**Лекция 1-2**

**Информационная безопасность: что ей угрожает и как с этим бороться**

**Аннотация:**В лекции рассмотрены основные понятия информационной безопасности. Ознакомление с ФЗ " Об информации, информационных технологиях и о защите информации".

Рассказываем, что на самом деле охватывает понятие «информационная безопасность», и разбираемся в видах информационных угроз и современных технологиях защиты данных.

**Что такое информационная безопасность**

Корпоративные системы и личная информация пользователей уязвимы для различных угроз: данные могут быть изменены или удалены в результате технических сбоев или кибератак.

Информационная безопасность (на англ. InfoSec) — это состояние систем, при котором элементы её инфраструктуры, например, оборудование, каналы передачи данных и хранилища данных, устойчивы к внешним и внутренним угрозам.

Например, данные с информацией о клиентах и заказах или финансовая отчётность очень важны для компании. Если они попадут в руки конкурентов или информация будет испорчена, искажена, удалена или уничтожена, это приведёт к негативным последствиям. Потеря клиентской базы повлияет на прибыль: информацию не получится использовать для повторных продаж и маркетингового продвижения. Финансовую отчётность могут запросить государственные органы для проверки, а её отсутствие приведет к штрафам или другим санкциям.

Личная информация пользователей — номера банковских карт, контакты или данные о записях к врачу тоже находится под угрозой. Системы хранения данных сайтов, где она содержится, или личные аккаунты пользователей могут взломать. Данными могут воспользоваться киберпреступники, чтобы оформить кредит от имени пользователя или продать его контакты и личную информацию третьим лицам.

Также информационная безопасность — это область знаний и профессиональное направление. Специалисты по информационной безопасности, или ИБ-специалисты, изучают уязвимости в системах и ищут способы повышения их устойчивости к угрозам.

Если говорить о пользователях, то поддержание информационной безопасности можно отнести к полезному навыку, который помогает не стать жертвами кибератак.

Однако стоит разделять понятия кибербезопасности и информационной безопасности. Кибербезопасность предполагает защиту цифровой информации от кражи в интернет-пространстве. Информационная безопасность включает методы защиты информации, которая может храниться на различных носителях — в облачном хранилище, на серверах и на обычной печатной бумаге.

Чтобы успешно бороться с угрозами информационной безопасности, ИБ-специалисты должны не только изучить виды угроз и методы защиты от них, но и освоить некоторые навыки разработчиков и аналитиков данных. На курсах Практикума можно освоить разные языки программирования и узнать, как устроена работа с данными.

**Принципы защиты информации**

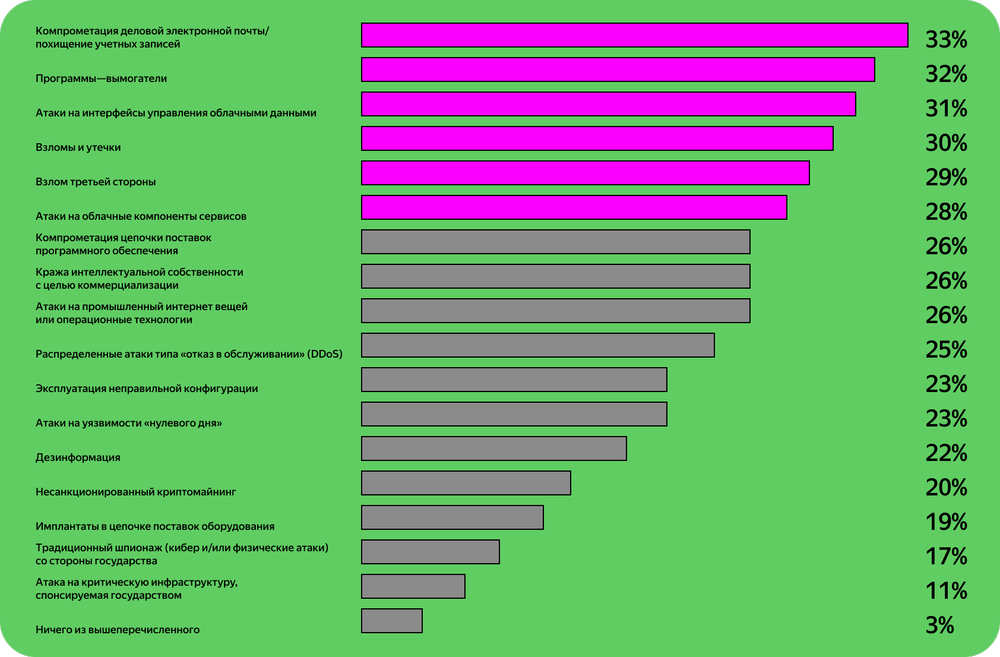
Защита информации строится на трёх принципах:  
  
**1. Конфиденциальность**  
Информация должна быть защищена от несанкционированного доступа. Например, подключение к корпоративным системам с личного компьютера ставит под угрозу конфиденциальность корпоративной информации. Или рассказ о внутренних процессах компании на собеседовании в другую организацию. Поэтому во многих компаниях разрешено пользоваться внутренним ПО только с рабочего компьютера, а сотрудники часто подписывают соглашение о неразглашении (на англ. non-disclosure agreement, или NDA).  
  
**2. Целостность**  
Информация не должна меняться без разрешения её собственника. Например, менять пароль от аккаунта может только его владелец. А чтобы защитить аккаунт от взлома, нужно использовать двухфакторную аутентификацию — дополнительное подтверждение действий по телефону или sms.  
  
Для защиты целостности корпоративной информации каждый пользователь должен иметь доступ только к тем данным, которые нужны для его работы. Например, администрировать базу клиентов компании может только аналитик или инженер данных. А сотрудники технической поддержки должны иметь доступ только к просмотру контактов и истории заказов.  
  
**3. Доступность**  
Сотрудники с санкционированным доступом к информации могут обращаться к ней для решения рабочих задач. При этом должны быть исключены ситуации, когда доступ внезапно пропадает. Например, если из-за технического сбоя или кибератаки на систему менеджеры по продажам не смогут войти в [CRM-систему](https://practicum.yandex.ru/blog/chto-takoe-crm-sistema-i-kak-ih-vybirat/) компании.



Модель, основанную на трёх принципах защиты информации, называют триадой CIA и часто визуализируют в виде треугольника

**Цели и виды угроз**

Цель угрозы информационной безопасности — получить доступ к персональным и корпоративным данным. Их кража или повреждение может негативно отразиться на жизни людей и деятельности компаний.   
  
Например, если преступники получат доступ к информации о составе семьи пользователя, они смогут воспользоваться ей для мошенничества: от имени родственника обманным путем вынудить перевести деньги на счёт злоумышленника.   
  
Важную корпоративную информацию тоже могут украсть и использовать в своих целях, например продать конкурентам данные о партнёрах.   
  
Случайное или намеренное повреждение или уничтожение данных может остановить работу компании и создать неудобства для пользователей. Например, если из-за технического сбоя на маркетплейсе пропадут данные о новых заказах, сотрудники склада не смогут их собрать и отправить клиентам.  
  
Можно выделить три основных вида угроз информационной безопасности:  
  
**1. Несанкционированный доступ**  
Доступ к компьютерным системам, сетям или данным могут получить киберпреступники или пользователи без подтвержденного доступа.   
  
Например, сотрудник из-за ошибки в корпоративной системе случайно авторизуется в программе, которой пользуется кадровая служба. Он получит доступ к личным делам руководителей и коллег. Или преступники взломают электронную почту пользователя и получат доступ к его аккаунтам в различных сервисах.   
  
**2. Нарушение целостности**  
Целостность данных подразумевает, что информация остаётся неизменной и точной. Угроза нарушения целостности — это возможность случайного или намеренного изменения или удаления данных без разрешения их создателя или владельца.   
  
Изменить или удалить данные могут сотрудники компании по неосторожности, вредоносное программное обеспечение или хакеры. Это может случиться и в результате технического сбоя. Злоумышленники, получив доступ к администрированию сайта, могут изменить цены на товары или услуги. Менеджеры по продажам не смогут подтвердить заказы по неправильным ценам, клиенты потеряют доверие к компании и перестанут пользоваться её услугами. А это финансовые потери.  
  
**3. Раскрытие информации**  
Этот тип угрозы информационной безопасности подразумевает утечку и распространение чувствительной информации, которая должна оставаться конфиденциальной. Например, раскрытие данных о партнерах приведёт к потере репутации компании, деталей разработки нового продукта — к копированию его конкурентами, а сценария нового фильма киностудии — к потере части зрителей и прибыли от проката.



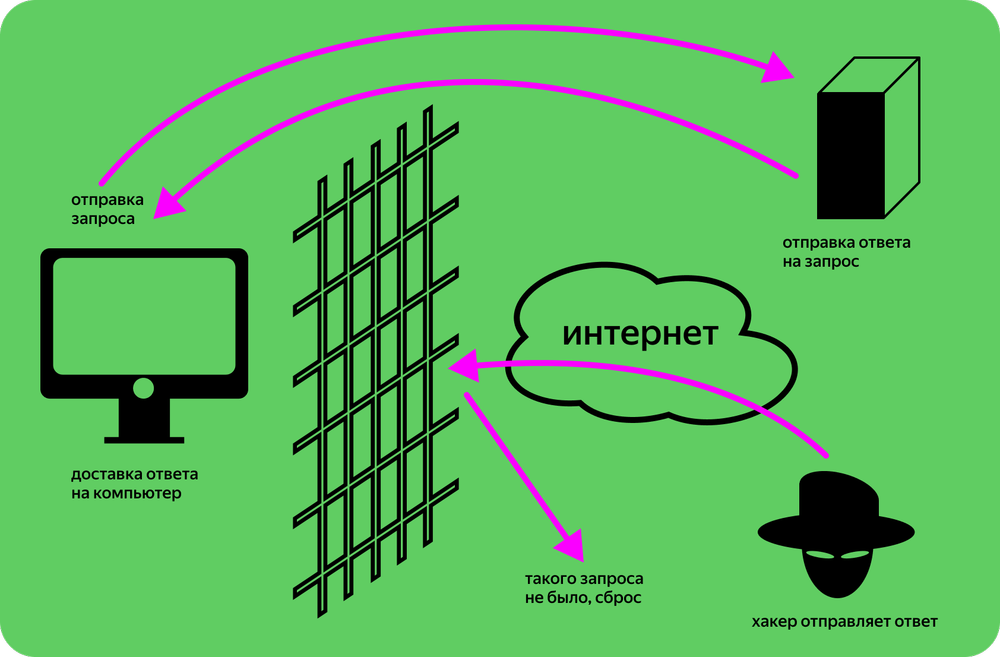
[По прогнозам Statista.com](https://www.statista.com/statistics/1300557/threat-outlook-by-reportable-incidents/), в 2023 году глобальные компании будут подвергаться всем трём видам угроз информационной безопасности. В топе: компрометация корпоративной электронной почты, программы-вымогатели, которые блокируют доступ к информации, и атаки на интерфейсы управления облачными данными

**Средства защиты информации**

Для защиты информации компании и пользователи используют различные инструменты и средства. Разделим их на основные виды:  
  
● **Технические средства**  
Это оборудование и методы защиты информации, которые на физическом уровне ограничивают доступ к ней. Например, специальные устройства, которые блокируют интернет-сигнал в переговорной, замок и сигнализация, которые перекрывают доступ в серверную или архив бумажных документов. Или пароль на телефоне, благодаря которому при краже устройства злоумышленник не получит доступ к данным в нём.  
  
● **Программные средства**  
ПО, которое может обнаружить и предотвратить угрозы безопасности цифровых данных. К нему относятся, в том числе антивирусные программы или программы для обнаружения и предотвращения вторжений. Такие программы выявляют и блокируют аномальную активность в трафике, например, авторизации в корпоративной системе из другой страны.   
  
К программным способам защиты информации также относят технологии для шифрования. Они переводят данные в набор символов, который нельзя расшифровать без ключей. Это защищает информацию от раскрытия в случае утечки.   
  
● **Организационные средства**  
Меры, которые принимает руководство компании. Например, разработка политики корпоративной безопасности и контроль за её соблюдением, обучение сотрудников и подписание NDA при приёме на работу.   
  
Если говорить о защите личной информации, к организационным мерам можно отнести правила информационной гигиены в интернете и жизни. Например, не переходить по сомнительным ссылкам, не оставлять гаджеты в общественных местах без присмотра, [использовать VPN](https://practicum.yandex.ru/blog/chto-takoe-vpn-i-zachem-on-nuzhen/) при подключении к общедоступной сети Wi-Fi в кафе.

