**.

Основа метода Касицкого – поиск биграмм. В случае, когда в шифруемом сообщении одна и та же биграмма повторяется на расстоянии, кратном длине ключевой фразы, она встретится на тех же позициях и в зашифрованном тексте. Найдя это расстояние и, получив все его делители, можно получить набор чисел – кандидатов на длину ключевой фразы.

Подсчитав частоты всех букв и биграмм в зашифрованном сообщении, можно предположить, что наиболее частая в нем буква обозначает одну из наиболее часто встречающихся букв алфавита. Аналогично сама частая биграмма зашифрованного текста соответствует, скорее всего, одной из наиболее часто встречающихся биграмм в незашифрованных текстах.

Список наиболее часто встречающихся в русских текстах биграмм в порядке убывания их вероятности: ТО НА НЕ ПО НО ЛА СТ ОН РА АЛ КО ГО КА ЛО НИ ОВ. На статических характеристиках текста основан метод разгадывания любого шифра из группы подстановочных шифров.

Наиболее часто встречающиеся в английских текстах биграммы в порядке убывания их вероятности

Биграмма	Процентное содержание	Биграмма	Процентное содержание
th	3,15	he	2,51
an	1,72	in	1,69
er	1,54	re	1,48
es	1,45	on	1,45
ea	1,31	ti	1,28
at	1,24	St	1,21
en	1,20	nd	1,18

Попробуем расшифровать методом Касицкого следующий текст:

«ОАИТАБНПХЮПМЪАЭМАЗЧАФРЮЯЦМАТВУШКГЮНШИЪДООЯВТХЧЪТЫ
ЖПЫТЕЭНХУАПНХДРСЕЗЬУНЯЗ»

Биграмма МА повторяется на расстоянии в 10 позиций, биграммы ТЫ и НХ на расстоянии в 5 позиций. Скорее всего длина ключевой последовательности равна 5.

Рассмотрим еще один метод определения длины ключа, предложенный **Фридманом**. Суть метода в циклическом сдвиге сообщения. Полученные таким образом сообщения записываются под оригинальным шифротекстом и подсчитывается число совпавших букв в верхней и нижней строке. На основе этих чисел вычисляется так называемый индекс совпадений, равный отношению количества совпадений к полной длине сообщения.

Циклически сдвигая сообщение получаем:

Сдвиг	Совпадений	Индекс
2	0	0
3	5	0.068
4	2	0.027
5	8	0.110
6	1	0.014

7	1	0.014
8	2	0.027

При сдвиге 5 индекс резко возрастает, следовательно длина ключевого слова скорее всего равна 5.

31. Научная криптология. Основы научной криптологии. Работы Шеннона.

К началу 30-х годов окончательно сформировались разделы математики, являющейся научной основой криптологии:

- Теория вероятностей и математическая статистика
- Общая алгебра
- Теория чисел

Начали активно развиваться:

- Теория алгоритмов
- Теория информации
- Кибернетика

Клод **Шеннон** в работе «Теория связи в секретных системах» сформулировал теоретические принципы криптографической защиты информации. Шеннон ввел понятия «рассеивание» и «перемешивание», обосновал возможность создания сколь угодно стойких криптосистем. Важной заслугой Шеннона являются исследования абсолютно криптостойких систем и доказательство их существования, а также существование криптостойких шифров и требуемые для этого условия. Он сформулировал основные требования предъявляемые к надежным шифрам:

- Ключ генерируется для каждого сообщения (каждый ключ используется только один раз)
- Ключ статистически надежен (то есть вероятности появления каждого из возможных символов равны, символы в ключевой последовательности независимы и случайны)
- Длина ключа равна или больше длины сообщения
- Исходный текст обладает некоторой избыточностью (что является критерием оценки правильности расшифровки)

32. Научная криптология. Абсолютно стойкие алгоритмы. Одноразовый блокнот Вернама.

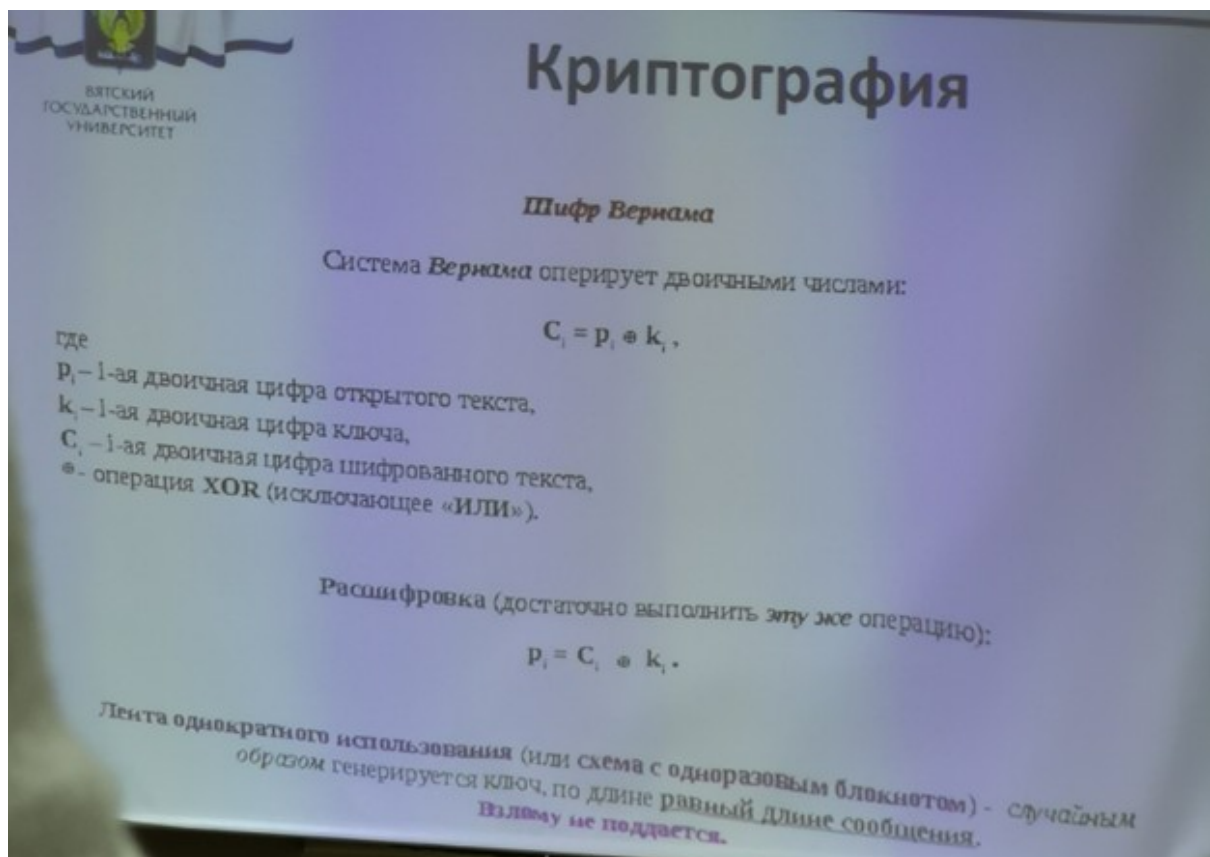
Об **абсолютной стойкости** говорят в случае, если криптосистема не может быть раскрыта ни теоретически, ни практически даже при наличии у атакующего бесконечно больших вычислительных ресурсов. Доказательство существования абсолютно стойких алгоритмов шифрования было выполнено Клодом Шенноном и опубликовано в работе «Теория связи в секретных системах»

Научная криптология и условия абсолютно стойких алгоритмов смотреть в 31 вопросе.

Шенноном было доказано, что примером абсолютного стойкого алгоритма является **шифр Вернама (одноразовый блокнот)**. При этом он считается одной из простейших криптосистем. Других шифров с этим свойством не существует. Шифр Вернама является самой безопасной криптосистемой из всех возможных. (Злоумышленник не может получить никакой информации об открытом тексте). Условия, которым должен

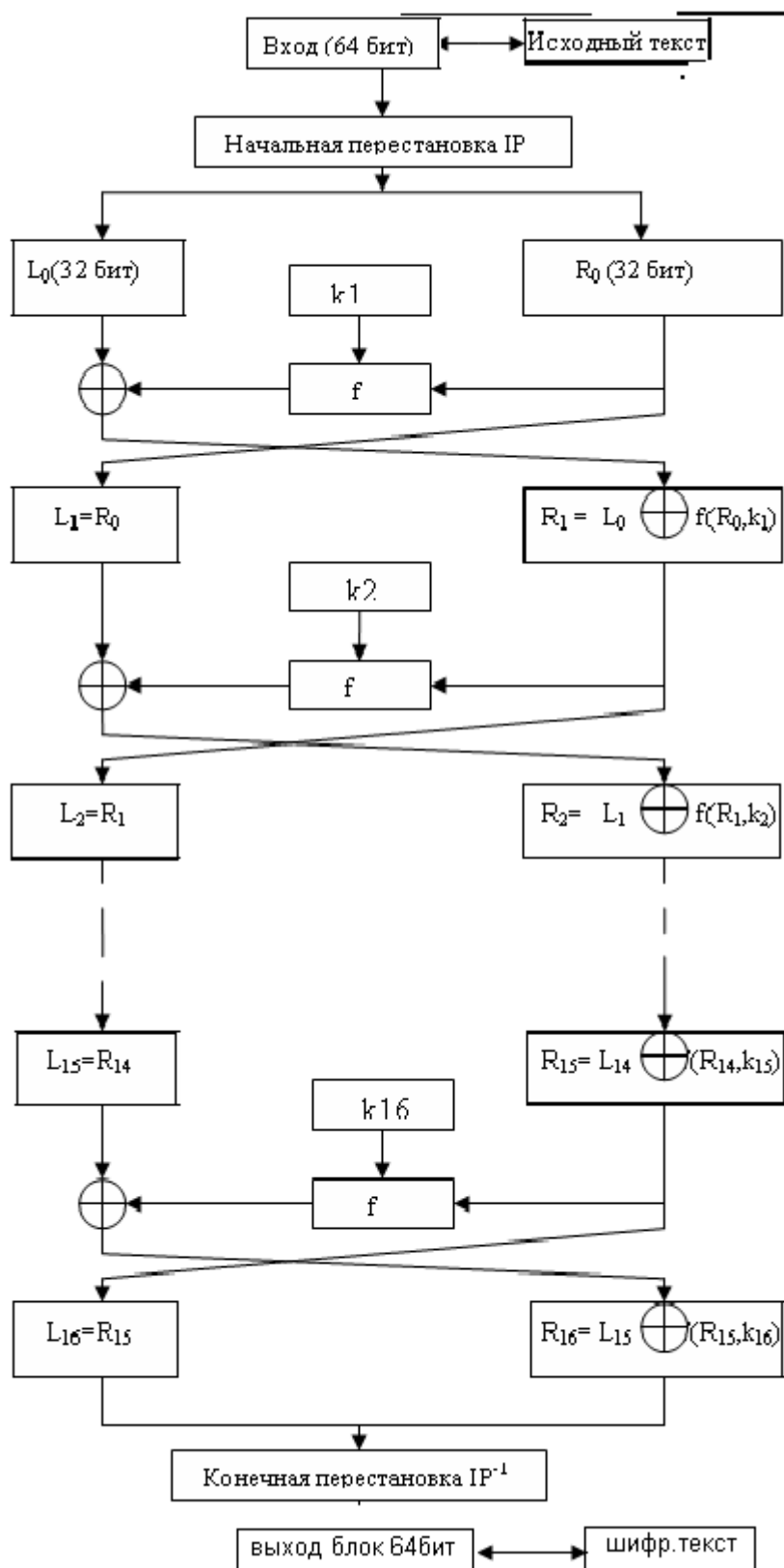
удовлетворять ключ, настолько сильны, что практическое использование шифра Вернама становится трудно осуществимым. Кроме того, чтобы расшифровать сообщение, адресат должен каким-то образом получить тот единственный секретный ключ, который использовался. Передать его по открытым канал связи невозможно – очевидно, что смысл шифрования пропадет. Значит, для его передачи необходимо использовать особый полностью защищенный канал связи. Но если есть такой канал, то зачем тогда передавать зашифрованное сообщение, ведь его длина совпадает с длиной ключа, так что проще передать исходный текст по секретному каналу. Если ключ является не случайным, а псевдослучайным и генерируется по какому-то сложному закону, то противник рано или поздно может определить этот закон и расшифровать все секретные сообщения.

Поэтому данная система используется только для передачи сообщений наивысшей секретности.



33. Компьютерная криптография. Блочные шифры. Алгоритм DES.

DES это блочный шифр, основанный на сети Фейстеля. Шифр имеет размер блока 64 бита и размер ключа 56 бит. Общая схема алгоритма:



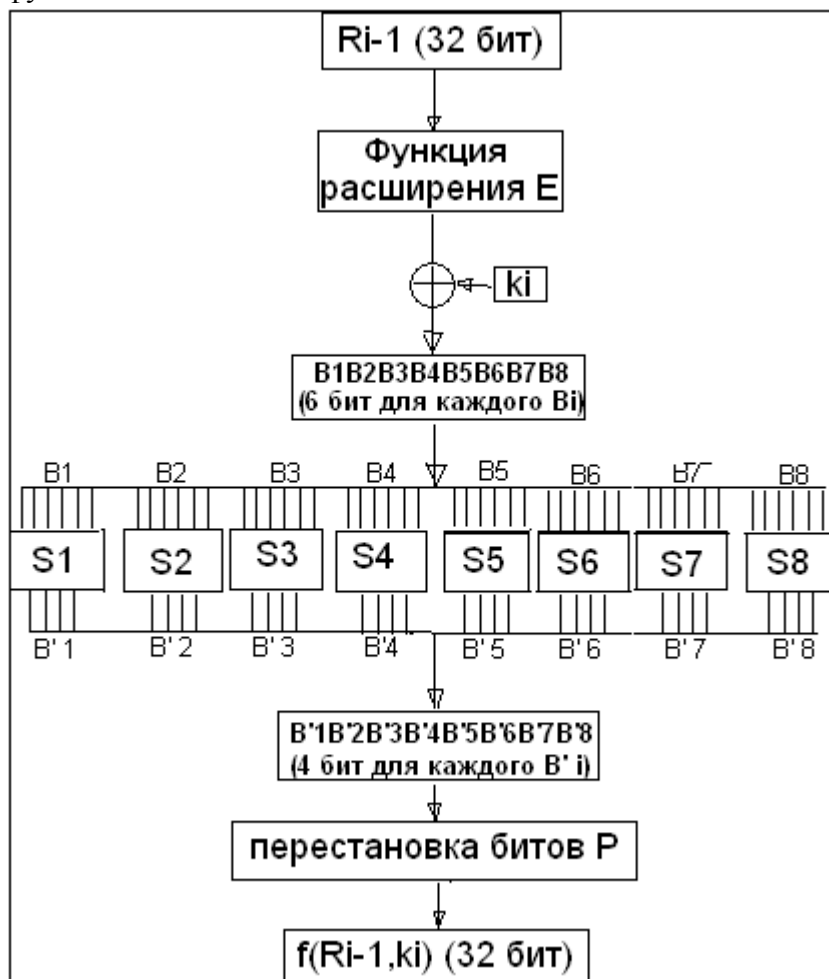
При шифровании над текстом производятся следующие операции:

1. Начальная перестановка бит. На этом этапе биты входного блока перемешиваются в определенном порядке.
2. После этого перемешанные биты разбиваются на две половины, которые поступают на вход функции Фейстеля. Для стандартного DES сеть Фейстеля

включает 16 раундов, но существуют и другие варианты алгоритма.

3. Два блока, полученных на последнем раунде преобразования объединяются и над полученным блоком производится еще одна перестановка.

На каждом раунде сети Фейстеля 32 младших бита сообщения проходят через функцию f :



Рассмотрим операции, выполняющиеся на этом этапе:

1. Входной блок проходит через функцию расширения E, которая преобразует 32-битный блок в блок длиной 48 бит.
2. Полученный блок складывается с раундовым ключом K_i .
3. Результат предыдущего шага разбивается на 8 блоков по 6 бит каждый.
4. Каждый из полученных блоков B_i проходит через функцию подстановки $S\text{-Box}_i$, которая заменяет 6-битную последовательность, 4-битным блоком.
5. Полученный в результате 32-битный блок проходит через перестановку P и возвращается в качестве результата функции f .

Раундовые 48-битные ключи получаются из 56-битного исходного ключа шифра по описанным в стандарте правилам.

Режимы использования DES:

1. Режим электронной кодовой книги: обычное использование DES как блочного шифра. Шифруемый текст разбивается на блоки, при этом каждый блок шифруется отдельно, не взаимодействуя с другими блоками

2. Режим сцепления блоков шифротекста. Каждый очередной блок M_i ($i \geq 1$), перед шифрованием складывается по модулю 2 с предыдущим блоком зашифрованного текста C_{i-1} . Вектор C_0 — начальный вектор, он меняется ежедневно и хранится в секрете
3. Режим обратной связи по шифротексту. В режиме CFB вырабатывается блочная «гамма» $Z_0, Z_1, \dots, Z_i = \text{DES}(C_{i-1}, k_i)$, $C_i = M_i \text{ xor } Z_i$. Начальный вектор C_0 является синхропосылкой и предназначен для того, чтобы разные наборы данных шифровались по-разному с использованием одного и того же секретного ключа. Синхропосылка посылается получателю в открытом виде вместе с зашифрованным файлом
4. Режим обратной связи по выходу. В режиме OFB вырабатывается блочная «гамма» $Z_0, Z_1, \dots, Z_i = \text{DES}(Z_{i-1}, k_i)$, $C_i = M_i \text{ xor } Z_i$, $i \geq 1$

34. Компьютерная криптография. Блочные шифры. Алгоритм ГОСТ 28147-89.

Блочный шифр — разновидность симметричного шифра, оперирующего группами бит фиксированной длины — блоками, характерный размер которых меняется в пределах 64–256 бит. Если исходный текст (или его остаток) меньше размера блока, перед шифрованием его дополняют. Фактически, блочный шифр представляет собой подстановку на алфавите блоков, которая, как следствие, может быть моно- или полиалфавитной.

ГОСТ 28147-89 — блочный шифр с 256-битным ключом и 32 циклами преобразования, оперирующий 64-битными блоками. Основа алгоритма шифра — сеть Фейстеля. Выделяют четыре режима работы ГОСТ 28147-89:

- простой замены
- гаммирование
- гаммирование с обратной связью
- режим выработки имитовставки.

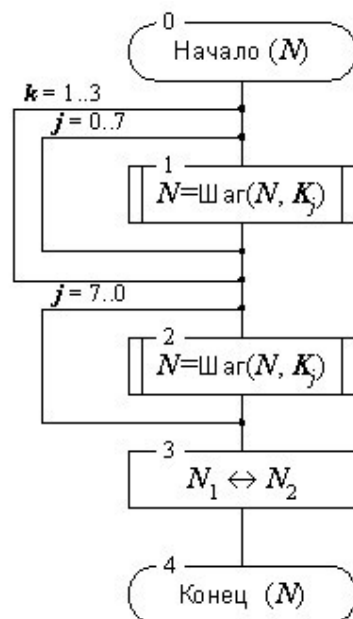
Алгоритм для режима простой замены:

Ниже приведены основные характеристики ключевых структур ГОСТа.

- Ключ является массивом из восьми 32-битовых элементов кода, далее в настоящей работе он обозначается символом K . В ГОСТе элементы ключа используются как 32-разрядные целые числа без знака. Таким образом, размер ключа составляет $32 \cdot 8 = 256$ бит или 32 байта.
- Таблица замен может быть представлена в виде матрицы размера 8×16 или 16×8 , содержащей 4-битовые заменяющие значения. Для языков программирования, в которых двумерные массивы расположены в оперативной памяти по строкам, естественным является первый вариант (8×16), его-то мы и возьмем за основу. Тогда узлы замены будут строками таблицы замен. Далее обозначается символом H . Таким образом, общий объем таблицы замен равен: $8 \text{ узлов} \times 16 \text{ элементов/узел} \times 4 \text{ бита/элемент} = 512 \text{ бит} = 64 \text{ байта}$.

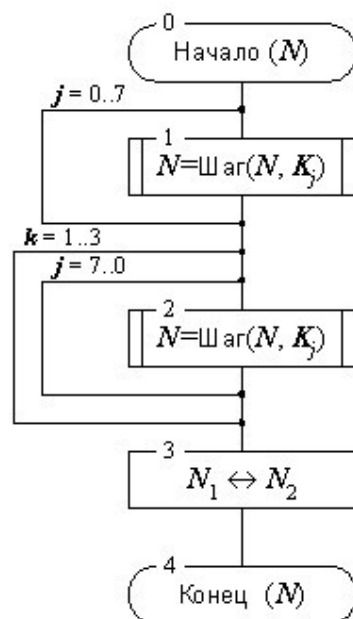
Основной шаг криптопреобразования по своей сути является оператором, определяющим преобразование 64-битового блока данных. Дополнительным параметром этого оператора является 32-битовый блок, в качестве которого используется какой-либо элемент ключа. Схема алгоритма основного шага приведена ниже.

[illegible]



Цикл расшифрования 32-Р:

$K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0$.



Про гаммирование и прочие шляпы:

<http://kaf401.rloc.ru/Criptfiles/gost28147/GOST28147.htm>

35. Компьютерная криптография. Алгоритм RSA.

RSA — криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших целых чисел.

Криптосистема RSA стала первой системой, пригодной и для шифрования, и для цифровой подписи.