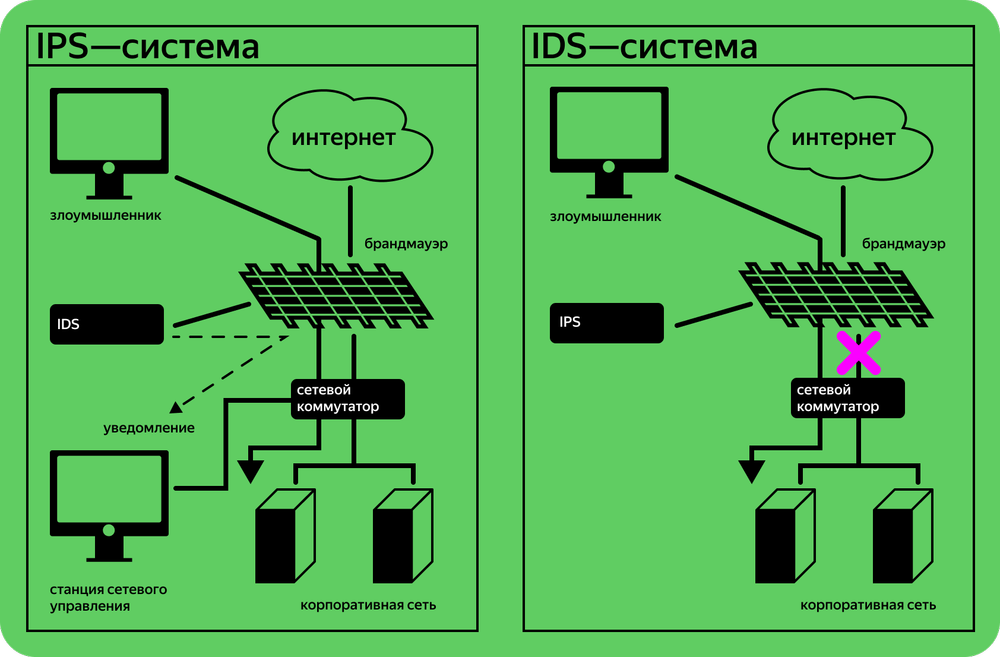
**Технологии защиты информации**

Перечислим некоторые современные технологии для защиты информации, которые относятся к программным средствам:  
  
**Криптография** — технология преобразования данных, с помощью которой они становятся зашифрованными с помощью специальных ключей или методов. Криптографические методы используют, например, государственные учреждения для создания цифровых подписей, банки — для денежных переводов, пользователи — когда заходят в интернет с подключенным VPN.  
  
**Блокчейн** — технология децентрализованного хранения данных. Данные разделяются на блоки (на англ. block), каждый из блоков связан с предыдущим, тем самым выстраивая цепочку (на англ. chain). Изменения данных в предыдущих блоках является ресурсоёмким процессом и в большинстве случаев невозможным. Поэтому всё, что попадает в сеть блокчейна, остается в неизменном состоянии навсегда. Этот способ используют, например, в здравоохранении — организации хранят в [блокчейне](https://practicum.yandex.ru/blog/chto-takoe-blokchain-i-kak-eto-rabotaet/) медицинские карты пациентов.  
  
**Брандмауэр** — технология, которая предоставляет защитный экран между устройством и внешними сетями. С помощью брандмауэра можно, например, распределить трафик между устройствами и ограничить доступ к определённым ресурсам. Брандмауэры устанавливают, например, в школах, чтобы оградить детей от запрещённого или опасного контента. Или в организациях, чтобы заблокировать спам, отправляемый потенциальными злоумышленниками на электронные почты сотрудников.



Брандмауэры фильтруют поступающий на устройство трафик

**IDS-системы** (сокр. от Intrusion Detection System) — технология для обнаружения вторжений. IDS отслеживает сетевой трафик или трафик внутри корпоративной системы и выявляет необычную активность, которая указывает на возможное нарушение безопасности. Например, попытки взлома сети или атаки на серверы. IDS-систему можно установить на уровне сети или на уровне отдельного устройства. В первом случае система будет анализировать весь трафик, во втором — только тот, что проходит через устройство.  
  
**IPS-системы** (сокр. от Intrusion Prevention System) — технология для предотвращения вторжений. В отличие от IDS, не только фиксирует потенциальные угрозы безопасности, но и принимает активные меры для защиты информации. Например, автоматически блокирует IP-адреса, с которых пытаются взломать систему. При этом IPS обнаруживает не только внешние атаки, но и внутренние — когда атака идёт с рабочего компьютера кого-то из сотрудников. Ещё IPS-система может сканировать скачиваемые файлы и не допускать установки вирусов на компьютеры пользователей.



IDS-системы уведомляют об угрозе информационной безопасности сотрудника, который должен её обеспечивать. IPS — блокирует угрозу. Для обеих систем нужны большие мощности, поэтому их лучше устанавливать после брандмауэра. Он будет фильтровать весь трафик, а IDS и IPS — то, что он пропустил

**DLP-системы** (сокр. от Data Loss Prevention) — технология, которая предотвращает утечку информации. Например, блокирует отправку конфиденциальных данных по электронной почте или через мессенджеры. С помощью системы можно также запретить распечатку документов с определённого устройства. Эту функцию можно активировать в случае увольнения сотрудника, чтобы он не смог забрать с собой корпоративную информацию ни в цифровом, ни в печатном виде.  
  
**EDR-системы** (сокр. от Endpoint Detection and Response) — технология для обнаружения вредоносной активности на конечных узлах сети, например компьютерах или смартфонах. EDR отслеживает подозрительные действия пользователей или попытки взлома устройств и отправляет уведомления о них ИБ-специалисту. По сути, системы типа EDR — это более современные виды антивирусных программ. Они в режиме реального времени выявляют сложные угрозы, например вредоносное ПО для корпоративного шпионажа. EDR-система анализирует активность устройства и выявляет отклонения от обычного паттерна.   
  
**UBA-аналитика** (сокр. от User Behavior Analytics) — технология, которая анализирует поведение пользователей в информационных системах и сетях, чтобы отслеживать подозрительную активность. Например, UBA может выявить несанкционированный доступ к учётной записи пользователя, проанализировав отклонения в его поведении. Это может быть вход в систему из другой страны или просмотр файлов, которые обычно не нужны пользователю для работы. Обнаружив отклонения, система UBA может заблокировать скомпрометированную учётную запись.

**1.1. Основные понятия информационной безопасности**

Прежде чем говорить об обеспечении безопасности персональных данных, необходимо определить, что же такое *информационная безопасность*. Термин "*информационная безопасность*" может иметь различный смысл и трактовку в зависимости от контекста. В данном курсе под **информационной безопасностью** мы будем понимать защищенность информации и поддерживающей инфраструктуры от случайных или преднамеренных воздействий естественного или искусственного характера, которые могут нанести *неприемлемый ущерб* *субъектам информационных отношений*, в том числе владельцам и пользователям информации и поддерживающей инфраструктуры [[1](https://intuit.ru/studies/courses/697/553/literature#literature.1)].

ГОСТ "*Защита информации*. Основные термины и определения" вводит понятие **информационной безопасности** как состояние защищенности информации, при котором обеспечены ее *конфиденциальность*, доступность и *целостность*.

* **Конфиденциальность** – состояние информации, при котором доступ к ней осуществляют только субъекты, имеющие на него право.
* **Целостность** – состояние информации, при котором отсутствует любое ее изменение либо изменение осуществляется только преднамеренно субъектами, имеющими на него право;
* **Доступность** – состояние информации, при котором субъекты, имеющие право доступа, могут реализовывать его беспрепятственно.

**Угрозы информационной безопасности** – совокупность условий и факторов, создающих потенциальную или реально существующую опасность нарушения безопасности информации [[2](https://intuit.ru/studies/courses/697/553/literature#literature.2),[3](https://intuit.ru/studies/courses/697/553/literature#literature.3)]. **Атакой**называется попытка реализации угрозы, а тот, кто предпринимает такую попытку, - **злоумышленником**. Потенциальные злоумышленники называются *источниками угрозы*.

*Угроза* является следствием наличия **уязвимых мест или уязвимостей** в информационной системе. Уязвимости могут возникать по разным причинам, например, в результате непреднамеренных ошибок программистов при написании программ.

Угрозы можно классифицировать по нескольким критериям:

* по *свойствам информации* (доступность, целостность, конфиденциальность), против которых угрозы направлены в первую очередь;
* по компонентам информационных систем, на которые угрозы нацелены (данные, программы, аппаратура, *поддерживающая инфраструктура*);
* по способу осуществления (случайные/преднамеренные, действия природного/техногенного характера);
* по расположению источника угроз (внутри/вне рассматриваемой ИС).

Обеспечение информационной безопасности является сложной задачей, для решения которой требуется *комплексный подход*. Выделяют следующие уровни защиты информации:

1. законодательный – законы, нормативные акты и прочие документы Казахстана и международного сообщества;
2. административный – комплекс мер, предпринимаемых локально руководством организации;
3. процедурный уровень – меры безопасности, реализуемые людьми;
4. *программно-технический уровень* – непосредственно средства защиты информации.

Законодательный уровень является основой для построения системы защиты информации, так как дает базовые понятия *предметной области* и определяет меру наказания для потенциальных злоумышленников. Этот уровень играет координирующую и направляющую роли и помогает поддерживать в обществе негативное (и карательное) *отношение* к людям, нарушающим информационную *безопасность*.

**1.2. ФЗ "Об информации, информационных технологиях и о защите информации"**

В законодательстве базовым законом в области защиты информации является ФЗ "Об информации, информационных технологиях и о защите информации". Поэтому основные понятия и решения, закрепленные в законе, требуют пристального рассмотрения.

Закон регулирует отношения, возникающие при:

* осуществлении права на поиск, получение, передачу, производство и распространение информации;
* применении информационных технологий;
* обеспечении защиты информации.

Закон дает основные определения в области защиты информации. Приведем некоторые из них:

* **информация**- сведения (сообщения, данные) независимо от формы их представления;
* **информационные технологии** - процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов;
* **информационная система** - совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств;
* **обладатель информации** - лицо, самостоятельно создавшее информацию либо получившее на основании закона или договора право разрешать или ограничивать доступ к информации, определяемой по каким-либо признакам;
* **оператор информационной системы** - гражданин или юридическое лицо, осуществляющие деятельность по эксплуатации информационной системы, в том числе по обработке информации, содержащейся в ее базах данных.
* **конфиденциальность информации** - обязательное для выполнения лицом, получившим доступ к определенной информации, требование не передавать такую информацию третьим лицам без согласия ее обладателя [[4](https://intuit.ru/studies/courses/697/553/literature#literature.4)].

В законе сформулированы принципы правового регулирования отношений в сфере информации, информационных технологий и защиты информации:

1. свобода поиска, получения, передачи, производства и распространения информации любым законным способом;
2. установление ограничений доступа к информации только федеральными законами;
3. открытость информации о деятельности государственных органов и органов местного самоуправления и свободный доступ к такой информации, кроме случаев, установленных федеральными законами;
4. равноправие языков народов Казахстана при создании информационных систем и их эксплуатации;
5. обеспечение безопасности Казахстана при создании информационных систем, их эксплуатации и защите содержащейся в них информации;
6. достоверность информации и своевременность ее предоставления;
7. неприкосновенность частной жизни, недопустимость сбора, хранения, использования и распространения информации о частной жизни лица без его согласия;
8. недопустимость установления нормативными правовыми актами каких-либо преимуществ применения одних информационных технологий перед другими, если только обязательность применения определенных информационных технологий для создания и эксплуатации государственных информационных систем не установлена федеральными законами.

Вся *информация* делится на **общедоступную** и ограниченного **доступа**. К общедоступной информации относятся общеизвестные сведения и иная *информация*, *доступ* к которой не ограничен. В законе, определяется *информация*, к которой нельзя ограничить *доступ*, например, *информация* об окружающей среде или деятельности государственных органов. Оговаривается также, что *ограничение доступа* к информации устанавливается федеральными законами в целях защиты основ конституционного строя, нравственности, здоровья, прав и законных интересов других лиц, обеспечения обороны страны и безопасности государства. Обязательным является соблюдение конфиденциальности информации, *доступ* к которой ограничен федеральными законами.

Запрещается требовать от гражданина (физического лица) предоставления информации о его частной жизни, в том числе информации, составляющей личную или семейную тайну, и получать такую информацию помимо воли гражданина (физического лица), если иное не предусмотрено федеральными законами.

Закон выделяет 4 категории информации в зависимости от порядка ее предоставления или распространения:

1. информацию, свободно распространяемую;
2. информацию, предоставляемую по соглашению лиц, участвующих в соответствующих отношениях;
3. информацию, которая в соответствии с федеральными законами подлежит предоставлению или распространению;
4. информацию, распространение которой в Казахстане ограничивается или запрещается.

Закон устанавливает равнозначность электронного сообщения, подписанного электронной цифровой подписью или иным аналогом собственноручной подписи, и документа, подписанного собственноручно.

Дается следующее *определение* защите информации - представляет собой принятие правовых, организационных и технических мер, направленных на:

1. обеспечение защиты информации от неправомерного доступа, уничтожения, модифицирования, блокирования, копирования, предоставления, распространения, а также от иных неправомерных действий в отношении такой информации;
2. соблюдение конфиденциальности информации ограниченного доступа;
3. реализацию права на доступ к информации.