Алгоритм создания открытого и секретного ключей

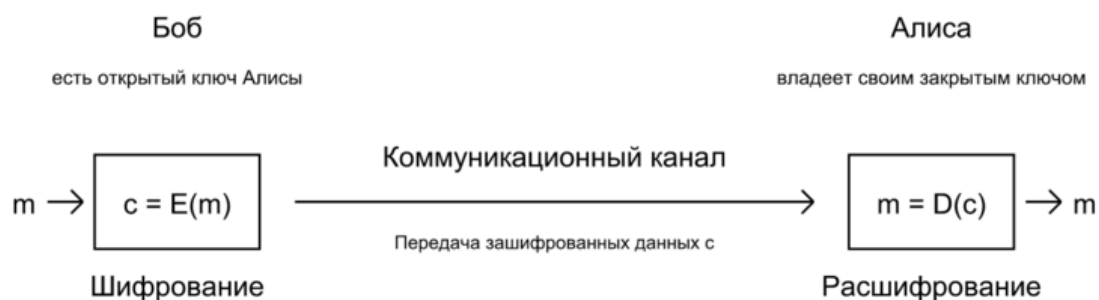
- Выбираются два различных случайных простых числа p и q заданного размера (например, 1024 бита каждое).
- Вычисляется их произведение $n = p \cdot q$, которое называется модулем.
- Вычисляется значение функции Эйлера от числа n : $f(n) = (p-1) \cdot (q-1)$
- Выбирается целое число e ($1 < e < f(n)$), взаимно простое со значением функции $f(n)$
- Вычисляется число d ($d \cdot e \bmod f(n) = 1$) (По расширенному алгоритму Евклида.)
- Пара (e, n) публикуется в качестве открытого ключа.
- Пара (d, n) играет роль закрытого ключа.

Алгоритм шифрования

- Взять открытый ключ (e, n)
- Взять открытый текст m
- Зашифровать сообщение с использованием открытого ключа: $C = m^e \bmod n$

Алгоритм расшифрования

- Принять зашифрованное сообщение C
- Взять свой закрытый ключ (d, n)
- Применить закрытый ключ для расшифрования сообщения: $m = C^d \bmod n$



Надёжность шифрования обеспечивается тем, что третьему лицу (стараящемуся взломать шифр) очень трудно вычислить закрытый ключ по открытому. Оба ключа вычисляются из одной пары простых чисел (p и q). То есть ключи связаны между собой. Но установить эту связь очень сложно.

Основной загвоздкой является декомпозиция модуля n на простые сомножители p и q . Если число является произведением двух очень больших простых чисел, то его очень трудно разложить на множители.

Подробнее тут: <https://neerc.ifmo.ru/wiki/index.php?title=RSA>

36. Хэш-функции. Основные понятия и определения. Требования к хэш-функциям.

- Хэш-функция – функция, преобразующая по детерминированному алгоритму входной массив данных определенной длины (ключ) в выходную битовую строку фиксированной длины (значение).
- Коллизия – это ситуация, когда разным ключам соответствует одно значение хэш-функции

Понятие «хорошей» хэш-функции:

1. функция должна быть простой с вычислительной точки зрения;
2. функция должна распределять ключи в хеш-таблице наиболее равномерно;
3. функция должна минимизировать число коллизий, то есть ситуаций, когда разным ключам соответствует одно значение хэш-функции.

37. Хэш-функции. Метод Деления с остатком. Метод умножения. Универсальное хэширование.

Метод Деления с остатком:

При построение хэш-функции методом деления с остатком ключу k ставится в соответствие остаток от деления k на m , где m — число возможных значений хэш-функции: $h(k) = k \bmod m$. Например, при размере хэш-таблицы $m=12$ и ключе $k=50$ $h(k) = 2$. Некоторых значений основания m следует избегать. Например, если $m = 2^p$, то хэш-функция — это просто p младших битов ключа k . Хорошие результаты дает выбор в качестве m простого числа, далекого от степеней двойки.

Метод умножения:

Пусть количество хэш-значений равно m . Зафиксируем константу A в интервале $0 < A < 1$

$h(k) = [m * \{k * A\}]$, где $\{k * A\}$ - дробная часть $k * A$, $[]$ - взятие целой части числа. Достоинство метода умножения в том, что качество хэш-функции мало зависит от выбора m . Обычно в качестве m выбирают степень двойки, поскольку в большинстве компьютеров умножение на такое m реализуется как сдвиг слова. Кнут предложил в качестве A использовать, число $A = 0,6180339887\dots$ - число, обратное значению золотого сечения.

Универсальное хэширование:

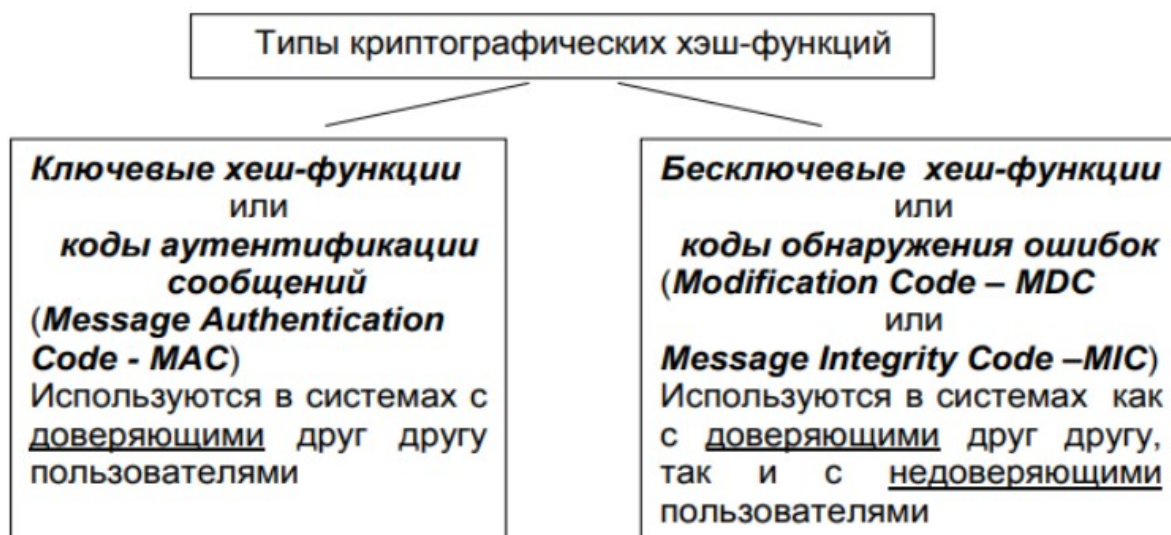
Основная идея универсального хэширования — выбирать хэш-функцию во время исполнения программы случайным образом из некоторого множества. При повторном вызове с теми же входными данными алгоритм будет работать уже по-другому.

При случайном выборе хэш-функции вероятность коллизии между двумя данными ключами должна равняться вероятности совпадения двух случайно выбранных хэш-значении (которая равна $1/m$).

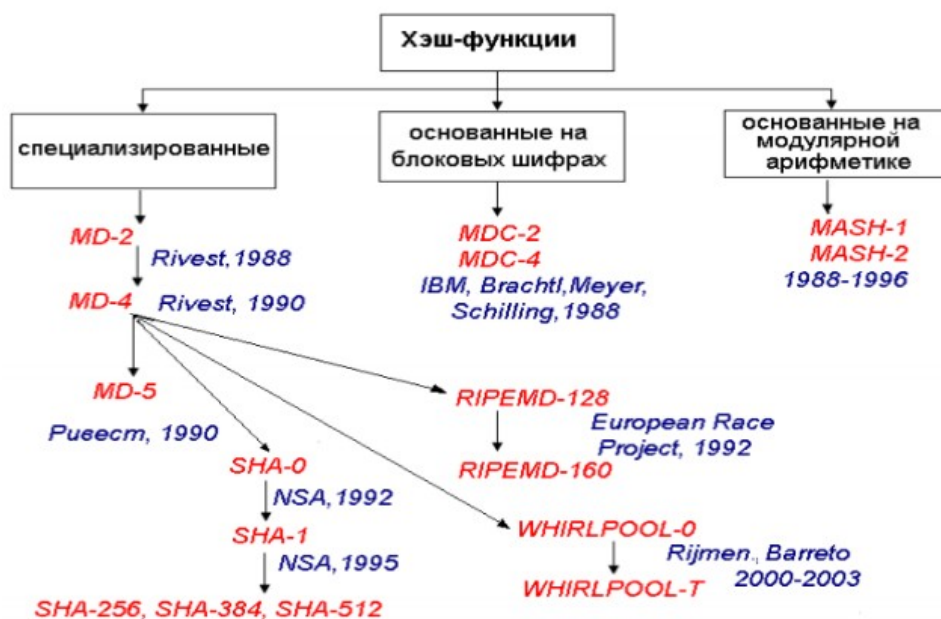
38. Криптографические хэш-функции. Типы криптографических хэш-функций. Сравнение SHA и MD5.

Хэш-функция $h(x)$ называется криптографической, если она удовлетворяет следующим требованиям:

- необратимость: для заданного значения хэш-функции c должно быть сложно определить такой ключ x , для которого $h(x) = c$
- стойкость к коллизиям первого рода: для заданного ключа x должно быть вычислительно невозможно подобрать другой ключ y , для которого $h(x) = h(y)$;
- стойкость к коллизиям второго рода: должно быть вычислительно невозможно подобрать пару ключей x и y , имеющих одинаковый хэш



- **Ключевые хеш-функции** дают возможность без дополнительных средств гарантировать как правильность источника данных, так и целостность данных.
- **Бесключевые хеш-функции** дают возможность с помощью дополнительных средств (шифрования, например) гарантировать целостность данных.



Сравнение SHA и MD5:

- **Безопасность:** наиболее очевидное и наиболее важное различие состоит в том, что дайджест SHA-1 на 32 бита длиннее, чем дайджест MD5. Если предположить, что оба алгоритма не содержат каких-либо структурированных данных, которые уязвимы для криптоаналитических атак, то SHA-1 является более стойким алгоритмом. Используя лобовую атаку, труднее создать произвольное сообщение, имеющее данный дайджест, если требуется порядка 2^{160} операций, как в случае алгоритма SHA-1, чем порядка 2^{128} операций, как в случае алгоритма MD5. Используя лобовую атаку, труднее создать два сообщения, имеющие одинаковый дайджест, если требуется порядка 2^{80} как в случае алгоритма SHA-1, чем порядка 2^{64} операций как в случае алгоритма

MD5.