Обладатель информации, оператор информационной системы в случаях, установленных законодательством Казахстана, обязаны обеспечить:

1. предотвращение несанкционированного доступа к информации и (или) передачи ее лицам, не имеющим права на доступ к информации;
2. своевременное обнаружение фактов несанкционированного доступа к информации;
3. предупреждение возможности неблагоприятных последствий нарушения порядка доступа к информации;
4. недопущение воздействия на технические средства обработки информации, в результате которого нарушается их функционирование;
5. возможность незамедлительного восстановления информации, модифицированной или уничтоженной вследствие несанкционированного доступа к ней;
6. постоянный контроль за обеспечением уровня защищенности информации.

Таким образом, Закон "Об информации, информационных технологиях и о защите информации" создает правовую основу информационного обмена и определяет *права* и обязанности его субъектов.

**Лекция 4 Автоматизированная и неавтоматизированная обработка персональных данных**

**Аннотация:**Цель лекции: познакомиться с автоматизированными и неавтоматизированными системами персональных данных. Рассмотреть основные принципы построения системы защиты персональных данных.

**Аннотация:**В лекции ведется изучение понятий и классификаций уязвимостей и угроз, рассмотрение наиболее распространенных атак. Состав и содержание организационно-распорядительной документации по защите ПД.

**4.1. Угрозы информационной безопасности**

При построении системы *защиты персональных данных* (далее СЗПД) ключевым этапом является построение частной модели угроз для конкретной организации. На основании этой модели в дальнейшем подбираются адекватные и достаточные средства защиты, в соответствии с принципами, рассмотренными в ["Автоматизированная и неавтоматизированная обработка персональных данных"](https://intuit.ru/studies/courses/697/553/lecture/12445).

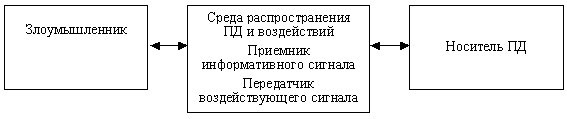
Под **угрозами безопасности ПД** при их обработке в ИСПД понимается совокупность условий и факторов, создающих опасность несанкционированного, в том числе *случайного, доступа* к персональным данным, результатом которого может стать уничтожение, изменение, блокирование, *копирование*, распространение персональных данных, а также иных несанкционированных действий при их обработке в информационной системе персональных данных.

Появление угроз безопасности может быть связано как с преднамеренными действиями злоумышленников, так и с непреднамеренными действиями персонала или пользователей ИСПД.

*Угрозы безопасности* могут быть реализованы двумя путями:

* через технические каналы утечки;
* путем несанкционированного доступа.

Обобщенная схема реализации канала угроз ПД показана на [рисунке 4.1](https://intuit.ru/studies/courses/697/553/lecture/12447?page=1#image.4.1)



**Рис. 4.1.**Обобщенная схема канала реализации угроз безопасности персональных данных

**Технический канал утечки информации** – совокупность носителя информации (средства обработки), физической среды распространения информативного сигнала и средств, которыми добывается защищаемая *информация*. Среда распространения бывает однородной, например, только воздух при распространении электромагнитного излучения, или неоднородной, когда сигнал переходит из одной среды в другую. Носителями ПД могут быть люди, работающие с ИСПД, технические средства, вспомогательные средства и т.д. Главное, что *информация* при этом отображается в виде полей, сигналов, образов, количественных характеристиках физических величин.

Как правило, выделяют следующие угрозы за счет реализации технических каналов утечки:

* угрозы утечки речевой информации. Фактически злоумышленник перехватывает информацию с помощью специальной аппаратуры в виде акустических, виброакустических волн, а также электромагнитного излучения, модулированного акустическим сигналом. В качестве средств могут использоваться различного рода электронные устройства, подключаемые либо к каналам связи, либо к техническим средствам обработки ПД.
* угрозы утечки видовой информации. В этом случае речь идет о непосредственном просмотре ПД при наличии прямой видимости между средством наблюдения и носителем ПД. В качестве средств наблюдения используются оптические средства и видеозакладки;
* угрозы утечки информации по каналам побочных электромагнитных излучений и наводок (ПЭМИН). Речь идет о перехвате побочных (не связанных с прямым функциональным значением элементов ИСПД) информативных электромагнитных полей и электрических сигналов, возникающих при обработке ПД техническими средствами ИСПД. Для регистрации ПЭМИН используется аппаратура в составе радиоприемных устройств и оконечных устройств восстановления информации. Кроме этого, перехват ПЭМИН возможен с использованием электронных устройств перехвата информации, подключенных к каналам связи или техническим средствам обработки ПД. Наводки электромагнитных излучений возникают при излучении элементами технических средств ИСПД информативных сигналов при наличии емкостной, индуктивной или гальванической связей соединительных линий технических средств ИСПД и различных вспомогательных устройств.

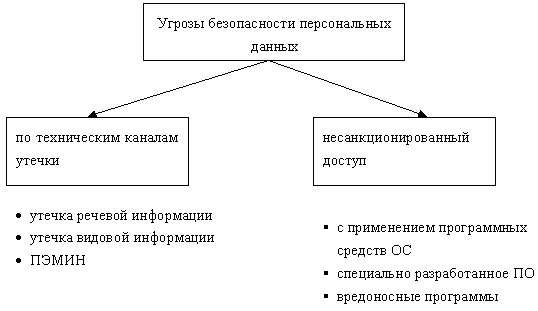
*Источниками угроз*, реализуемых за счет **несанкционированного доступа к базам данных** с использованием штатного или специально разработанного программного обеспечения, являются субъекты, действия которых нарушают регламентируемые в ИСПД *правила разграничения доступа* к информации. Этими субъектами могут быть:

* нарушитель;
* носитель вредоносной программы;
* аппаратная закладка.

Под **нарушителем** здесь и далее понимается физическое лицо (лица), случайно или преднамеренно совершающее действия, следствием которых является нарушение безопасности ПД при их обработке техническими средствами в информационных системах. С точки зрения наличия *права* легального доступа в помещения, в которых размещены *аппаратные средства*, обеспечивающие *доступ* к ресурсам ИСПД, нарушители подразделяются на два типа:

* нарушители, не имеющие доступа к ИСПД, реализующие угрозы из внешних сетей связи общего пользования и (или) сетей международного информационного обмена, – внешние нарушители;
* нарушители, имеющие доступ к ИСПД, включая пользователей ИСПД, реализующие угрозы непосредственно в ИСПД, – внутренние нарушители.

Для ИСПД, предоставляющих информационные услуги удаленным пользователям, внешними нарушителями могут являться лица, имеющие возможность осуществлять несанкционированный *доступ* к информации с использованием специальных программных воздействий, алгоритмических или программных закладок через автоматизированные рабочие места, *терминальные устройства* ИСПД, подключенные к сетям общего пользования.

Обобщим полученные знания с помощью [рисунка 4.2](https://intuit.ru/studies/courses/697/553/lecture/12447?page=1#image.4.2).

**Рис. 4.2.**Классификация угроз безопасности персональных данных по способу реализации

Угрозы можно классифицировать *по* различным признакам, например, *по* виду нарушаемого *свойства информации* (*конфиденциальность*, *целостность*, доступность), *по* типу ИСПД, на которые направлена *атака*, *по* типу используемой для атаки уязвимости.

**4.2. Общая характеристика уязвимостей информационной системы персональных данных**

Появление потенциальных угроз безопасности связано с наличием слабых мест в ИСПД - уязвимостей. **Уязвимость информационной системы персональных данных** – недостаток или слабое *место* в системном или прикладном программном (программно-аппаратном) обеспечении ИСПД, которые могут быть использованы для реализации *угрозы безопасности* персональных данных.

Причинами возникновения уязвимостей в общем случае являются:

1. ошибки при разработке программного обеспечения;
2. преднамеренные изменения программного обеспечения с целью внесения уязвимостей;
3. неправильные настройки программного обеспечения;
4. несанкционированное внедрение вредоносных программ;
5. неумышленные действия пользователей;
6. сбои в работе программного и аппаратного обеспечения.

Уязвимости, как и угрозы, можно классифицировать *по* различным признакам:

1. по типу ПО – системное или прикладное.
2. по этапу жизненного цикла ПО, на котором возникла уязвимость – проектирование, эксплуатация и пр.
3. по причине возникновения уязвимости, например, недостатки механизмов аутентификации сетевых протоколов.
4. по характеру последствий от реализации атак – изменение прав доступа, подбор пароля, вывод из строя системы в целом и пр.

Наиболее часто используемые уязвимости относятся к протоколам сетевого взаимодействия и к операционным системам, в том числе к прикладному программному обеспечению.

Уязвимости операционной системы и прикладного *ПО* в частном случае могут представлять:

* функции, процедуры, изменение параметров которых определенным образом позволяет использовать их для несанкционированного доступа без обнаружения таких изменений операционной системой;
* фрагменты кода программ ("дыры", "люки"), введенные разработчиком, позволяющие обходить процедуры идентификации, аутентификации, проверки целостности и др.;
* отсутствие необходимых средств защиты (аутентификации, проверки целостности, проверки форматов сообщений, блокирования несанкционированно модифицированных функций и т.п.);
* ошибки в программах (в объявлении переменных, функций и процедур, в кодах программ), которые при определенных условиях (например, при выполнении логических переходов) приводят к сбоям, в том числе к сбоям функционирования средств и систем защиты информации.

Уязвимости протоколов сетевого взаимодействия связаны с особенностями их программной реализации и обусловлены ограничениями на размеры применяемого буфера, недостатками процедуры аутентификации, отсутствием проверок правильности служебной информации и др. Так, например, протокол прикладного уровня *FTP*, широко используемый в Интернете, производит аутентификацию на базе открытого текста, тем самым позволяя перехватывать данные учетной записи.

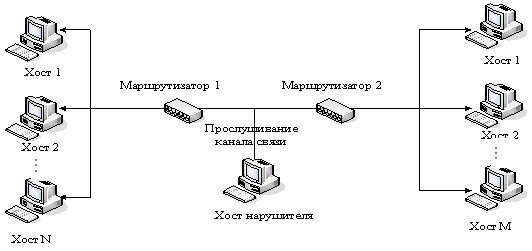
Прежде чем приступать к построению системы защиты информации необходимо провести *анализ уязвимостей* ИСПД и попытаться сократить их количество, то есть использовать метод превентивности. Можно закрыть лишние порты, поставить "заплатки" на *программное обеспечение* (например, *service* pack для *Windows*), ввести более сильные методы аутентификации и т.п. Эти меры могут существенно сократить материальные, временные и трудовые *затраты* на построение системы *защиты персональных данных* в дальнейшем.

**4.3. Наиболее часто реализуемые угрозы**

В связи с повсеместным развитием Интернета наиболее часто атаки производятся с использованием уязвимостей протоколов сетевого взаимодействия. Рассмотрим 7 наиболее распространенных атак.

1. **Анализ сетевого трафика**

Данный вид атаки направлен в первую очередь на получение пароля и идентификатора пользователя путем "прослушивания сети". Реализуется это с помощью *sniffer* – специальная программа-анализатор, которая перехватывает все пакеты, идущие по сети. И если протокол, например, FTP или TELNET, передает аутентификационную информацию в открытом виде, то злоумышленник легко получает доступ к учетной записи пользователя.



**Рис. 4.3.**Схема реализации угрозы “Анализ сетевого трафика”

1. **Сканирование сети**

Суть данной атаки состоит в сборе информации о топологии сети, об открытых портах, используемых протоколах и т.п. Как правило, реализация данной угрозы предшествует дальнейшим действиям злоумышленника с использованием полученных в результате сканирования данных.

1. **Угроза выявления пароля**

*Целью атаки* является преодоление парольной защиты и получении НСД к чужой информации. Методов для кражи пароля очень много: простой перебор всех возможных значений пароля, перебор с помощью специальных программ (*атака словаря*), перехват пароля с помощью программы-анализатора сетевого трафика.

1. **Подмена доверенного объекта сети и передача по каналам связи сообщений от его имени с присвоением его прав доступа.** Доверенный объект – это элемент сети, легально подключенный к серверу.

Такая угроза эффективно реализуется в системах, где применяются нестойкие алгоритмы идентификации и аутентификации хостов, пользователей и т.д.

Могут быть выделены две разновидности процесса реализации указанной угрозы: с установлением и без установления виртуального соединения.

Процесс реализации с установлением виртуального соединения состоит в присвоении прав *доверенного субъекта* взаимодействия, что позволяет нарушителю вести сеанс работы с объектом сети от имени *доверенного субъекта*. Реализация угрозы данного типа требует преодоления системы идентификации и аутентификации сообщений (например, атака rsh-службы UNIX-хоста).

Процесс реализации угрозы без установления виртуального соединения может иметь место в сетях, осуществляющих идентификацию передаваемых сообщений только по сетевому адресу отправителя. Сущность заключается в передаче служебных сообщений от имени сетевых управляющих устройств (например, от имени маршрутизаторов) об изменении маршрутно-адресных данных.

В результате реализации угрозы нарушитель получает права доступа, установленные его пользователем для доверенного абонента, к техническому средству ИСПД– цели угроз.

1. **Навязывание ложного маршрута сети**

Данная атака стала возможной из-за недостатков протоколов маршрутизации (RIP, OSPF, *LSP*) и управления сетью (ICMP, SNMP), таких как слабая аутентификация маршрутизаторов. Суть атаки состоит в том, что злоумышленник, используя уязвимости протоколов, вносит несанкционированные изменения в маршрутно-адресные таблицы.

1. **Внедрение ложного объекта сети**

Когда изначально объекты сети не знают информацию друг о друге, то для построения адресных таблиц и последующего взаимодействия, используется *механизм запрос* (как правило, широковещательный) - ответ с искомой информацией. При этом если нарушитель перехватил такой запрос, то он может выдать ложный ответ, изменить таблицу маршрутизации всей сети, и выдать себя за легального субъекта сети. В дальнейшем все пакеты, направленные к легальному субъекту, будут проходить через злоумышленника.

1. **Отказ в обслуживании**

Этот тип атак является одним из самых распространенных в настоящее время. Целью такой атаки является отказ в обслуживании, то есть нарушение доступности информации для законных субъектов информационного обмена.

Могут быть выделены несколько разновидностей таких угроз:

* скрытый отказ в обслуживании, вызванный привлечением части ресурсов ИСПД на обработку пакетов, передаваемых злоумышленником со снижением пропускной способности каналов связи, производительности сетевых устройств, нарушением требований ко времени обработки запросов. Примерами реализации угроз подобного рода могут служить: направленный шторм эхо-запросов по *протоколу ICMP* (Ping *flooding*), шторм запросов на установление TCP-соединений (SYN-*flooding*), шторм запросов к FTP-серверу;
* явный отказ в обслуживании, вызванный исчерпанием ресурсов ИСПДн при обработке пакетов, передаваемых злоумышленником (занятие всей полосы пропускания каналов связи, переполнение *очередей запросов* на обслуживание), при котором легальные запросы не могут быть переданы через сеть из-за недоступности среды передачи либо получают отказ в обслуживании ввиду переполнения *очередей запросов*, дискового пространства памяти и т.д. Примерами угроз данного типа могут служить шторм широковещательных ICMP-эхо-запросов (Smurf), направленный шторм (SYN-*flooding*), шторм сообщений почтовому серверу (*Spam*);
* явный отказ в обслуживании, вызванный нарушением логической связности между техническими средствами ИСПДн при передаче нарушителем управляющих сообщений от имени сетевых устройств, приводящих к изменению маршрутно-адресных данных (например, ICMP Redirect Host, DNS-*flooding*) или идентификационной и аутентификационной информации;
* явный отказ в обслуживании, вызванный передачей злоумышленником пакетов с нестандартными атрибутами (угрозы типа "Land", "TearDrop", "Bonk", "*Nuke*", "UDP-*bomb*") или имеющих длину, превышающую максимально допустимый размер (угроза типа "Ping Death"), что может привести к сбою сетевых устройств, участвующих в обработке запросов, при условии наличия ошибок в программах, реализующих протоколы сетевого обмена.

Результатом реализации данной угрозы может стать нарушение работоспособности соответствующей службы предоставления удаленного доступа к ПД в ИСПД, передача с одного адреса такого количества запросов на подключение к техническому средству в составе ИСПД, какое максимально может "вместить" трафик (направленный "шторм запросов"), что влечет за собой *переполнение* очереди запросов и отказ одной из сетевых служб или полную остановку компьютера из-за невозможности системы заниматься ничем другим, кроме обработки запросов.

Мы рассмотрели наиболее часто реализуемые угрозы при сетевом взаимодействии. На практике угроз значительно больше. Частная модель угроз безопасности строится на основе двух документов ФСТЭК – "**Базовая модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных"** и "**Методика определения актуальных угроз безопасности персональных данных при их обработке в ИСПД"**. Если организация большая, и в ней несколько систем ИСПД, наиболее разумным решением будет привлечение квалифицированных специалистов сторонних фирм для построения частной модели угроз и проектирования СЗПД.

**4.4. Организационно-распорядительная документация по защите ПД**

Помимо технических и процедурных решений создаваемой системы *защиты персональных данных*, оператор должен обеспечить разработку организационно - распорядительных документов, которые будут регулировать все возникающие вопросы *по* обеспечению безопасности ПД при их обработке в ИСПД и эксплуатации системы *защиты персональных данных* (далее СЗПД). Таких документов достаточно много, основные из них:

1. **Положение по обеспечению безопасности ПД**. Это внутренний (локальный) документ организации. Строгой формы данного документа нет, но он должен удовлетворять требованиям ТК и ФЗ-152, а, следовательно, в нем должно быть указано:

* цель и задачи в области *защиты персональных данных*;
* понятие и состав персональных данных;
* в каких структурных подразделениях и на каких носителях (бумажных, электронных) накапливаются и хранятся эти данные;
* как происходит сбор и хранение персональных данных;
* как они обрабатываются и используются;
* кто (по должностям) в пределах фирмы имеет к ним доступ;
* принципы защиты ПД, в том числе от несанкционированного доступа;
* права работника в целях обеспечения защиты своих персональных данных;
* ответственность за разглашение конфиденциальной информации, связанной с персональными данными работников.

Положение *по* обеспечению безопасности персональных данных утверждается руководителем организации или уполномоченным им лицом, вводится в действие приказом руководителя. Работодатель обязан ознакомить работника с Положением… под подпись.

2. Для организации системы допуска и учета лиц, допущенных к работе с ПД в ИСПД, - *Список* лиц, допущенных к обработке ПД (перечень *по* должностям тех, кому *доступ* к ПД необходим для выполнения служебных обязанностей) и *Матрица* доступа (должна отражать полномочия пользователей *по* выполнению конкретных действий в отношении конкретных информационных ресурсов ИСПД – чтение, *запись*, корректировка, удаление).

В первую *очередь* *доступ* необходимо оформить сотрудникам кадровой службы, поскольку они собирают и формируют данные о работнике, а также сотрудникам бухгалтерии. Помимо того, *доступ* к этим сведениям могут получить руководители структурных подразделений (например, начальники отделов) – и это также необходимо отразить в списке. Однако все они вправе запрашивать не любые данные, а только те, которые необходимы для выполнения конкретных трудовых функций (например, чтобы рассчитать льготы *по* налогам, бухгалтерия получит не все сведения о работнике, а только данные о количестве его иждивенцев). Поэтому целесообразно прописать перечень информационных ресурсов, к которым пользователи допущены.

*Список* лиц, допущенных к обработке ПД, можно оформить в виде приложения к Положению *по* обеспечению безопасности персональных данных или отдельным документом, утвержденным руководителем.

3. Частная модель угроз (если ИСПД несколько, то модель угроз разрабатывается на каждую из них) – разрабатывается *по* результатам предварительного обследования. Страна предлагает Базовую модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных, согласно которой при создании частной модели должны быть рассмотрены:

* угрозы утечки информации по техническим каналам;
* угрозы несанкционированного доступа, связанные с действиями нарушителей, имеющих доступ к ИСПД, реализующих угрозы непосредственно в ИСПД. При этом необходимо в качестве потенциальных нарушителей рассматривать легальных пользователей ИСПД;
* угрозы несанкционированного доступа, связанные с действиями нарушителей, не имеющих доступа к ИСПД, реализующих угрозы из внешних сетей связи общего пользования и(или) сетей международного информационного обмена.

Разработанная модель угроз утверждается руководителем.

4. На основании утвержденной модели угроз ИСПД необходимо разработать требования *по* обеспечению безопасности ПД при их обработке в ИСПД. Требования, как и модель угроз, - это самостоятельный документ, который должен быть утвержден руководителем организации.

Для разработки модели угроз и требований оператору целесообразно привлекать специалистов организаций-лицензиатов ФСТЭК.

5. Инструкции в части обеспечения безопасности ПД при их обработке в ИСПД.

6. Рекомендации (инструкции) *по* использованию программных и аппаратных средств защиты информации.

Мы рассмотрели только основные организационно-распорядительные документы. Помимо перечисленных документов необходимо составить Акт классификации ИСПД, Технический паспорт ИСПД, Электронный журнал регистрации обращений пользователей ИСПД на получение ПД, Регламент разграничения прав доступа, приказы о назначении лиц, работающих с ИСПД и т.п.

Кроме того, до проведения всех мероприятий *по* защите ПД оператор должен назначить должностное лицо или (если ИСПД достаточно велика) структурное подразделение, ответственные за обеспечение безопасности ПД. Решение о назначении оформляется приказом руководителя. Задачи, функции и полномочия должностного лица (*подразделения*), ответственного за обеспечение безопасности ПД, определяются внутренними организационно-распорядительными документами (должностными инструкциями, регламентами).

**Лекция 5:**

**Порядок организации защиты персональных данных. Организационно-распорядительная документация**

**5.1.Общий порядок организации обеспечения безопасности персональных данных в информационных системах персональных данных**

Основные принципы и правила обеспечения безопасности ПД в информационных системах регулируются **"Положением об обеспечении безопасности персональных данных при их обработке в информационных системах персональных данных"**, утвержденным Постановлением Правительства При защите персональных данных должно быть обеспечено:

1. предотвращение несанкционированного доступа к ПД и передачи их лицам, не имеющим соответствующих прав;
2. своевременное обнаружение фактов НСД к ПД;
3. предотвращение воздействия на технические средства обработки ПД, которое может нарушить их функционирование;
4. возможность немедленного восстановления ПД в случае их модификации или уничтожения в результате НСД.
5. постоянный контроль уровня защищенности персональных данных[[18](https://intuit.ru/studies/courses/697/553/literature#literature.18)].

Организация безопасности ПД в ИСПД происходит в порядке, который предусматривает следующие этапы:

1. оценка обстановки;
2. обоснование требований безопасности ПД и постановка задач защиты;
3. разработка замысла обеспечения безопасности;
4. выбор мер и способов защиты в соответствии с требованиями безопасности и замыслом защиты;
5. решения вопросов управления защитой;
6. реализация замысла защиты;
7. планирование мероприятий по защите;
8. создание СЗПД;
9. разработка документов для эксплуатации СЗПД и организации обеспечения безопасности ИСПД.

Прежде всего, необходимо ограничить *физический доступ* к защищаемой информации. В основе защиты от физического доступа лежат организационные мероприятия. Для организации физической защиты помещений и технических средств обработки ПД в первую *очередь* документально заверяются границы контролируемой зоны, ограничивается *доступ* в помещения, где обрабатываются ПД, производится их охрана в нерабочее время, определяется порядок и специальное *место* хранения материальных носителей с ПД, опечатываются корпуса ПЭВМ.

**5.2. Оценка обстановки и формирование замысла защиты персональных данных**

Оценка обстановки является этапом, во многом определяющим эффективность решения задач обеспечения безопасности ПД. В ее основе лежит комплексное обследование организации и ИСПД, использующихся для обработки ПД. Прежде всего, определяется *информация*, которую необходимо защищать, производится ее *категорирование* и оценивается необходимость защиты от таких угроз, как уничтожение или хищение аппаратных средств или носителей с ПД, утечки информации по техническим каналам, от НСД и прочих рассмотренных ранее угроз.

При оценке обстановки учитывается степень ущерба в случае успешной реализации одной из угроз. Рассмотрим основные подэтапы оценки обстановки:

1. Анализ информационных ресурсов:
   * Определение состава, содержания и местонахождения ПД, подлежащих защите;
   * *Категорирование* ПД;
   * Оценка выполнения обязанностей по обеспечению безопасности ПД оператором в текущий момент времени.
2. *Анализ уязвимых* звеньев и возможных угроз безопасности ПД:
   * Оценка возможности физического доступа к ИСПД;
   * Выявление технических каналов утечки информации;
   * Анализ возможностей программно-математического воздействия на ИСПД;
   * Анализ возможностей электромагнитного воздействия на ПД.
3. Оценка ущерба от реализации угроз
   * Оценка непосредственного ущерба от реализации угроз;
   * Оценка опосредованного ущерба от реализации угроз;
4. Анализ имеющихся в распоряжении мер и средств защиты ПД:
   * от физического доступа;
   * от утечки по техническим каналам утечки информации;
   * от НСД;
   * от программно-математических воздействий;
   * от электромагнитных воздействий.

В зависимости от объекта, причинение ущерба которому, в конечном счете, вызывается неправомерными действиями с ПД, рассматриваются два вида ущерба: непосредственный и опосредованный.

**Непосредственный ущерб** связан с причинением физического, материального, финансового или морального вреда непосредственно субъекту ПД. Он возникает за счет незаконного использования (в том числе распространения) ПД или за счет несанкционированной модификации этих данных и может проявляться в виде:

* нанесения вреда здоровью субъекта ПД;
* незапланированных и (или) непроизводительных финансовых или материальных затрат субъекта;
* потери субъектом свободы действий вследствие шантажа и угроз, осуществляемых с использованием ПД;
* нарушения конституционных прав субъекта вследствие вмешательства в его личную жизнь путем осуществления контактов с ним по различным поводам без его на то согласия (например – рассылка персонифицированных рекламных предложений и т.п.).

**Опосредованный ущерб** связан с причинением вреда обществу и (или) государству вследствие нарушения нормальной деятельности экономических, политических, военных, медицинских, правоохранительных, социальных, кредитно-финансовых и иных государственных органов, органов местного самоуправления, муниципальных органов, организаций различных форм собственности за счет неправомерных действий с ПД.