- *Скорость*: так как оба алгоритма выполняют сложение по модулю 2^{32} , они рассчитаны на 32-битную архитектуру. SHA-1 содержит больше шагов (80 вместо 64) и выполняется на 160-битном буфере по сравнению со 128-битным буфером MD5. Таким образом, SHA-1 должен выполняться приблизительно на 25% медленнее, чем MD5 на той же аппаратуре.
- *Простота и компактность*: оба алгоритма просты и в описании, и в реализации, не требуют больших программ или подстановочных таблиц. Тем не менее, SHA-1 применяет одношаговую структуру по сравнению с четырьмя структурами, используемыми в MD5. Более того, обработка слов в буфере одинаковая для всех шагов SHA-1, в то время как в MD5 структура слов специфична для каждого шага.

Лекция 7: Криптографические хеш-функции.

<https://www.intuit.ru/studies/courses/691/547/lecture/12381?page=1>

39. Электронная цифровая подпись: определение, юридическая сила, требования к ЭЦП, преимущества ЭЦП.

ЭЦП – это аналог обычной подписи, применяют, чтобы придать юридическую силу документации, находящейся на электронном носителе.

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

Юридическая сила электронной цифровой подписи признается при наличии в автоматизированной информационной системе программно-технических средств, обеспечивающих идентификацию подписи, и соблюдении установленного режима их использования.

Федеральный закон "Об электронной подписи" от 06.04.2011 N63-ФЗ регулирует отношения в области использования электронных подписей при совершении гражданско-правовых сделок, оказании государственных и муниципальных услуг, исполнении государственных и муниципальных функций, при совершении иных юридически значимых действий, в том числе в случаях, установленных другими федеральными законами.

К ЭЦП предъявляются два основных требования:

- легкость проверки подлинности подписи
- высокая сложность подделки подписи

Преимущества:

- упрощение и ускорение процесса обмена данными (особенно когда ведется сотрудничество с зарубежными компаниями);
- сокращение расходов, связанных с документооборотом;
- повышение уровня безопасности для информации, носящей коммерческий характер.

40. Электронная цифровая подпись. Принцип работы. Виды ЭЦП. Ключ и сертификат ЭЦП.

ЭЦП - аналог обычной подписи, который применяют, чтобы придать юридическую силу документации, находящейся на электронном носителе.

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

Принцип работы ЭЦП:

Процесс его получения происходит в два этапа. С помощью ПО и математической функции происходит вычисление отпечатка сообщения, обладающего некоторыми особенностями:

1. По отпечатку сообщения невозможно восстановить тело документа;
2. Отпечаток уникален для каждого сообщения и имеет фиксированную длину.

На втором этапе ПО и ключ шифрует отпечаток, расшифровать который можно будет только открытым личным ключом электронной подписи.

Как работает ЭЦП при получении электронного документа:

- При помощи ПО адресат расшифровывает подписанный документ для получения отпечатка исходного документа;
- При помощи ПО и функции хэширования вычисляется отпечаток полученного документа;
- Во время проверки сравниваются отпечатки полученного и исходного электронного документа.

Если отпечатки равны, то ЭЦП подтверждает целостность и верность документа. Если во время пересылки в тело документа были внесены изменения и нарушена его целостность, то это будет отображено.

Определение подлинности информации реализуется путем установки факта, что полученные данные были отправлены подписавшим документ с помощью электронной цифровой подписи, и то что данные не были искажены.

Виды ЭЦП:

- простая подтверждает, что электронное сообщение отправлено конкретным лицом. Сообщение приравнивается к бумажному документу, если стороны заранее об этом договорились, а также в предусмотренных законом случаях.
- усиленная неквалифицированная позволяет идентифицировать отправителя и подтвердить, что документ никто не изменял. Сообщение приравнивается к бумажному документу, если стороны заранее об этом договорились, а также в предусмотренных законом случаях.
- усиленная квалифицированная (дополнительно к «У не К») подтверждается сертификатом, выданным аккредитованным удостоверяющим центром. Сообщение с УК со всех случаях приравнивается к бумажному документу с собственноручной подписью.



С понятием ЭЦП тесно связаны два других: ключ и сертификат электронной подписи.

Сертификат является электронным (и/или бумажным) документом:

- выдаётся на ФИО конкретного человека (должностного лица) содержит персональные данные;
- подписывается ЭП Удостоверяющего центра, который тем самым подтверждает его действительность;
- сертификат в себе содержит открытый ключ Пользователя (поэтому открытый ключ называют сертификатом).

Сертификат подтверждает, что ЭП принадлежит конкретному лицу. Он бывает усиленным и обычным.

Ключ - это символы, находящиеся в последовательности. Обычно они используются парой. Первый это сама подпись, другой подтверждает, что она подлинная. Для подписи каждого вновь создаваемого документа, формируется новый ключ.

Ключевая пара состоит из двух частей: открытой и закрытой. Оба этих ключа выдаются и создаются удостоверяющими центрами с помощью специальной программы шифрования.

Закрытый ключ - уникальная последовательность символов, предназначенная для создания ЭП и для расшифровки сообщений. Это частная, приватная информация, которая известна только ее владельцу.

Закрытый ключ генерируется на рабочем месте пользователя и сохраняется (только у пользователя) на съемный носитель (флешка, токен, смарт карта) или в реестр Windows. Такой закрытый ключ необходимо хранить в секретном месте со всеми мерами предосторожностей.

На основе закрытого ключа создается открытый ключ (обратный процесс здесь невозможен, так как подобрать закрытый ключ по открытому ключу нельзя).

Открытый ключ - уникальная последовательность символов, предназначенная для проверки подлинности ЭП. Это открытая, общеизвестная информация доступна любому пользователю системы электронного документооборота (ЭДО).

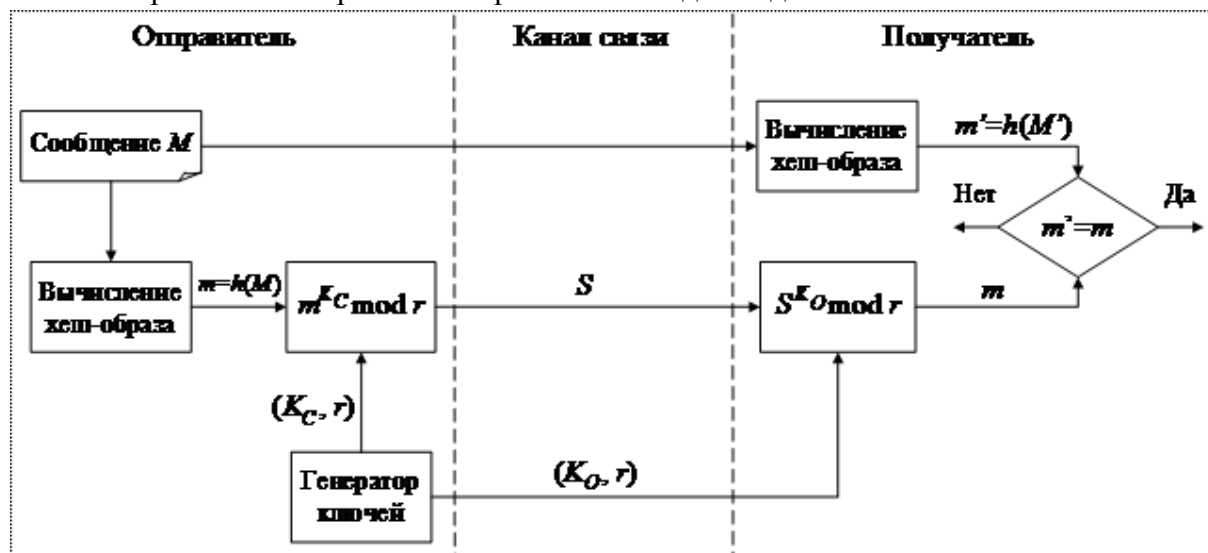
41. Электронная цифровая подпись. Алгоритмы в основе ЭЦП.

Система ЭЦП на основе RSA.

Сначала нужно вычислить пару ключей.

Отправитель (автор) электронных документов вычисляет два больших простых числа P и Q , затем находит произведение и значение функции:

- $N = P * Q$; $\phi(N) = (P-1)(Q-1)$.
- Затем отправитель вычисляет число E из условий:
 $E < \phi(N)$, $\text{НОД}(E, \phi(N)) = 1$
- и вычисляется число D , мультипликативно обратное к числу E по модулю $\phi(N)$, то есть число, удовлетворяющее сравнению:
 $E * D \equiv 1 \pmod{\phi(N)}$ (D называется *секретной экспонентой*; обычно оно вычисляется при помощи расширенного алгоритма Евклида);
- Пара чисел (E, N) является открытым ключом. Такую пару автор передает партнерам по переписке для проверки его цифровых подписей. Число D сохраняется автором как секретный ключ для подписания.



Алгоритм цифровой подписи Эль Гамала (EGSA)

Основная идея обоснована на практической невозможности фальсификации цифровой подписи. Для этого нужна более сложная вычислительная задача, чем разложение на множители большого целого числа, например вычисление дискретного логарифма в кольце. Также Эль Гамалу удалось избежать слабости алгоритма ЭЦП RSA, связанной с подделкой ЭЦП без определения секретного ключа.

Чтобы генерировать пару ключей, нужно выбрать простое целое число P и G , причем $G < P$.

Получатель и отправитель подписанного документа используют одинаковые большие числа P ($\sim 10^{308} = \sim 2^{1024}$) и G ($\sim 10^{154} = \sim 1^{512}$) которые не секретные.

Отправитель выбирает случайное целое число X , $1 < X < (P - 1)$ и вычисляет: $Y = G^X \text{ mod } P$;

Число Y является открытым ключом, который используется для проверки подписи

отправителя. Число X является секретным ключом отправителя для подписи документов.

Чтобы подписать сообщение M , сначала нужно чтобы отправитель захэшировал его с помощью хэш функции h в целое число m : $m = h(M)$, $1 < m < (P - 1)$ и сгенерировал случайное целое число K : $1 < K < (P - 1)$, такое, что K и $(P - 1)$ будут взаимно простыми.

Потом отправитель вычисляет целое число a : $a = G^K \bmod P$, используя расширенный алгоритм Евклида, вычисляет с помощью секретного ключа X целое число b : $m = X * a + K * b \pmod{P - 1}$;

Пара чисел (a, b) образуют цифровую подпись S : $S = (a, b)$;

Тройка чисел (M, a, b) транспортируется получателю, в то время как пара чисел (X, K) держится в секрете. Получатель получив сообщение (M, a, b) , должен вычислить число m : $m = h(M)$, затем получатель вычисляет: $A = Y^a a^b \bmod (P)$ и признает сообщение M подлинным, если $A = G^m \bmod (P)$.

Можно строго математически доказать, что последнее равенство будет равно тогда, когда подпись S под документом M получена с помощью именно секретного ключа X , из которого был получен открытый ключ Y .

Алгоритм цифровой подписи DSA

DSA (Digital Signature Algorithm) - это развитие алгоритмов цифровой подписи Эль Гамала и К.Шнорра.

Получатель и отправитель электронного документа реализуют при вычислении большие целые числа G и P простые числа L бит каждое ($512 < L < 1024$), q - простое число длиной 160 бит (делитель числа $(P - 1)$).

Числа P , G , q открытые и могут быть общими для пользователей.

Отправитель берет случайное целое число X , $1 < X < q$.

Число X - секретный ключ отправителя для создания ЭЦП.

Отправитель вычисляет:

$Y = G^X \bmod P$. число Y - открытый ключ.

Чтобы подписать документ M , отправитель хэширует его в целое хэш значение m :

$m = h(M)$, $1 < m < q$, потом генерирует случайное целое число K , $1 < K < q$, и вычисляет:

$r = (G^K \bmod P) \bmod q$.

Также нужно вычислить: $s = ((m + r * X) / K) \bmod q$;

Пара чисел $S = (r, s)$ образуют цифровую подпись.

Получатель проверяет выполнение условий: $0 < r < q$, $0 < s < q$.

Если хоть одно условие не выполнено, то подпись нужно отвергнуть.

Если же выполнены все условия, то получатель вычисляет:

$$w = (l/s) \bmod q, \text{ хэш значения } m = h(M) \text{ и числа } u_1 = (m * w) \bmod q, u_2 = (r * w) \bmod q.$$

Затем получатель с помощью открытого ключа Y вычисляет:

$$v = ((G^{u_1} * Y^{u_2}) \bmod P) \bmod q;$$

Если условие $v = r$ выполняется, тогда подпись S под документом подлинная.

42. Что такое DoS-атака? Перечислите группы методов обнаружения DoS-атак. Перечислите меры противодействия DoS-атакам. Назовите причины, из-за которых может возникнуть DoS-условие

DoS-атака — атака на вычислительную систему с целью довести её до отказа.

группы методов обнаружения DoS-атак.

- сигнатурные — основанные на качественном анализе трафика.
- статистические — основанные на количественном анализе трафика.
- гибридные (комбинированные) — сочетающие в себе достоинства обоих вышеперечисленных методов.

меры противодействия DoS-атакам.

- Предотвращение. Профилактика причин, побуждающих тех или иных лиц организовывать и предпринять DDoS-атаки. (Очень часто кибератаки вообще являются следствиями личных обид, политических, религиозных и иных разногласий, провоцирующего поведения жертвы и т. п.). Нужно вовремя устранить причины DDoS-атак, после этого сделать выводы, чтобы избежать таких атак в будущем.
- Ответные меры. Применяя технические и правовые меры, нужно как можно активнее воздействовать на источника и организатора DDoS-атаки. В настоящее время даже существуют специальные фирмы, которые помогают найти не только человека, который провел атаку, но даже и самого организатора.
- Программное обеспечение. На рынке современного программного и аппаратного обеспечения существует и такое, которое способно защитить малый и средний бизнес от слабых DDoS-атак. Эти средства обычно представляют собой небольшой сервер.
- Фильтрация и блэкхолинг. Блокирование трафика, исходящего от атакующих машин. Эффективность этих методов снижается по мере приближения к объекту атаки и повышается по мере приближения к атакующей машине. В этом случае фильтрация может быть двух видов: использование межсетевых экранов и списков ACL. Использование межсетевых экранов блокирует конкретный поток трафика, но не позволяет отделить «хороший» трафик от «плохого». ACL списки фильтруют второстепенные протоколы и не затрагивают протоколы TCP. Это не замедляет скорость работы сервера, но бесполезно в том случае, если злоумышленник использует первостепенные запросы.[39]
- Обратный DDoS — перенаправление трафика, используемого для атаки, на атакующего. При достаточной мощности атакующего сервера позволяет не только успешно отразить атаку, но и вывести из строя сервер атакующего.
- Устранение уязвимостей. Не работает против флуд-атак, для которых

«уязвимостью» является конечность тех или иных системных ресурсов. Данная мера нацелена на устранение ошибок в системах и службах.

- Наращивание ресурсов. Абсолютной защиты, естественно, не дает, но является хорошим фоном для применения других видов защиты от DDoS-атак.
- Рассредоточение. Построение распределенных и дублирование систем, которые не прекратят обслуживать пользователей, даже если некоторые их элементы станут недоступны из-за DoS-атаки.
- Уклонение. Увод непосредственной цели атаки (доменного имени или IP-адреса) подальше от других ресурсов, которые часто также подвергаются воздействию вместе с непосредственной целью атаки.
- Активные ответные меры. Воздействие на источники, организатора или центр управления атакой, как техногенными, так и организационно-правовыми средствами.
- Использование оборудования для отражения DDoS-атак. Например, DefensePro® (Radware), SecureSphere® (Imperva), Периметр (МФИ Софт), Arbor Peakflow®, Riorey, Impletac iCore и от других производителей. Устройства развёртываются перед серверами и маршрутизаторами, фильтруя входящий трафик.
- Приобретение сервиса по защите от DDoS-атак. Актуально в случае превышения флудом пропускной способности сетевого канала.

Причины, из-за которых может возникнуть DoS-условие.

- Ошибка в программном коде, приводящая к обращению к неиспользуемому фрагменту адресного пространства, выполнению недопустимой инструкции или другой необрабатываемой исключительной ситуации, когда происходит аварийное завершение программы-сервера — серверной программы. Классическим примером является обращение по нулевому (англ. null) адресу.
- Недостаточная проверка данных пользователя, приводящая к бесконечному либо длительному циклу или повышенному длительному потреблению процессорных ресурсов (вплоть до исчерпания процессорных ресурсов) либо выделению большого объёма оперативной памяти (вплоть до исчерпания доступной памяти).

43. Что такое DDoS-атака? Назовите два типа DDoS атак Как классифицируются DDoS-атаки?

DDoS-атака - DoS-атака, выполняемая одновременно с большого числа компьютеров

Назовите два типа DDoS атак

- DDoSLayer3&4 по модели OSI. Одна из характеристик данной атаки – большое количество пакетов, которыми атакуется ресурс. На данный момент средняя мощность атаки по миру –9,7 Gb/s и 19 Mpps.
- DDoSLayer7 по модели OSI, то есть атака на уровень приложений. Как правило, атака не содержит большое количество пакетов (на порядки ниже, чем при DDoSL3&4), скорее характеризуется точечным ударом по слабому месту атакуемого сайта.

Как классифицируются DDoS-атаки?

- Насыщение полосы пропускания
- Атаки на уровне протоколов
- Атаки на уровне приложений

44. Что такое атаки на уровне протоколов? Что такое атаки на уровне приложений?

Атаки на уровне протоколов. Как и следует из названия, атаки этого типа используют ограничения и уязвимости различных сетевых протоколов. Они «бомбардируют» сервер паразитными пакетами, и он становится неспособным обработать запросы легальных пользователей. В качестве примера можно привести SYN-flood, teardrop и другие атаки, нарушающие нормальное движение пакетов внутри протокола на разных стадиях.

Атаки на уровне приложений, которые нарушают нормальное функционирование системы, используя уязвимости и слабые места приложений и операционных систем. Эти атаки незаметны для стандартных анализаторов, так как составляют порой до 1 Kpps. Стандартные меры защиты не могут выявить столь мелкий всплеск трафика, следовательно для защиты требуется всегда постоянная фильтрация и комплекс очистки всегда должен знать алгоритмы работы самого приложения.

45. Что такое Флуд (англ. flood)? Какова основная цель использования флуда? Как

защитится от флуда?

Флуд - лавина пустых, бессмысленных запросов того или иного уровня, которые принимающий узел вынужден обрабатывать

Какова основная цель использования флуда?

Полностью забить канал связи, насытить полосу пропускания

Как защитится от флуда?

- Увеличить производительность оборудования
- Использовать системы фильтрации трафика до передачи его на сервер
- Установка квот на использование ресурсов
- Рассредоточение ресурсов по независимым серверам

46. Как защититься от перегрузки аппаратных ресурсов? Что такое Storm?

Как защититься от перегрузки аппаратных ресурсов?

- Увеличить производительность оборудования и объем дискового пространства. При работе сервера в штатном режиме свободными должны оставаться не менее 25-30% ресурсов.
- Задействовать системы анализа и фильтрации трафика до передачи его на сервер.
- Лимитировать использование аппаратных ресурсов компонентами системы

- (установить квоты).
- Хранить лог-файлы сервера на отдельном накопителе.
- Рассредоточить ресурсы по нескольким независимым друг от друга серверам. Так, чтобы при отказе одной части другие сохраняли работоспособность.

Storm - рассылка бессмысленных URL-запросов

47. Перечислите методы противодействия эксплуатации уязвимостей в софте.

- Своевременно устанавливать обновления, закрывающие уязвимости операционных систем и приложений.
- Изолировать от стороннего доступа все службы, предназначенные для решения административных задач.
- Использовать средства постоянного мониторинга работы ОС сервера и программ (поведенческий анализ и т. п.).
- Отказаться от потенциально уязвимых программ (бесплатных, самописных, редко обновляемых) в пользу проверенных и хорошо защищенных.
- Использовать готовые средства защиты систем от DoS и DDoS-атак, которые существуют как в виде аппаратных, так и программных комплексов.

48. Основные схемы подключения межсетевых экранов. Схемы защиты сети с использованием экранирующего маршрутизатора. Схемы с защищаемой закрытой и не защищаемой открытой подсетями.

Межсетевой экран (МСЭ)—это устройство обеспечения безопасности сети, которое осуществляет мониторинг входящего и исходящего сетевого трафика и на основании установленного набора правил безопасности принимает решения, пропустить или заблокировать конкретный трафик.

Схемы защиты сети с использованием экранирующего маршрутизатора

Экранирующий маршрутизатор предназначен для фильтрации пакетов сообщений и обеспечивает прозрачное взаимодействие между внутренней и внешней сетями. Функционирует на сетевом уровне эталонной модели OSI.

Решение о том, пропустить или отбраковать данные, принимается для каждого пакета независимо на основе заданных правил фильтрации.

Схема защиты:

Открытая внешняя сеть <-> Экранирующий маршрутизатор <-> Защищаемая внут. сеть

Экранирующий маршрутизатор выполняет фильтрацию пакетов по содержимому их заголовков, анализируя следующие поля:

- адрес отправителя;
- адрес получателя;
- тип пакета;
- флаг фрагментации пакета;
- номер порта источника;
- номер порта получателя.