Разработка замысла защиты является важным этапом построения СЗПД, в ходе которого определяются основные направления *защиты персональных данных*, и производится выбор способов защиты. К способам защиты относятся как технические средства, так и организационные меры. В качестве технических средств защиты следует использовать сертифицированные средства защиты. Основные этапы формирования замысла защиты показаны на [рисунке 5.1](https://intuit.ru/studies/courses/697/553/lecture/12448?page=1#image.5.1).



**Рис. 5.1.**Формирование замысла защиты персональных данных

К основным вопросам управления относятся:

1. распределение функций управления доступом к данным и их обработкой между должностными лицами;
2. определение порядка изменения правил доступа к защищаемой информации;
3. определение порядка изменения правил доступа к резервируемым информационным и аппаратным ресурсам;
4. определение порядка действий должностных лиц в случае возникновения нештатных ситуаций;
5. определение порядка проведения контрольных мероприятий и действий по его результатам.

Для поддержания эффективного уровня *защиты персональных данных* необходимо своевременно решать вопросы по управлению защитой, а также основные вопросы, такие как подготовка кадров, финансирование и закупка необходимого оборудования. Только *комплексный подход* может гарантировать достаточность принятых мер *защиты персональных данных*.

**Лекция 6:**

**Классификация ИСПД**

**Аннотация:**Цель лекции: рассмотреть критерии, на основании которых производится классификация ИСПД и цели классификации.

На начальном этапе построения СЗПД определяется *класс* для каждой ИСПД в конкретной организации. От класса ИСПД зависит, какие требования *по* обеспечению безопасности должны выполняться для заданной ИСПД, следовательно, выбор мер, способов защиты и их *стоимость*. Классификация ИСПД проводится в соответствии сертифицирующим органом страны. **"Об утверждении порядка проведения классификации информационных систем персональных данных"** (утратил силу 31 декабря 2013 г.).

Классификация ИСПД проводится на этапе ее создания или в ходе эксплуатации, но обязательно до построения СЗПД. В общем случае все *информационные системы*, обрабатывающие *персональные данные*, подразделяются на 2 *класса в зависимости* от характеристик безопасности обрабатываемых данных:

**Типовые информационные системы** – системы, где требуется обеспечить только *конфиденциальность* обрабатываемых персональных данных.

**Специальные информационные системы** – системы, где требуется обеспечить хотя бы одну из характеристик безопасности, отличную от конфиденциальности (например, *целостность* или доступность). К специальным информационным системам должны быть отнесены:

1. ИСПД, связанные с обработкой ПД о состоянии здоровья субъектов ПД;
2. ИСПД, принимающие решения на основании исключительно автоматизированной обработки ПД. При этом принятые решения могут повлечь за собой юридические последствия для субъекта ПД или иным способом затронуть его законные права и интересы.

Согласно предлагаемой в Приказе методике ИСПД классифицируется в зависимости от количества субъектов, чьи данные обрабатываются, и типа обрабатываемых персональных данных.

В зависимости от объема обрабатываемых в ИСПД данных XНПД выделяют следующие категории ИСПД:

**1 категория** – в информационной системе одновременно обрабатываются *персональные данные* **более чем 100 000** субъектов ПД или *персональные данные* субъектов ПД в пределах субъекта;

**2 категория** – в информационной системе одновременно обрабатываются *персональные данные* **от 1000 до 100 000**субъектов ПД или *персональные данные* субъектов ПД, работающих в отрасли экономики, в органе государственной власти, проживающих в пределах муниципального образования;

**3 категория** – в информационной системе одновременно обрабатываются *персональные данные* **менее чем 1000** субъектов ПД или *персональные данные* субъектов ПД в пределах конкретной организации.

Таким образом, данная категория ИСПД определяется на основании количества субъектов ПД, чьи данные обрабатываются в системе.

Определяются следующие категории обрабатываемых в информационной системе персональных данных (ХПД):

**категория 1** – *персональные данные*, касающиеся расовой, национальной принадлежности, политических взглядов, религиозных и философских убеждений, состояния здоровья, интимной жизни;

**категория 2** – *персональные данные*, позволяющие идентифицировать субъекта персональных данных и получить о нем дополнительную информацию, за исключением персональных данных, относящихся к категории 1;

**категория 3** – *персональные данные*, позволяющие идентифицировать субъекта персональных данных;

**категория 4** – обезличенные и (или) общедоступные *персональные данные*.

*По* результатам анализа вышеперечисленных данных определяется *класс* ИСПД в соответствии с [таблицей 6.1](https://intuit.ru/studies/courses/697/553/lecture/12450?page=1#table.6.1).

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 6.1. Определение класса информационной системы | | | |
| **ХНПД** | **Категория 3** | **Категория 2** | **Категория 1** |
| **ХПД** |  | | |
| категория 4 | К4 | К4 | К4 |
| категория 3 | К3 | К3 | К2 |
| категория 2 | К3 | К2 | К1 |
| категория 1 | К1 | К1 | К1 |

Рассмотрим, что значит каждый *класс* ИСПД в отдельности:

* **класс 1 (К1)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, может привести к значительным негативным последствиям для субъектов ПД;
* **класс 2 (К2)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, может привести к негативным последствиям для субъектов ПД;
* **класс 3 (К3)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, может привести к незначительным негативным последствиям для субъектов ПД;
* **класс 4 (К4)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, не приводит к негативным последствиям для субъектов ПД.

Наивысшим считается *класс* 1. Если в составе ИСПД выделяют несколько подсистем, то *класс* ИСПД в целом будет соответствовать наиболее высокому классу входящих компонентов.

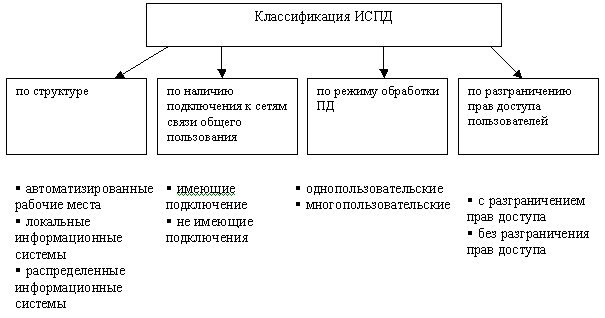
Таким образом, чем выше *класс* ИСПД, тем выше требования *по* обеспечению безопасности персональных данных.

Порядок определения класса для специальных систем несколько отличается от типовых. *Класс* специальных ИСПД определяется на основе частной модели угроз организации в соответствии с методическими документами ФСТЭК. Отнесение информационной системы к специальной позволяет существенно снизить *затраты* на построение СЗПД, так как оператор в данном случае может обоснованно выбрать минимальное количество актуальных угроз, от которых необходима защита ПД. Например, если в системе есть сведения о доходах человека (например, 1С), такая система может быть отнесена к специальной, так как затрагиваются законные интересы человека. То же самое относится к информации об инвалидности, расовой принадлежности и пр. Отнесение ИСПД к специальной на практике является достаточно спорным моментом.

*Класс* ИСПД может быть пересмотрен:

* по решению оператора после проведения анализа и оценки угроз безопасности с учетом особенностей или (и) изменений информационной системы;
* по результатам испытаний по контролю за выполнением требований по обработке ПД.

ИСПД также классифицируются *по* структуре, *по* наличию подключений к сетям общего пользования, *по* режиму обработки ПД, *по* разграничению прав доступа. Приведенные классификации отображены схематически на [рисунке 6.1](https://intuit.ru/studies/courses/697/553/lecture/12450?page=1#image.6.1).



**Рис. 6.1.**

Классификация ИСПД - не формальная процедура, так как на основании класса системы будут определяться меры и способы *защиты персональных данных*, существенно отличающиеся друг от друга. Более того, для 1 и 2 класса систем обязательно *лицензирование* используемых технических средств защиты, а сами ИСПД должны быть аттестованы. Результат классификации оформляется соответствующим актом оператора.

При классификации информационных систем на практике выявляется множество факторов, впоследствии влияющих на построение СЗПД. Например, если в информационной системе помимо стандартных данных о сотрудниках, таких как ФИО, *адрес* и телефон, хранятся фотографии, значит, система обрабатывает биометрические ПД. А если наряду со стандартной информацией обрабатывается дата рождения, можно сказать, что ИСПД содержит дополнительную информацию о субъекте и отнести ее не к классу 3, а к классу 2.

При внимательном анализе на предпроектном этапе построения СЗПД, можно оптимизировать некоторые моменты обработки ПД. Так, например, если в системе учета кадров вместо ФИО использовать табельный номер работника, речь будет идти об обработке обезличенных персональных данных, требования к безопасности которых минимальны. *Оптимизация* представления данных в системе может повлиять на проектирование СЗПД и существенно сократить *затраты* в конечном итоге.

г. **"Об утверждении порядка проведения классификации информационных систем персональных данных"** (утратил силу 31 декабря 2013 г.).

Классификация ИСПД проводится на этапе ее создания или в ходе эксплуатации, но обязательно до построения СЗПД. В общем случае все *информационные системы*, обрабатывающие *персональные данные*, подразделяются на 2 *класса в зависимости* от характеристик безопасности обрабатываемых данных:

**Типовые информационные системы** – системы, где требуется обеспечить только *конфиденциальность* обрабатываемых персональных данных.

**Специальные информационные системы** – системы, где требуется обеспечить хотя бы одну из характеристик безопасности, отличную от конфиденциальности (например, *целостность* или доступность). К специальным информационным системам должны быть отнесены:

1. ИСПД, связанные с обработкой ПД о состоянии здоровья субъектов ПД;
2. ИСПД, принимающие решения на основании исключительно автоматизированной обработки ПД. При этом принятые решения могут повлечь за собой юридические последствия для субъекта ПД или иным способом затронуть его законные права и интересы.

Согласно предлагаемой в Приказе методике ИСПД классифицируется в зависимости от количества субъектов, чьи данные обрабатываются, и типа обрабатываемых персональных данных.

В зависимости от объема обрабатываемых в ИСПД данных XНПД выделяют следующие категории ИСПД:

**1 категория** – в информационной системе одновременно обрабатываются *персональные данные* **более чем 100 000** субъектов ПД или *персональные данные* субъектов ПД в пределах субъекта;

**2 категория** – в информационной системе одновременно обрабатываются *персональные данные* **от 1000 до 100 000**субъектов ПД или *персональные данные* субъектов ПД, работающих в отрасли экономики, в органе государственной власти, проживающих в пределах муниципального образования;

**3 категория** – в информационной системе одновременно обрабатываются *персональные данные* **менее чем 1000** субъектов ПД или *персональные данные* субъектов ПД в пределах конкретной организации.

Таким образом, данная категория ИСПД определяется на основании количества субъектов ПД, чьи данные обрабатываются в системе.

Определяются следующие категории обрабатываемых в информационной системе персональных данных (ХПД):

**категория 1** – *персональные данные*, касающиеся расовой, национальной принадлежности, политических взглядов, религиозных и философских убеждений, состояния здоровья, интимной жизни;

**категория 2** – *персональные данные*, позволяющие идентифицировать субъекта персональных данных и получить о нем дополнительную информацию, за исключением персональных данных, относящихся к категории 1;

**категория 3** – *персональные данные*, позволяющие идентифицировать субъекта персональных данных;

**категория 4** – обезличенные и (или) общедоступные *персональные данные*.

*По* результатам анализа вышеперечисленных данных определяется *класс* ИСПД в соответствии с [таблицей 6.1](https://intuit.ru/studies/courses/697/553/lecture/12450?page=1#table.6.1).

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 6.1. Определение класса информационной системы | | | |
| **ХНПД** | **Категория 3** | **Категория 2** | **Категория 1** |
| **ХПД** |  | | |
| категория 4 | К4 | К4 | К4 |
| категория 3 | К3 | К3 | К2 |
| категория 2 | К3 | К2 | К1 |
| категория 1 | К1 | К1 | К1 |

Рассмотрим, что значит каждый *класс* ИСПД в отдельности:

* **класс 1 (К1)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, может привести к значительным негативным последствиям для субъектов ПД;
* **класс 2 (К2)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, может привести к негативным последствиям для субъектов ПД;
* **класс 3 (К3)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, может привести к незначительным негативным последствиям для субъектов ПД;
* **класс 4 (К4)** – информационные системы, для которых нарушение заданной характеристики безопасности персональных данных, обрабатываемых в них, не приводит к негативным последствиям для субъектов ПД.

Наивысшим считается *класс* 1. Если в составе ИСПД выделяют несколько подсистем, то *класс* ИСПД в целом будет соответствовать наиболее высокому классу входящих компонентов.

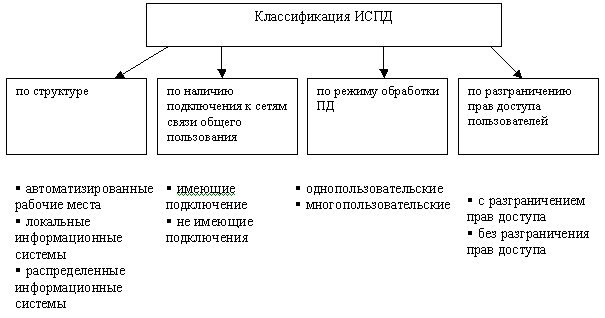
Таким образом, чем выше *класс* ИСПД, тем выше требования *по* обеспечению безопасности персональных данных.