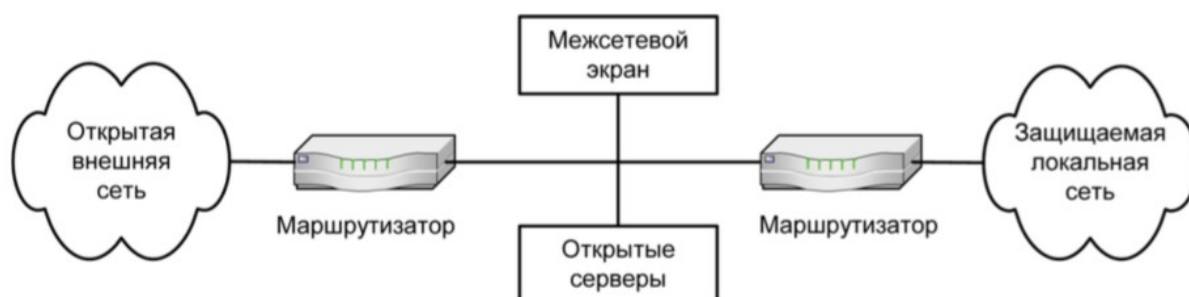
При обработке каждого пакета экранирующий маршрутизатор последовательно

просматривает заданную таблицу правил, пока не найдет правила, с которым согласуется полная ассоциация пакета (параметры). Если экранирующий маршрутизатор получил пакет, не соответствующий ни одному из табличных правил, он применяет правило, заданное по умолчанию (обычно отбрасывает пакет).

Схемы с защищаемой закрытой и не защищаемой открытой подсетями

Схема защиты:

Открытая внешняя сеть <-> Маршрутизатор <-> Межсетевой экран + открытая подсеть <-> Маршрутизатор <-> Защищаемая локальная сеть



Открытая подсеть - сеть, доступ к которой для потенциально враждебной стороны частично открыт. В нее могут входить HTTP, FTP, SMTP и другие серверы.

49. Основные схемы подключения межсетевых экранов. Схемы с раздельной защитой, закрытой и открытой подсетей. Схемы единой защиты локальной сети.

Схемы с раздельной защитой закрытой и открытой подсетей

Такая схема может быть построена на основе одного МЭ (межсетевой экран) с тремя сетевыми интерфейсами или на основе двух МЭ с двумя сетевыми интерфейсами. В обоих случаях доступ к открытой и закрытой подсетям локальной сети возможен только через межсетевой экран. При этом доступ к открытой подсети не позволяет осуществить доступ к закрытой подсети.

Из этих двух схем большую степень безопасности межсетевых взаимодействий обеспечивает схема с двумя МЭ, каждый из которых образует отдельный эшелон защиты закрытой подсети. Защищаемая открытая подсеть здесь выступает в качестве экранирующей подсети

Обычно экранирующую подсеть конфигурируют таким образом, чтобы обеспечить доступ к компьютерам подсети как из потенциально враждебной внешней сети, так и из закрытой подсети локальной сети. Однако прямой обмен информационными пакетами между внешней сетью и закрытой подсетью невозможен. При атаке системы с экранирующей подсетью необходимо преодолеть по крайней мере две независимые линии защиты, что является весьма сложной задачей.

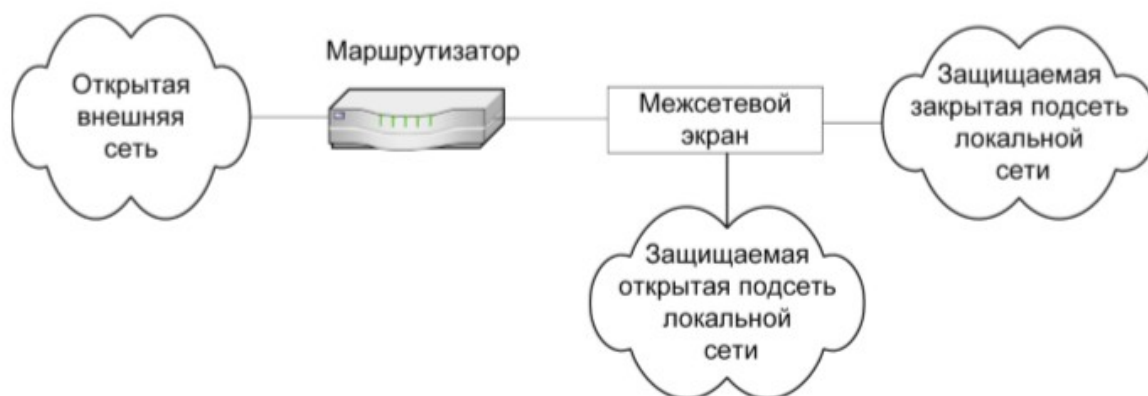


Рис. 9. Схема с раздельной защитой закрытой и открытой подсетей на основе одного МЭ с тремя сетевыми интерфейсами

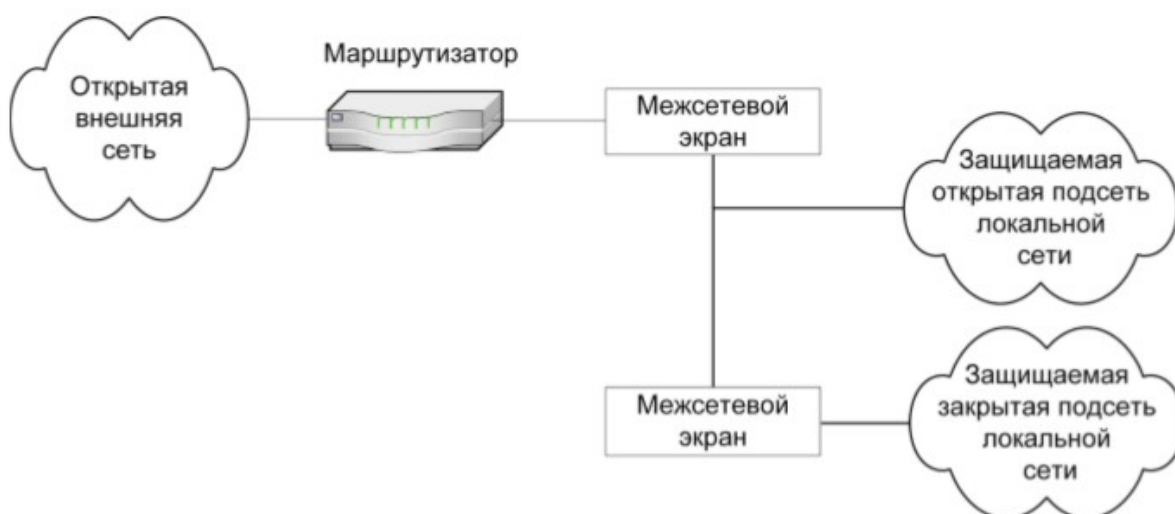


Рис. 10. Схема с раздельной защитой закрытой и открытой подсетей на основе двух МЭ с двумя сетевыми интерфейсами

Схемы единой защиты локальной сети

Данная схема является наиболее простым решением, при котором МЭ целиком экранирует локальную сеть от потенциально враждебной внешней сети. Между маршрутизатором и МЭ имеется только один путь, по которому идет весь трафик. Данный вариант МЭ реализует политику безопасности, основанную на принципе «запрещено все, что явно не разрешено»; при этом пользователю недоступны все службы, кроме тех, для которых определены соответствующие полномочия. Обычно маршрутизатор настраивается таким образом, что МЭ является единственным видимым снаружи сетевым устройством.



Рис. 7. Схема единой защиты локальной сети

50. Политика работы межсетевого экрана. Принцип «запрещено все, что явно не разрешено» и Принцип «разрешено все, что явно не запрещено»

Политика работы межсетевого экрана задает базовый принцип управления межсетевым взаимодействием, положенный в основу функционирования МЭ.

Может быть выбран один из двух таких принципов:

- запрещено все, что явно не разрешено;
- разрешено все, что явно не запрещено.

Фактически выбор принципа устанавливает, насколько «подозрительной» или «доверительной» должна быть система защиты.

При выборе принципа **«запрещено все, что явно не разрешено»** межсетевой экран настраивается таким образом, чтобы блокировать любые явно неразрешенные межсетевые взаимодействия. Данный принцип соответствует классической модели доступа, используемой во всех областях информационной безопасности. Такой подход позволяет адекватно реализовать принцип минимизации привилегий, поэтому с точки зрения безопасности он является лучшим. Администратор безопасности должен на каждый тип разрешенного взаимодействия задавать одно и более правил доступа. Администратор не сможет по забывчивости оставить разрешенными какие-либо полномочия, так как по умолчанию они будут запрещены. Доступные лишние сервисы могут быть использованы во вред безопасности, что особенно характерно для закрытого и сложного программного обеспечения, в котором могут быть различные ошибки и некорректности. Следует отметить, что правила доступа, сформулированные в соответствии с этим принципом, могут доставлять пользователям определенные неудобства.

При выборе принципа **«разрешено все, что явно не запрещено»** межсетевой экран настраивается таким образом, чтобы блокировать только явно запрещенные межсетевые взаимодействия. В этом случае повышается удобство использования сетевых сервисов со стороны пользователей, но снижается безопасность межсетевого взаимодействия. Пользователи имеют больше возможностей обойти межсетевой экран, например, могут получить доступ к новым сервисам, не запрещенным политикой (или даже не указанным в политике), или запустить неразрешенные сервисы на нестандартных портах TCP/UDP, которые не запрещены политикой. Администратор может учесть не все действия, которые запрещены пользователям. Ему приходится работать в режиме реагирования, предсказывая и запрещая те межсетевые взаимодействия, которые отрицательно воздействуют на безопасность сети. При

реализации данного принципа внутренняя сеть оказывается менее защищенной от нападений хакеров. Поэтому производители межсетевых экранов обычно отказываются от использования данного принципа.

51. Технология NAT. Достоинства и недостатки NAT-технологии.

NAT - механизм в сетях TCP/IP, позволяющий изменять IP-адрес в заголовке пакета, проходящего через устройство маршрутизации трафика.

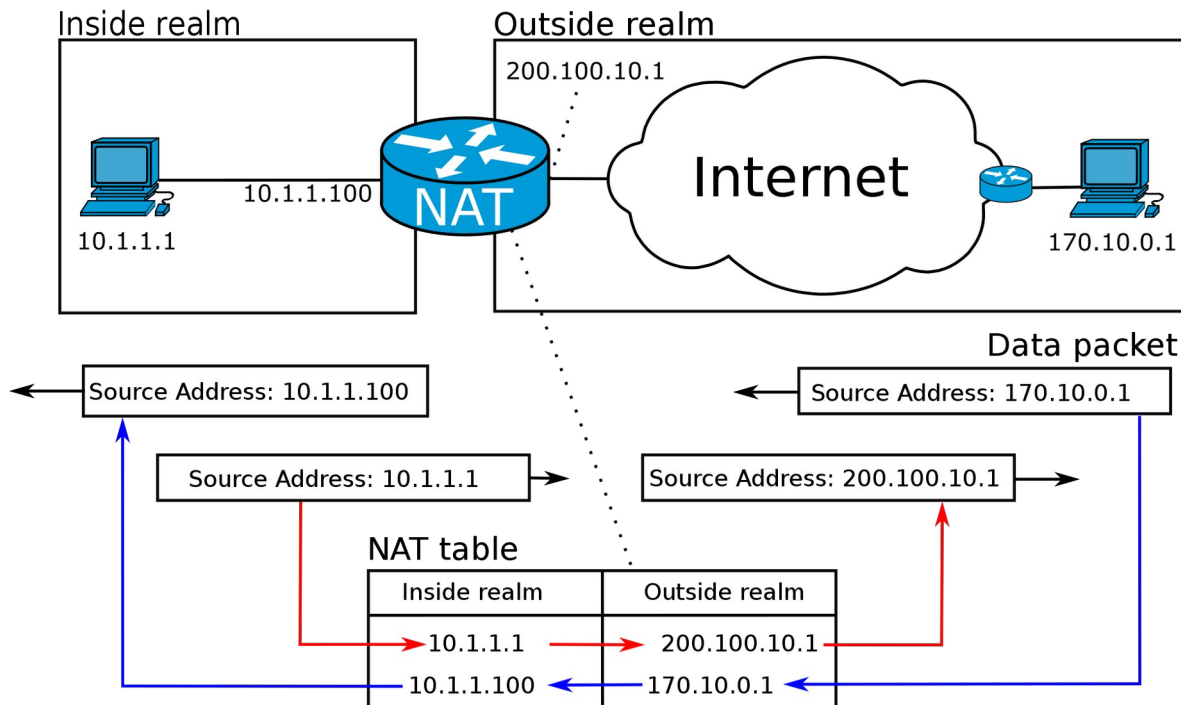
Функции (достоинства) NAT:

- Позволяет сэкономить IP-адреса, транслируя несколько внутренних IP-адресов в один внешний публичный IP-адрес
- Позволяет предотвратить или ограничить обращение снаружи к внутренним хостам, оставляя возможность обращения из внутренней сети во внешнюю
- Позволяет скрыть определенные внутренние сервисы внутренних хостов/серверов

Недостатки NAT:

- Не все протоколы могут «преодолеть» NAT
- Из-за трансляции адресов «много в один» появляются дополнительные сложности с идентификацией пользователей и необходимость хранить полные логи трансляций
- Атака DoS со стороны узла, осуществляющего NAT – если NAT используется для подключения многих пользователей к одному и тому же сервису, это может вызвать иллюзию DoS-атаки на сервис

Пикча



52. Классификация угроз, реализуемых по сети. Атаки в сетях на основе стека протоколов TCP/IP. Защита от sniffеров.

Классификация угроз, реализуемых по сети

1. характер угрозы:

- Пассивная – угроза, которая не оказывает влияния на работу

информационной системы, но может нарушить правила доступа к защищаемой информации. Пример: использование снифферов для "прослушивания" сети.

- Активная – угроза, которая воздействуют на компоненты информационной системы, при реализации которой оказывается непосредственное влияние на работу системы. Пример: DDOS-атака в виде шторма ТСР-запросами.
2. цель реализации угрозы: конфиденциальность, доступность, целостность информации.
 3. условие начала атаки:
 - по запросу от атакуемого. То есть злоумышленник ожидает передачи запроса определенного типа, который и будет условием начала НСД.
 - по наступлению ожидаемого события на атакуемом объекте.
 - безусловное воздействие – злоумышленник ничего не ждет, то есть угроза реализуется сразу и безотносительно к состоянию атакуемого объекта.
 4. наличие обратной связи с атакуемым объектом:
 - с обратной связью, то есть на некоторые запросы злоумышленнику необходимо получить ответ. Таким образом, между атакуемым и атакующим есть обратная связь, позволяющая злоумышленнику следить за состоянием атакуемого объекта и адекватно реагировать на его изменения.
 - без обратной связи – соответственно, нет обратной связи и необходимости злоумышленнику реагировать на изменения атакуемого объекта.
 5. расположение нарушителя относительно атакуемой информационной системы: внутрисегментный и межсегментные. Сегмент сети – физическое объединение хостов, технических средств и других компонентов сети, имеющих сетевой адрес. Например, один сегмент образуют компьютеры, подключенные к общей шине на основе Token Ring.
 6. уровень эталонной модели ISO/OSI, на котором реализуется угроза: физический, канальный, сетевой, транспортный, сеансовый, представительный, прикладной.

Атаки в сетях на основе стека протоколов TCP/IP

Протокол	Уязвимость
FTP	<ul style="list-style-type: none">● Аутентификация на базе открытого текста (пароли пересылаются в незашифрованном виде) aka basic authentication● Доступ по умолчанию● Наличие двух открытых портов
Telnet	Аутентификация на базе открытого текста (пароли пересылаются в незашифрованном виде)

UDP	Отсутствие механизма предотвращения перегрузок буфера
TCP	Отсутствие механизма проверки корректности заполнения служебных заголовков пакета
DNS	Отсутствие средств проверки аутентификации полученных данных от источника
SMTP	Отсутствие поддержки аутентификации заголовков сообщений

Защита от снифферов:

- Сильная аутентификация, например, использование одноразовых паролей
- Антиснифферы – аппаратные или программные средства, способные выявить работу сниффера в сегменте сети.
- Коммутируемая инфраструктура.
- Криптографические методы.

53. Спуфинг и антиспуфинг.

IP-спуфинг - подмена доверенного объекта сети. Под доверенным в данном случае понимается объект сети (компьютер, маршрутизатор, межсетевой экран и т.п.), легально подключенный к серверу.

Суть угрозы заключается в том, что злоумышленник выдает себя за доверенный объект сети. Это можно сделать двумя способами. Во-первых, воспользоваться IP-адресом, находящимся в пределах диапазона санкционированных IP-адресов. Во-вторых, авторизованным внешним адресом, которому разрешается доступ к определенным сетевым ресурсам. Атаки данного типа часто являются отправной точкой для прочих атак.

Спуфинг может быть использован злоумышленником для обхода настроек межсетевых экранов, а также для организации DoS-атак по отношению к третьим лицам.

Антиспуфинг – это фильтр для защиты от спама, который блокирует сообщения, отправленные с одного из локальных доменов, но с неавторизованного IP-адреса

Для ослабления угрозы (но не ее ликвидации) можно использовать следующее:

- Контроль доступа. Можно настроить контроль доступа на отсечение любого трафика, поступающего из внешней сети с исходным адресом внутри сети. Этот метод является действенным, если санкционированы только внутренние адреса и не работает, если есть санкционированные внешние адреса.
- Фильтрация RFC 2827 – данный тип фильтрации позволяет пресечь попытки спуфинга чужих сетей пользователями вашей сети. Для этого необходимо отбраковывать любой исходящий трафик, исходный адрес которого не является одним из IP-адресов вашей организации. Часто этот тип фильтрации выполняется провайдером. В результате отбраковывается весь трафик, который не имеет исходного адреса, ожидаемого на определенном интерфейсе. К примеру, если ISP предоставляет соединение с IP-адресом 15.1.1.0/24, он может настроить фильтр таким образом, чтобы с данного интерфейса на маршрутизатор ISP допускался только трафик, поступающий с адреса 15.1.1.0/24. Заметим, что до тех пор, пока все провайдеры не внедрят этот тип фильтрации, его эффективность будет намного ниже возможной.

- Внедрение дополнительных методов аутентификации. IP-спуфинг возможен только в случае аутентификации на основе IP. Если ввести какие-то дополнительные меры по аутентификации, например, криптографические, атака становится бесполезной.

54. Преобразование адресов при использовании функции NAT. Три базовые концепции трансляции адресов. Четыре типа трансляции сетевых адресов.

Пример действия механизма NAT:

1. Пользователь корпоративной сети отправляет запрос в Интернет, который поступает на внутренний интерфейс маршрутизатора, сервер доступа или межсетевого экрана (устройство NAT).
2. Устройство NAT получает пакет и делает запись в таблице отслеживания соединений, которая управляет преобразованием адресов.
3. Затем подменяет адрес источника пакета собственным внешним общедоступным IP-адресом и посылает пакет по месту назначения в Интернет.
4. Узел назначения получает пакет и передает ответ обратно устройству NAT.
5. Устройство NAT, в свою очередь, получив этот пакет, отыскивает отправителя исходного пакета в таблице отслеживания соединений, заменяет IP-адрес назначения на соответствующий частный IP-адрес и передает пакет на исходный компьютер. Поскольку устройство NAT посылает пакеты от имени всех внутренних компьютеров, оно изменяет исходный сетевой порт и данная информация хранится в таблице отслеживания соединений.

Три базовые концепции трансляции адресов:

- Статический NAT отображает локальные IP-адреса на конкретные публичные адреса на основании один к одному. Применяется, когда локальный хост должен быть доступен извне с использованием фиксированных адресов.
- Динамический NAT отображает набор частных адресов на некое множество публичных IP-адресов. Если число локальных хостов не превышает число имеющихся публичных адресов, каждому локальному адресу будет гарантироваться соответствие публичного адреса. В противном случае, число хостов, которые могут одновременно получить доступ во внешние сети, будет ограничено количеством публичных адресов.
- Маскарадный NAT (NAPT, NAT Overload, PAT, маскарадинг) – форма динамического NAT, который отображает несколько частных адресов в единственный публичный IP-адрес, используя различные порты. Известен также как PAT (Port Address Translation).

Четыре типа трансляции сетевых адресов:

- Full Cone (Полный конус). Любой внешний хост может отправлять пакеты локальному хосту, используя внешний IP-адрес локального хоста
- Restricted Cone (Ограниченный конус). Внешний хост может отправлять пакеты локальному хосту, только если до этого получил от него (локального хоста) хотя бы один пакет
- Port Restricted Cone (Порт ограниченного конуса). Аналогично предыдущему, только помимо IP-адреса еще учитывается порт
- Symmetric (Симметричный). Для каждого запроса составляется однозначное

отображение:	на	каждую	четверку
<внут. IP, внут. порт, IP назначения, порт назначения>		имеется	пара
<внеш. IP, внеш. порт>			

55. Опишите, как работает межсетевой экран.

Фильтрация трафика происходит на основе заранее установленных правил безопасности. Для этого создается специальная таблица, куда заносится описание допустимых и недопустимых к передаче данных. Межсетевой экран не пропускает трафик, если одно из запрещающих правил из таблицы срабатывает.

Файрволы могут запрещать или разрешать доступ, основываясь на разных параметрах: IP-адресах, доменных именах, протоколах и номерах портов, а также комбинировать их.

- IP-адреса. Каждое устройство, использующее протокол IP, обладает уникальным адресом. Вы можете задать определенный адрес или диапазон, чтобы пресечь попытки получения пакетов. Или наоборот — дать доступ только определенному кругу IP-адресов.
- Порты. Это точки, которые дают приложениям доступ к инфраструктуре сети. К примеру, протокол ftp пользуется портом 21, а порт 80 предназначен для приложений, используемых для просмотра сайтов. Таким образом, мы получаем возможность воспрепятствовать доступу к определенным приложениям и сервисам.
- Доменное имя. Адрес ресурса в интернете также является параметром для фильтрации. Можно запретить пропускать трафик с одного или нескольких сайтов. Пользователь будет огражден от неприемлемого контента, а сеть от пагубного воздействия.
- Протокол. Файрвол настраивается так, чтобы пропускать трафик одного протокола или блокировать доступ к одному из них. Тип протокола указывает на набор параметров защиты и задачу, которую выполняет используемое им приложение.