Порядок определения класса для специальных систем несколько отличается от типовых. *Класс* специальных ИСПД определяется на основе частной модели угроз организации в соответствии с методическими документами ФСТЭК. Отнесение информационной системы к специальной позволяет существенно снизить *затраты* на построение СЗПД, так как оператор в данном случае может обоснованно выбрать минимальное количество актуальных угроз, от которых необходима защита ПД. Например, если в системе есть сведения о доходах человека (например, 1С), такая система может быть отнесена к специальной, так как затрагиваются законные интересы человека. То же самое относится к информации об инвалидности, расовой принадлежности и пр. Отнесение ИСПД к специальной на практике является достаточно спорным моментом.

*Класс* ИСПД может быть пересмотрен:

* по решению оператора после проведения анализа и оценки угроз безопасности с учетом особенностей или (и) изменений информационной системы;
* по результатам испытаний по контролю за выполнением требований по обработке ПД.

ИСПД также классифицируются *по* структуре, *по* наличию подключений к сетям общего пользования, *по* режиму обработки ПД, *по* разграничению прав доступа. Приведенные классификации отображены схематически на [рисунке 6.1](https://intuit.ru/studies/courses/697/553/lecture/12450?page=1#image.6.1).



**Рис. 6.1.**

Классификация ИСПД - не формальная процедура, так как на основании класса системы будут определяться меры и способы *защиты персональных данных*, существенно отличающиеся друг от друга. Более того, для 1 и 2 класса систем обязательно *лицензирование* используемых технических средств защиты, а сами ИСПД должны быть аттестованы. Результат классификации оформляется соответствующим актом оператора.

При классификации информационных систем на практике выявляется множество факторов, впоследствии влияющих на построение СЗПД. Например, если в информационной системе помимо стандартных данных о сотрудниках, таких как ФИО, *адрес* и телефон, хранятся фотографии, значит, система обрабатывает биометрические ПД. А если наряду со стандартной информацией обрабатывается дата рождения, можно сказать, что ИСПД содержит дополнительную информацию о субъекте и отнести ее не к классу 3, а к классу 2.

При внимательном анализе на предпроектном этапе построения СЗПД, можно оптимизировать некоторые моменты обработки ПД. Так, например, если в системе учета кадров вместо ФИО использовать табельный номер работника, речь будет идти об обработке обезличенных персональных данных, требования к безопасности которых минимальны. *Оптимизация* представления данных в системе может повлиять на проектирование СЗПД и существенно сократить *затраты* в конечном итоге.

Лекция 8:

**Профили защиты, порядок их регистрации и сертификации**

**A**

 |

**Аннотация:**Цель лекции: дать определение профиля защиты, изучить его структуру и содержание, определить необходимость его использования и порядок сертификации.

**8.1. Понятие профиля защиты**

Требования к безопасности конкретных средств и информационных систем устанавливаются на основании угроз, которые уже есть или прогнозируются, в соответствии с политикой безопасности и условий применения этих средств(систем). Требования, которые являются общими для некоторого типа продуктов или информационных систем, можно объединить в структуру, называемую профилем защиты. Критерии оценки безопасности информационных технологий"

**профиль защиты (ПЗ)** - это независимая от реализации совокупность требований безопасности для некоторой категории изделий ИТ, отвечающая специфическим запросам потребителя.

**Продукт ИТ** - совокупность программных, программно-аппаратных и/или аппаратных средств ИТ, предоставляющая определенные функциональные возможности и предназначенная для непосредственного использования или включения в различные системы ИТ.

**Изделие ИТ** - обобщенный термин для продуктов и систем ИТ[[23](https://intuit.ru/studies/courses/697/553/literature#literature.23)].

ПЗ не регламентирует, каким образом должны быть выполнены данные требования, тем самым предоставляя возможность разработчику системы защиты самостоятельно выбирать средства защиты. ПЗ может применяться либо к определенному классу продуктов, например, операционным системам или межсетевым экранам, и к совокупности продуктов, образующих систему информационной технологии (например, виртуальные частные *сети, PKI*). Использование профилей защиты преследует три основные задачи:

1. стандартизация наборов требований к информационным продуктам;
2. *оценка безопасности*;
3. проведение сравнительного анализа уровней безопасности различных изделий ИТ.

ПЗ подлежат оценке, регистрации и сертификации в соответствии с руководящими документами сертифицирующими органами.

Разработчиком ПЗ может быть как юридическое, так и физическое лицо.

ПЗ должен содержать:

1. потребности пользователя *изделия ИТ* в обеспечении информационной безопасности;
2. описание *среды безопасности* *изделия ИТ* – обоснованность применения данного ИТ с учетом угроз среды, политики безопасности и пр.
3. цели безопасности *изделия ИТ*- то есть что должно быть сделано в результате использования данного ИТ;
4. функциональные требования к безопасности и *требования доверия* к безопасности. Функциональные требования отображают то, что должно выполнять ИТ и его среда, а *требования доверия* к безопасности отображают степень уверенности в функционале данного ИТ. Совокупность этих требований должна обеспечить достижение целей безопасности;
5. обоснование достаточности выдвинутых требований.

*Требования безопасности* зависят от класса защищенности *изделия ИТ*, который в свою *очередь* зависит от ценности информации, потенциальных угроз, развития соответствующих средств безопасности на настоящий момент и стоимости и времени на оценку безопасности данного *изделия ИТ*. Для определения требований безопасности в зависимости от класса защищенности *изделия ИТ* необходимо использовать Руководящий документ "Руководство по разработке семейств профилей защиты".

**8.2. Содержание профиля защиты**

Структура ПЗ в общем виде приведена на [рисунке 8.1](https://intuit.ru/studies/courses/697/553/lecture/12453?page=1#image.8.1).



**Рис. 8.1.**

*Идентификация* ПЗ должна обеспечить маркировку и описательную информацию, необходимые для однозначной идентификации и регистрации ПЗ.

Аннотация должна давать общую характеристику ПЗ и иметь описательную форму. При этом она должна быть достаточно полной для определения потенциальным пользователем необходимости использования данного ПЗ.

В раздел "Описание *изделия ИТ*" включается сопроводительная *информация* об изделии ИТ, предназначенная для пояснения его назначения и требований безопасности.

В раздел ПЗ "*Среда безопасности* *изделия ИТ*" включается описание аспектов *среды безопасности* *изделия ИТ*: *предположения безопасности*, потенциальных угроз и политики безопасности организации. *Предположения безопасности* содержат описание среды, в котором изделие будет использоваться:

* информация относительно предполагаемого использования ИТ - предполагаемая область применения, потенциальная значимость активов и возможные ограничения использования;
* информацию относительно среды применения, включая аспекты физического окружения, персонала и внешних связей.

*Цели безопасности* должны быть четко изложены, отражать намерение противостоять определенному набору угроз и *соответствовать политике безопасности*.

В раздел ПЗ "*Требования безопасности* *изделия ИТ*" включаются функциональные *требования безопасности* *изделия ИТ*, *требования доверия* к безопасности, а также *требования безопасности* программного, программно-аппаратного и аппаратного обеспечения ИТ-среды *изделия ИТ*.

В раздел ПЗ "Замечания по применению ПЗ" может включаться любая дополнительная *информация*, которую разработчик ПЗ считает полезной. Замечания по применению могут быть распределены по соответствующим разделам ПЗ [[25](https://intuit.ru/studies/courses/697/553/literature#literature.25)].

В разделе ПЗ "Обоснование" демонстрируется, что ПЗ специфицирует полную и взаимосвязанную совокупность требований безопасности *изделия ИТ*, и что соответствующее *изделие ИТ* учитывает идентифицированные аспекты *среды безопасности*. Раздел "Обоснование" может быть оформлен в виде отдельного документа.

Сертификат ПЗ - документ, удостоверяющий соответствие ПЗ критериям, приведенным в разделе 4 части 3 Руководящего документа "Критерии оценки безопасности информационных технологий". *Цель оценки* ПЗ - показать полноту, *непротиворечивость* и техническую верность конкретного ПЗ. Для повышения согласованности выводов, полученных при оценке, ее результаты могут быть представлены на сертификацию. *Сертификация* представляет собой независимую инспекцию результатов оценки, которая завершается их утверждением или выдачей сертификата. Сведения о сертификатах обычно публикуются и являются общедоступными.

**8.3. Регистрация профиля защиты**

После составления *профиля защиты* и его оценки (сертификации) он должен быть зарегистрирован в соответствии с Руководящим документом *Безопасность* информационных технологий. Руководство по регистрации профилей защиты". Данный документ определяет процедуру регистрации не только ПЗ, но и пакетов.**Пакет** - многократно используемая совокупность функциональных компонентов или компонентов доверия, объединенных для достижения определенных целей безопасности [[25](https://intuit.ru/studies/courses/697/553/literature#literature.25)]. В результате регистрации ПЗ (или пакет) получает регистрационную метку, которая уникально его идентифицирует в специальном реестре. **Реестр** - совокупность записей (в электронном или электронном и бумажном виде), включающих в себя регистрационные метки, а также связанную с ними дополнительную информацию[[26](https://intuit.ru/studies/courses/697/553/literature#literature.26)]. Каждая *запись* в реестре состоит из трех частей, разделенных дефисом:

* тип элемента реестра;
* год регистрации;
* регистрационный номер.

Тип элемента реестра может иметь три значения:

* "ПЗ" – для *профиля защиты*;
* "ПД" для пакета требований доверия;
* "ФП" – для *функционального пакета*.

Год регистрации — год внесения элемента в реестр (четыре цифры).

Регистрационный номер — порядковый номер в текущем году (три цифры).

Пример: ПЗ-2010-005.

Для регистрации ПЗ заявителю необходимо отправить в регистрационный орган (ОР) заявку. Заявка обязательно должна содержать следующую информацию:

* Название организации (если заявитель юридическое лицо) или ФИО (если заявитель физическое лицо) и контактную информацию. Контактная информация должна включать почтовый адрес и/или адрес E-mail, номер телефона и/или факса.
* Тип объекта, который заявитель хочет зарегистрировать.
* Определение представленного объекта как нового или заменяющего уже существующий элемент в реестре. Если объект заменяющий, необходимо указать регистрационные метки элементов, подлежащих удалению или замене.
* Подтверждение от заявителей заменяемых элементов, что при принятии заменяющих элементов они согласны на удаление своих элементов.
* Определение объекта как завершенного или в качестве проекта.
* Описание нового ПЗ или пакета
* Аннотацию ПЗ или пакета.
* Декларацию, что описание ПЗ или пакета, представленного для регистрации, удовлетворяет требованиям руководящего документа по регистрации.

ОР в свою *очередь* либо отклоняет заявку(если в ней не полный перечень информации), либо присваивает регистрационную метку и вводит в реестр со статусом "проходящий подтверждение соответствия". Затем о результате этой процедуры сообщается заявителю. Начальная обработка занимает не более 14 дней после получения заявки. После этого ОР выполняет проверку описания, представленного в заявке. Если выявляются какие-то недостатки или несоответствия, ОР уведомляет заявителя и тот в свою *очередь* должен внести коррективы в течение 14 дней. Процедура подтверждения соответствия должна быть завершена в течение 3 месяцев с момента получения заявки на регистрацию.

Лекция 9:

**Построение системы защиты персональных данных**

**A**

**Аннотация:**Цель лекции: определить содержание этапов организации обеспечения безопасности ПД, а также необходимые организационные и технические мероприятия в рамках построения СЗПД.

**9.1. Основные этапы при построении системы защиты персональных данных**

Система защиты ПД может быть как отдельной системой, так и подсистемой в составе системы защиты информации организации в целом. Как правило, выполнение *работ* по построению СЗПД происходит поэтапно и включает в себя следующие стадии:

1. предпроектная стадия или оценка обстановки;
2. стадия проектирования;
3. ввод в действие СЗПД.

Предпроектная стадия или оценка обстановки. В самом начале построения СЗПД производится оценка обстановки. На данном этапе производятся следующие работы:

1. разрабатывается перечень информационных систем организации, которые работают с ПД;
2. определение состава ПД и необходимость их обработки;
3. определение перечня ПД, которые необходимо защищать;
4. определение контролируемой зоны и расположения компонентов ИСПД относительно границ этой зоны;
5. строится модель корпоративной сети;
6. определение топологии и конфигурации ИСПД, программ и технических средств, которые используются или предполагаются к использованию для обработки ПД;
7. установление связей ИСПД с другими системами организации;
8. определение режимов обработки ПД;
9. определение угроз безопасности;
10. определение класса ИСПД;
11. уточняется степень участия персонала в обработке ПД.

Важным пунктом данного этапа является построение **частной модели угроз** и предварительная **классификация ИСПД** данной организации. Классификация ИСПД производится в соответствии с приказом "**Об утверждении Порядка проведения классификации информационных систем персональных данных"**от 13 февраля 2008 года.Результатом данного этапа является формирование частного технического задания (ТЗ) к СЗПД .

*Техническое задание* должно содержать:

* обоснование разработки СЗПД;
* исходные данные создаваемой (модернизируемой) ИСПД в техническом, программном, информационном и организационном аспектах;
* класс ИСПД;
* ссылку на нормативные документы, с учетом которых будет разрабатываться СЗПД и приниматься в эксплуатацию ИСПД;
* конкретизацию мероприятий и требований к СЗПД;
* перечень предполагаемых к использованию сертифицированных средств защиты информации;
* обоснование проведения разработок собственных средств защиты информации при невозможности или нецелесообразности использования имеющихся на рынке сертифицированных средств защиты информации;
* состав, содержание и сроки проведения работ по этапам разработки и внедрения СЗПД.

1. стадия проектирования. На данном этапе разрабатывается задание и проект проведения работ в соответствии с ТЗ, выполняются все требуемые работы, в том числе закупка технических средств защиты и их сертификация в случае необходимости. Разрабатывается система доступа к ПД должностных лиц и определяются ответственные за эксплуатацию средств защиты информации.

Важным подэтапом является разработка эксплуатационной документации на ИСПД и средства защиты информации, а также организационно-распорядительной документации по защите информации (приказов, инструкций и других документов).

Этап проектирования является наиболее важным и трудоемким, так как при реализации СЗПД необходимо учесть множество факторов, таких как масштабирование, совместимость средств защиты со штатным программным обеспечением, возможность периодического тестирования системы защиты и замены отдельных компонентов системы в случае необходимости.

1. ввод в действие СЗПД. Данный этап включает в себя:
   * генерация пакета прикладных программ в комплексе с программными средствами защиты информации;
   * опытная эксплуатация средств защиты информации в комплексе с другими техническими и программными средствами;
   * приемо-сдаточные испытания;
   * организация охраны и физической защиты помещений ИСПД;
   * оценка соответствия ИСПД требованиям безопасности ПД.

В случае если в процессе опытной эксплуатации выявляются недостатки разработанной СЗПД, проводится ее доработка.

**9.2. Комплекс организационных и технических мероприятий в рамках СЗПД**

Для обеспечения безопасности персональных данных организации требуется провести комплекс технических и организационных мероприятий в рамках построения СЗПД и ее эксплуатации.

Организационные меры носят административный и процедурный характер и регламентируют процессы функционирования ИСПД, обработку ПД и действия персонала. Организационные меры включают в себя:

1. разработка организационно-распорядительных документов, предназначенных для регламентации процессов хранения, обработки, сбора и накопления персональных данных, а также их защиту;
2. уведомление уполномоченного органа о намерении обрабатывать ПД;
3. получение письменного согласия на обработку ПД от субъектов ПД;
4. определение должностных лиц, которые будут работать с ПД;
5. организация доступа в помещения, где будет вестись обработка ПД;
6. разработка должностных инструкций по работе с ПД;
7. определение сроков хранения ПД;
8. планирование мероприятий по защите ПД;
9. обучение персонала.

Организационные меры индивидуальны для каждой организации и в первую *очередь* определяются классом ИСПД, которые обрабатывают *персональные данные*. Чем выше *класс* ИСПД, тем больше мероприятий организационного характера требуется для ее защиты. Организационные меры необходимы для регламентации функционирования системы в целом, но явно не достаточны. Для обеспечения безопасности они должны быть подкреплены использованием технических средств защиты.

Техническое мероприятие – это мероприятие по защите информации, предусматривающее применение специальных технических средств, а также реализацию технических решений. Техническая защита по своей структуре и содержанию является более сложным и трудоемким процессом в отличие от организационных мероприятий и предусматривает выполнение следующих условий:

1. для выполнения работ по технической защите требуется лицензия;
2. для выбора адекватных и достаточных средств защиты необходимо тщательное обследование ИСПД, построение модели угроз и классификация ИСПД;
3. на основе данного обследования формируется перечень требований по обеспечению безопасности;
4. требуется провести работы по проектированию, созданию и вводу в эксплуатацию СЗПД;
5. для проведения аттестации(сертификации) на соответствие ИСПД требованиям законодательства необходимо наличие соответствующих лицензий.

Согласно Указу Президента "Об утверждении перечня сведений конфиденциального характера" *персональные данные* относятся к категории сведений конфиденциального характера. На основании закона "О лицензировании отдельных видов деятельности" техническая защита конфиденциальной информации (в нашем случае персональных данных) относится к лицензированному виду деятельности. Таким образом, для проведения мероприятий по защите персональных данных необходимо привлекать организации, имеющие соответствующие лицензии. *Деятельность* по защите информации без наличия соответствующих лицензий влечет за собой как административную, так и уголовную ответственность.

**9.3. Уведомление об обработке персональных данных**

До начала обработки персональных данных оператор обязан уведомить о своем намерении в соответствии "О персональных данных".Уведомление должно быть в письменной форме с подписью уполномоченного лица или в электронной форме с *ЭЦП*. Уведомление должно содержать:

* наименование (фамилия, имя, отчество), адрес оператора;
* цель обработки персональных данных;
* категории персональных данных;
* категории субъектов, персональные данные которых обрабатываются;
* правовое основание обработки персональных данных;
* перечень действий с персональными данными, общее описание используемых оператором способов обработки персональных данных;
* описание мер, которые оператор обязуется осуществлять при обработке персональных данных, по обеспечению безопасности персональных данных при их обработке;
* дата начала обработки персональных данных;
* срок или условие прекращения обработки персональных данных.

В *поле* **цель обработки персональных данных** указываются цели обработки персональных данных, указанные в учредительных документах оператора, либо фактические цели обработки.

В *поле* **категории персональных данных** перечисляются все категории обрабатываемых ПД (основные, специальные, биометрические).

В *поле* **категории субъектов персональных данных**указываются категории физических лиц и виды отношений с ними, *персональные данные* которых обрабатываются**(**например: работники, состоящие в трудовых отношениях с Оператором, физические лица (*абонент*, пассажир, заемщик, вкладчик, страхователь, заказчик и др.), состоящие в договорных и иных гражданско-правовых отношениях Оператором и др.).

В *поле* **правовое основание обработки персональных данных** необходимо указать не только соответствующие статьи из ФЗ "О персональных данных", но и статьи из других правовых документов, регламентирующих обработку ПД в данном случае. Например, при обработке ПД сотрудников - ст. 85-90 Трудового кодекса РК.

В этом *поле* также указывается номер, дата выдачи и наименование лицензии на осуществляемый вид деятельности, с указанием на пункты, в которых предусмотрено *исключение* передачи ПД третьим лицам без согласия субъекта ПД.

В *поле* **перечень действий с персональными данными, общее описание используемых Оператором способов обработки персональных данных**, указываются действия, совершаемые Оператором с персональными данными, а также описание используемых Оператором способов обработки персональных данных:

* неавтоматизированная обработка персональных данных;
* исключительно автоматизированная обработка персональных данных с передачей полученной информации по сети или без таковой;
* смешанная обработка персональных данных.

При этом, если обработка автоматизированная или смешанная, необходимо указать, передается *информация* только по локальной сети или с использованием Интернета.

В *поле* **описание мер**описываются меры, предусмотренные статьями 18 и 19 ФЗ "О персональных данных", в том числе сведения о шифровальных средствах, ФИО и контакты лиц, ответственных за обработку ПД, *класс* ИСПД, организационные и технические меры, применяемые оператором в целях защиты ПД.

В *поле* срок или условие прекращения обработки ПД указывается конкретная дата или условие, при выполнении которого обработка будет прекращена.

**Лекция 10. Технические меры защиты конфиденциальной информации**

Эта категория мер дополняет организационные и устраняет их недостатки. Реализуются путем использования технических решений.

Для их реализации могут использоваться такие решения, как:

* Антивирусы для защиты отдельных компьютеров и целых сетей.
* Межсетевые экраны для фильтрации трафика.
* Системы предотвращения вторжений (IPS).
* Системы обнаружения вторжений (IDS).
* [Системы предотвращения утечек (DLP).](https://rt-solar.ru/products/solar_dozor/)
* Решения для мониторинга IT-инфраструктур.
* VPN.
* SIEM-системы для сбора и анализа сведений от различных компонентов системы информационной безопасности.
* СКУД и их отдельные компоненты (видеонаблюдение, сигнализация и т.д.).
* Аппаратные и программные средства для аутентификации и авторизации пользователей.

В некоторых источниках встречается упоминания про морально-этические меры защиты конфиденциальной информацию. К ним относят устоявшиеся в обществе нормы поведения. Понятно, что об эффективности таких мер говорить не приходится, ведь здесь все держится, можно сказать, на доверии и принятии человеком сложившихся общественных норм. Но все равно, некоторые компании умудряются использовать их в своей работе, даже оформляя в письменном виде (например, в форме кодекса чести и подобных документов).

**Тема:**Защита информации, антивирусная защита.

 План:

1.                      Защита информации.

2.                      Компьютерные вирусы и антивирусные программы.

**1. Защита информации**

|  |
| --- |
|  |
|  |  |

**Информационная безопасность** – это обеспечение защищенности информации от случайных или преднамеренных воздействий естественного или искусственного характера, чреватых нанесением ущерба владельцам или пользователям информации.

**Компьютерные вирусы и антивирусные программы**

Вредоносные программы можно разделить на три класса: черви, вирусы и троянские программы.

Черви — это класс вредоносных программ, использующих для распространения сетевые ресурсы. Используют сети, электронную почту и другие информационные каналы для заражения компьютеров.

Вирусы — это программы, которые заражают другие программы — добавляют в них свой код, чтобы получить управление при запуске зараженных файлов.

Троянские программы — программы, которые выполняют на поражаемых компьютерах несанкционированные пользователем действия, т.е. в зависимости от каких-либо условий уничтожают информацию на дисках, приводят систему к зависанию, воруют конфиденциальную информацию и т.д.

**Компьютерные вирусы** — это специально написанные программы, способные самопроизвольно без ведома пользователя присоединяться к другим программам, создавать свои копии и внедрять их в файлы, системные области дисков и компьютерные сети с целью нарушения работы программ, порчи файловой системы, создание всевозможных помех в работе компьютера.

Компьютерные вирусы имеют много сходного с живыми микроорганизмами, это:

·  скрытность;

·  способность к размножению;

·  приспособляемость к среде;

·  передвижение;

·  самопроизвольное внедрение в другие объекты и т. д.

**Классификация компьютерных вирусов**

|  |  |  |
| --- | --- | --- |
| Основные компьютерные вирусы | | |
| По среде обитания | По поражению ОС | По деструктивным возможностям |
| Файловые вирусы | DOS - вирусы | Безвредные |
| Загрузочные вирусы | OS/2 – вирусы | Неопасные вирусы |
| Сетевые вирусы | Windows - вирусы | Опасные вирусы |
| Макровирусы | Мас OS - вирусы | Очень опасные |

Обеспечить абсолютную защиту информации от компьютерного вируса невозможно. Однако соблюдение определённых правил позволит снизить вероятность заражения или тяжёлых последствий в результате заражения:

· не покупайте и не используйте диски с незарегистрированными авторскими программами;

· организуйте учёт каждого своего диска;

· контролируйте размеры всех файлов – системных и рабочих;

· не запускайте на своём компьютере сомнительные программы, в первую очередь, файлы, присланные незнакомыми людьми;

· будьте очень осторожны при получении электронных писем, например, письмо должно быть на русском, а оно пришло на английском языке;

· если вы подключены к глобальной сети, не копируйте в свой компьютер сомнительные игры или программы;

· широко используйте антивирусные программы;

· если имеется возможность, приобретите два компьютера, один для работы, другой для игр и работы в Интернет;

· чаще инструктируйте пользователей об опасности электронной эпидемии, объясняйте пути проникновения вируса в компьютер.

Но, если всё же произошло заражение компьютерным вирусом вашей электронной машины, придерживайтесь следующих рекомендаций:

· попытайтесь уничтожить вирус, пользуясь специальными антивирусными программами;

· если вы затрудняетесь найти и удалить вирус, пригласите более опытных пользователей, чтобы они помогли вам его уничтожить;

· обратите внимание, подключён ли ваш компьютер к сети, или подключены ли другие ПК к вашей ЭВМ, в этом случае немедленно отключите все компьютеры от сети.

Различают несколько видов антивирусных программ:

|  |
| --- |
|  |
|  |  |

 Иммунизаторы (вакцины) применяют в случае, если отсутствуют программы – доктора, которые лечат этот вирус. Вакцинация возможна только от известных вирусов. Вакцина модифицирует программу или диск таким образом, чтобы это не отражалось на нормальной работе программ в компьютере. Вирус же будет воспринимать программы заражёнными и не внедрится в них.

Антивирусные программы разработаны для обнаружения, удаления и защиты от компьютерных вирусов. Наиболее известные:

•   Антивирус Касперского (AVP) лаборатории Касперского

•    Adinf, DoctorWeb, Aidstest — фирмы «Диалог-Наука»

•    Norton Antivirus от Symantec.

•    NOD32 от компании ESET

Антивирусные программы могут использовать различные принципы для поиска и лечения зараженных файлов.