56. Технология персонального сетевого экранирования.

Для индивидуальных пользователей представляет интерес технология персонального сетевого экранирования. В этом случае сетевой экран устанавливается на защищаемый персональный компьютер.

Такой экран, называемый персональным экраном компьютера (Personal Firewall), или системой сетевого экранирования. Он контролирует весь исходящий и входящий трафик независимо от всех прочих системных защитных средств.

При экранировании отдельного компьютера поддерживается доступность сетевых сервисов, но уменьшается нагрузка, индуцированная внешней активностью. В результате снижается уязвимость внутренних сервисов защищаемого таким образом компьютера, поскольку сторонний злоумышленник должен сначала преодолеть экран, где защитные средства сконфигурированы особенно тщательно и жестко.

Для защиты рабочего места пользователя необходимо иметь, кроме

персонального МЭ, антивирусное ПО с актуальными сигнатурами, защиту доступа в корпоративную сеть через VPN.

В качестве примера персонального сетевого экрана можно указать межсетевой экран Брандмауэр Windows, который служит первой линией защиты персонального компьютера от различного рода вредоносных программ.

57. Распределенный межсетевой экран.

Распределенный межсетевой экран представляет собой централизованно управляемую совокупность сетевых мини-экранов, защищающих отдельные компьютеры сети.

При построении распределенных систем МЭ их функциональные компоненты распределяются по узлам сети и могут обладать различной функциональностью. При обнаружении подозрительных на атаку признаков управляющие модули распределенного МЭ могут адаптивно изменять конфигурацию, состав и расположение компонентов.

Главное отличие распределенного межсетевых экранов от персонального экрана заключается в наличии у распределенного межсетевых экранов функции централизованного управления. Если персональные сетевые экраны управляются только с того компьютера, на котором они установлены, и идеально подходят для домашнего применения, то распределенные межсетевые экраны могут управляться централизованно, с единой консоли управления, установленной в главном офисе организации.

58. Функции посредничества межсетевых экранов. Выполнение функций посредничества межсетевым экраном.

Функции посредничества МЭ выполняет с помощью специальных программ, называемых **программами-посредниками или экранирующими агентами**. Данные программы являются резидентными и запрещают непосредственную передачу пакетов сообщений между внешней и внутренней сетями.

При необходимости доступа из внутренней сети во внешнюю сеть или наоборот вначале должно быть установлено логическое соединение с программой-посредником, функционирующей на компьютере МЭ. Программа-посредник проверяет допустимость запрошенного межсетевого взаимодействия и при его разрешении сама устанавливает отдельное соединение с требуемым компьютером. Далее обмен информацией между компьютерами внутренней и внешней сети осуществляется через программного посредника, который может выполнять фильтрацию потока сообщений, а также осуществлять другие защитные функции.

Следует иметь в виду, что МЭ может выполнять функции фильтрации без применения программ-посредников, обеспечивая прозрачное взаимодействие между внутренней и внешней сетями. Вместе с тем программные посредники могут и не осуществлять фильтрацию потока сообщений.

В общем случае программы-посредники, блокируя прозрачную передачу потока сообщений, могут выполнять следующие функции:

- проверку подлинности передаваемых данных;
- фильтрацию и преобразование потока сообщений, например динамический

- поиск вирусов и прозрачное шифрование информации;
- разграничение доступа к ресурсам внутренней сети;
- разграничение доступа к ресурсам внешней сети;
- кэширование данных, запрашиваемых из внешней сети; идентификацию и аутентификацию пользователей;
- трансляцию внутренних сетевых адресов для исходящих пакетов сообщений;
- регистрацию событий, реагирование на задаваемые события, а также анализ зарегистрированной информации и генерацию отчетов

Программный посредник анализирует поступающие к нему пакеты данных, и если какой-либо объект не соответствует заданным критериям, то посредник либо блокирует его дальнейшее продвижение, либо выполняет соответствующие преобразования, например обезвреживание обнаруженных компьютерных вирусов. При анализе содержимого пакетов важно, чтобы экранирующий агент мог автоматически распаковывать проходящие файловые архивы.

59. Классификация межсетевых экранов по функционированию на уровнях модели OSI. Управляемые коммутаторы. Пакетные фильтры

Классификация по функционированию на уровнях модели OSI:

- управляемые коммутаторы
- пакетные фильтры (экранирующий маршрутизатор – Screening Router)
- шлюзы сеансового уровня (экранирующий транспорт);
- прикладные шлюзы, посредники прикладного уровня (Application Gateway);
- шлюзы экспертного уровня, инспекторы состояния (Stateful Inspection Firewall).

Управляемые коммутаторы иногда причисляют к классу межсетевых экранов, так как они осуществляют фильтрацию трафика между сетями или узлами сети. Однако они работают на канальном уровне и разделяют трафик в рамках локальной сети, а значит не могут быть использованы для обработки трафика из внешних сетей (например, из Интернета).

Многие производители сетевого оборудования, такие как Cisco, HP, ZyXEL, предоставляют в своих коммутаторах возможность фильтрации трафика на основе MAC-адресов, содержащихся в заголовках фреймов.

Однако данный метод фильтрации не является эффективным, так как аппаратно-установленный в сетевой карте MAC-адрес легко меняется программным путем, поскольку значение, указанное через драйвер, имеет более высокий приоритет, чем зашитое в плату. Поэтому многие современные коммутаторы позволяют использовать другие параметры в качестве признака фильтрации — например, VLAN ID.

Технология виртуальных локальных сетей (англ. Virtual Local Area Network)

позволяет создавать группы хостов, трафик которых полностью изолирован от других узлов сети. При реализации политики безопасности в рамках корпоративной сети, основу которых составляют управляемые коммутаторы, они могут быть мощным и достаточно дешёвым решением. **Взаимодействуя только с протоколами канального уровня, такие межсетевые экраны фильтруют трафик с очень высокой скоростью.**

Основным недостатком такого решения является невозможность анализа протоколов более высоких уровней.

Пакетные фильтры функционируют на сетевом уровне и контролируют прохождение трафика на основе информации, содержащейся в заголовке пакетов. Многие межсетевые экраны данного типа могут оперировать заголовками протоколов и более высокого, транспортного, уровня (например, TCP или UDP).

Пакетные фильтры одними из первых появились на рынке межсетевых экранов и по сей день остаются самым распространённым их типом. Данная технология реализована в подавляющем большинстве маршрутизаторов и даже в некоторых коммутаторах. При анализе заголовка сетевого пакета могут использоваться следующие параметры:

- IP-адреса источника и получателя;
- тип транспортного протокола;
- поля служебных заголовков протоколов сетевого и транспортного уровней;
- порт источника и получателя.

Пакетные фильтры могут быть реализованы в следующих компонентах сетевой инфраструктуры:

- пограничные маршрутизаторы
- операционные системы;
- персональные межсетевые экраны.

Так как пакетные фильтры обычно проверяют данные только в заголовках сетевого и транспортного уровней, они могут выполнять это достаточно быстро. Поэтому пакетные фильтры, встроенные в пограничные маршрутизаторы, идеальны для размещения на границе с сетью с низкой степенью доверия.

Однако в пакетных фильтрах отсутствует возможность анализа протоколов более высоких уровней сетевой модели OSI. Кроме того, пакетные фильтры обычно уязвимы для атак, которые используют подделку сетевого адреса

60. Классификация межсетевых экранов по функционированию на уровнях модели OSI. Шлюзы сеансового уровня. Прикладные шлюзы, посредники прикладного уровня (Application Gateway).

Межсетевые экраны прикладного уровня, как и шлюзы сеансового уровня, исключают прямое взаимодействие двух узлов. Однако, функционируя на прикладном уровне, они **способны «понимать» контекст передаваемого трафика**.

Межсетевые экраны, реализующие эту технологию, содержат несколько приложений-посредников (англ. application proxy), каждое из которых обслуживает свой прикладной протокол. Такой межсетевой экран способен выявлять в передаваемых сообщениях и блокировать несуществующие или нежелательные последовательности команд, что зачастую означает DoS-атаку, либо запрещать использование некоторых команд (например, FTP PUT, которая даёт возможность пользователю записывать информацию на FTP сервер).

Посредник прикладного уровня может определять тип передаваемой информации. Например, это позволяет заблокировать почтовое сообщение, содержащее исполняемый файл.

Посредники прикладного уровня способны выполнять аутентификацию пользователя, а также проверять, что SSL-сертификаты подписаны конкретным центром

Недостатками данного типа межсетевых экранов являются большие затраты времени и ресурсов на анализ каждого пакета. По этой причине они обычно не подходят для приложений реального времени. **Другим недостатком** является невозможность автоматического подключения поддержки новых сетевых приложений и протоколов, так как для каждого из них необходим свой агент.

61. Классификация межсетевых экранов по функционированию на уровнях модели OSI. Шлюзы экспертного уровня, инспекторы состояния (Stateful Inspection Firewall).

Каждый из вышеперечисленных типов межсетевых экранов используется для защиты корпоративных сетей и обладает рядом преимуществ. Однако, куда эффективней было бы собрать все эти преимущества в одном устройстве и **получить межсетевой экран, осуществляющий фильтрацию трафика с сетевого по прикладной уровень**. Данная идея была реализована в инспекторах состояний, совмещающих в себе высокую производительность и защищённость. Данный класс межсетевых экранов позволяет контролировать:

- каждый передаваемый пакет — на основе таблицы правил;
- каждую сессию — на основе таблицы состояний;
- каждое приложение — на основе разработанных посредников.

Осуществляя фильтрацию трафика по принципу шлюза сеансового уровня, данный класс межсетевых экранов не вмешивается в процесс установления соединения между узлами. **Поэтому производительность инспектора состояний заметно выше**,

чем у посредника прикладного уровня и шлюза сеансового уровня, и сравнима с производительностью пакетных фильтров.

Ещё одно достоинство инспекторов состояния — прозрачность для пользователя: для клиентского программного обеспечения не потребуется дополнительная настройка.

Термин инспектор состояния (англ. stateful inspection), внедрённый компанией Check Point Software, полюбился производителям сетевого оборудования настолько, что сейчас практически каждый межсетевой экран причисляют к этой технологии, даже если он и не реализует её полностью.

По используемой технологии:

- контроль состояния протокола (Stateful Inspection)
- на основе модулей посредников (прокси).

По исполнению:

- аппаратно-программный;
- программный.

По схеме подключения:

- схема единой защиты сети;
- схема с защищаемым закрытым и незащищаемым открытым сегментами сети;
- схема с отдельной защитой закрытого и открытого сегментов сети.