**Программы-мониторы** постоянно находятся в оперативной памяти компьютера и проверяют все файлы в реальном режиме времени.

**Программы-сканеры** производят проверку системы по команде пользователя.

*Правовая охрана программ и данных*

Программы по их юридическому статусу можно разделить на группы:

•   Лицензионные (коммерческие),

•   Бесплатные (freeware)

•   условно бесплатные (shareware)

•   свободно распространяемые программы (open source).

**Лекция 11 -12 . Безопасность на транспортном уровне: SSL и TLS**

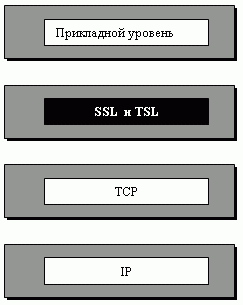
**Аннотация:**Эта лекция имеет несколько целей: обсудить потребности в услугах безопасности на транспортном уровне модели Интернет, обсудить общую архитектуру SSL, обсудить общую архитектуру TLS, сравнить и показать отличия SSL и TLS.

Конфиденциальность данных - это статус, предоставленный данным и определяющий требуемую степень их защиты. По существу - это свойство информации быть известной только допущенным и прошедшим проверку (авторизированным) субъектам системы (пользователям, процессам, программам). Для остальных субъектов системы эта информация должна быть неизвестной.

*Безопасность транспортного уровня* обеспечивает услуги безопасности "из конца в конец" для приложений, которые используют протоколы транспортного уровня, такие как *TCP*. Основные идеи предназначены для того, чтобы обеспечить услуги безопасности на сети *Интернет*. Например, когда в сети имеются интерактивно работающие *онлайн*(*online*)-магазины, то желательны следующие услуги безопасности:

1. Клиент должен убедиться, что сервер принадлежит фактическому продавцу, а не самозванцу. Клиент не хочет сообщать самозванцу номер кредитной карточки (*установление подлинности объекта*).
2. Клиент и продавец должны быть убеждены, что содержание сообщения не изменено в течение передачи (целостность сообщения).
3. Клиент и продавец должны быть убеждены, что самозванец не перехватит чувствительную информацию, такую как номер кредитной карточки (конфиденциальность).

Сегодня применяются в основном два протокола обеспечения безопасности на транспортном уровне: *Протокол "Уровень безопасных розеток" (SSL - Secure Socket Layer)* и *Протокол Безопасности Транспортного уровня (TLS - Transport Layer Security)*. Мы сначала обсудим *SSL*, затем *TLS*, а потом их сравним и покажем их отличия друг от друга. [рис. 7.1](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.1) показывает *место* *SSL* и *TLS* в модели *Интернет* (модель протоколов *TCP*/IP).



**Рис. 7.1.**Место SSL и TSL в модели Интернет

Одна из целей этих протоколов состоит в том, чтобы обеспечить *сервер* и клиента услугами установления подлинности, конфиденциальности и целостности данных. *Прикладной уровень* программ клиент-*сервер* (*client-server*), таких как *Язык передачи гипертекста (HTTP)*, который использует услуги *TCP*, может инкапсулировать свои данные в пакеты *SSL*. Если *сервер* и клиент согласованы с функционирующими программами *SSL* (или *TLS*), то клиент может использовать *URL* https: // ... вместо http:// ..., для того чтобы разрешить сообщениям *HTTP* инкапсулироваться в пакеты *SSL* (или *TLS*). Например, номера кредитной карточки могут быть безопасно переданы через *Интернет* для *онлайн*-покупателей.

**7.1. SSL-архитектура**

*SSL* разработан, чтобы обеспечить *безопасность* и услуги сжатия данным, сгенерированным прикладным уровнем. Как правило, *SSL* может получить данные от любого протокола прикладного уровня, но обычно он получает их от *протокола HTTP*. Данные, полученные от приложения, сжаты (дополнительно), подписаны и зашифрованы, а затем их передают к протоколу транспортного уровня, такому как *TCP*. *Фирма* Netscape разработала *SSL* в 1994 году . Версии 2 и 3 были выпущены в 1995 году. В этой лекции мы рассмотрим только *SSL* V. 3.

**Услуги**

SSL обеспечивает несколько услуг для данных, полученных от прикладного уровня.

**Фрагментация**

Сначала SSL делит данные на блоки 214 байтов или меньше.

**Сжатие**

Каждый фрагмент данных сжат с использованием одного из методов сжатия без потери методом, согласованным по договору между клиентом и сервером. Эта услуга является дополнительной.

**Целостность сообщения**

Чтобы сохранять целостность данных, SSL использует ключевую хэш-функцию для создания кода проверки подлинности (MAC).

**Конфиденциальность**

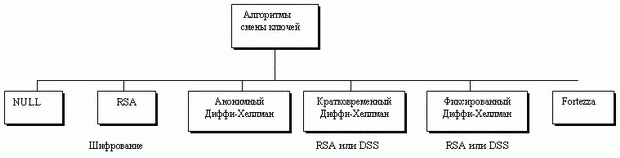
Чтобы обеспечить конфиденциальность, первоначальные данные и код проверки подлинности (MAC) зашифрованы, с использованием криптографии с симметричными ключами.

**Организация кадра**

К зашифрованной полезной нагрузке добавляется заголовок. Полезную нагрузку затем передают достоверному протоколу транспортного уровня.

**Алгоритмы смены ключей**

Как мы увидим позднее, для обмена подлинными и конфиденциальными сообщениями клиенту и серверу нужны шесть криптографических объектов секретности (четыре ключа и два вектора инициализации). Однако чтобы создать их, между этими двумя сторонами должен быть установлен один *предварительный главный секретный код* (pre-master secret). SSL определяет шесть методов обмена ключами, чтобы установить этот предварительный объект секретности: NULL, RSA, *анонимный Диффи-Хеллман* (Diffie-Hellman), *кратковременный Диффи-Хеллман*, *фиксированный Диффи-Хеллман* и *Fortezza*, как это показано на [рис. 7.2](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.2).

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_02.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_02.gif)  
**Рис. 7.2.**Методы замены ключей

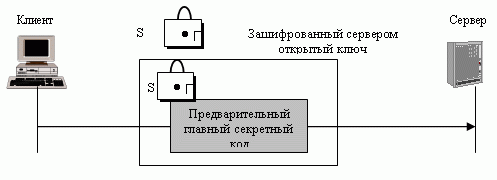
**NULL (ПУСТОЙ УКАЗАТЕЛЬ)**

В этом методе нет никакой смены ключей. Между клиентом и сервером не установлен *предварительный главный секретный код*.

**И клиент и сервер должны знать значение***предварительного главного секретного кода*.

**RSA**

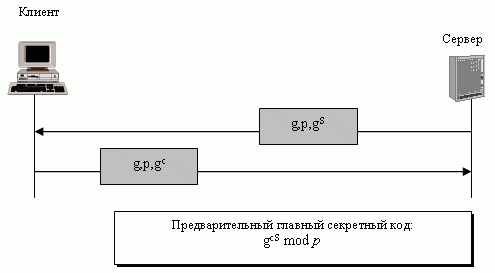
В этом методе предварительный главный секретный код - 48-байтовое случайное число, созданное клиентом, зашифрованное открытым ключом RSА-сервера и передаваемое серверу. Сервер должен передать свой сертификат шифрования/дешифрования RSA. [рис. 7.3](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.3) иллюстрирует идею.



**Рис. 7.3.**RSA-смена ключа; открытый ключ сервера

*Анонимный протокол Диффи-Хеллмана*

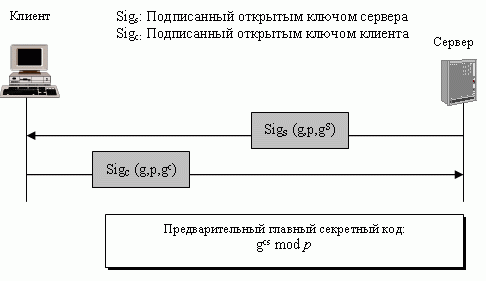
Это самый простой и наиболее ненадежный метод. *Предварительный главный секретный код* устанавливают между клиентом и сервером, используя протокол Диффи-Хеллмана. При этом передают половину ключа в исходном тексте - это называется *анонимным протоколом Диффи-Хеллмана*, потому что ни одна сторона не известна другой. Как мы уже обсуждали, самый серьезный недостаток этого метода - возможность атаки "посредника". [рис. 7.4](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.4) иллюстрирует идею анонимного метода.



**Рис. 7.4.**Анонимный протокол Диффи-Хеллмана смены ключей

*Кратковременный метод Диффи-Хеллмана*

Чтобы сорвать атаку "посредника", может быть использована *кратковременная смена ключей методом Диффи-Хеллмана*. Каждая сторона передает ключ Диффи-Хеллмана, подписанный своим секретным ключом. На приемной стороне должны проверить подпись, используя открытый ключ передатчика. Обмен открытыми ключами для проверки использует либо RSA-, либо DSS-сертификат цифровой подписи. [рис. 7.5](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.5) иллюстрирует идею.



**Рис. 7.5.**Кратковременный протокол Диффи-Хеллмана смены ключей

*Фиксированный метод Диффи-Хеллмана*

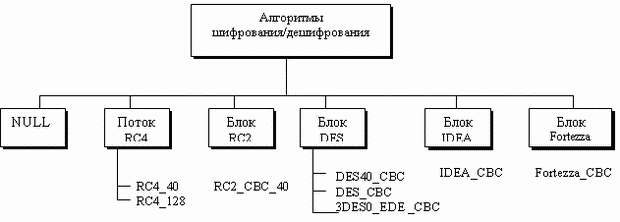
Другое решение - *фиксированный метод Диффи-Хеллмана*. Все объекты в группе могут подготовить фиксированные параметры ( g и p). Затем каждый объект может создать фиксированную половину ключа ( gx). Для дополнительной безопасности каждая отдельная половина ключа Диффи-Хеллмана вставляется в сертификат, проверенный центром сертификации (CA). Другими словами, две стороны отдельно не обмениваются полуключами; CA передает полуключи в специальном сертификате RSA или DSS. Когда клиент должен вычислить *предварительный главный секретный код*, он использует свой собственный фиксированный полуключ и полуключ сервера, полученный в сертификате. Сервер делает то же самое, но в обратном порядке. Обратите внимание, что в этом методе не передаются сообщения смены ключей, а происходит только обмен сертификатами.

*Fortezza*

*Fortezza* (образован из итальянского слова "крепость") - зарегистрированная торговая марка американского Агентства Национальной безопасности (*NSA* - *National Security* *Agency*). Это семейство протоколов безопасности, разработанных для Отдела Защиты. Мы здесь не обсуждаем *Fortezza* из-за его сложности.

**Алгоритмы шифрования/дешифрования**

Есть несколько возможностей выбора алгоритма шифрования/дешифрования. Мы можем разделить алгоритмы на 6 групп, как это показано на [рис. 7.6](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.6). Все протоколы блока используют 8-байтовый вектор инициализации (IV), кроме *Fortezza*, который применяет 20 байтов IV.

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_06.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_06.gif)  
**Рис. 7.6.**Алгоритмы шифрования/дешифрования

**NULL**

NULL - категория, которая просто определяет отсутствие алгоритма шифрации/дешифрации.

**Поток RC**

В режиме потока RC определены два потока алгоритма: RC4-40 (ключ на 40 битов) и RC4-128 (ключ на 128 битов).

**Блок RS**

В режиме блока RC определен один алгоритм: RC2\_CBC\_40 (ключ на 40 битов).

*CBC* (Cipher Block Chaining) - сцепление шифрованных блоков.

**DES**

Все алгоритмы DES определены в режиме блока. DES40\_CBC использует ключ на 40 битов. Стандартные DES определены как DES\_CBC. 3DES\_EDE\_CBC используют ключ на 168 битов.

**IDEA**

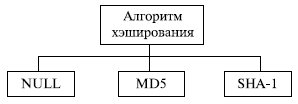
В режиме блока IDEA определен один алгоритм - IDEA\_CBC, с ключом на 128 битов.

**Fortezza**

В режиме блока *Fortezza* определен один алгоритм - FORTEZZA\_CBC, с ключом на 96 бит.

**Алгоритмы хэширования**

SSL использует *алгоритмы хэширования*, чтобы обеспечить целостность сообщения (установление подлинности сообщения). Имеются хэш-функции, показанные на [рис. 7.7](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#image.7.7).



**Рис. 7.7.**

Null **(Пустой указатель)**

Две стороны могут отказаться использовать *алгоритм хэширования*. В этом случае сообщение не заверено.

**MD5**

Две стороны могут выбрать MD5 как *алгоритм хэширования*. В этом случае используется *алгоритм хэширования* MD5 - 128-битовый.

**SHA-1**

Две стороны могут выбрать SHA как *алгоритм хэширования*. В этом случае используется *алгоритм хэширования* SHA-1 на 160 битов.

**Набор шифров**

Комбинация смены ключей, хэширования и алгоритмов шифрования определяет *набор шифров* для каждого сеанса SSL. [табл. 7.1](https://intuit.ru/studies/courses/553/409/lecture/9387?page=1#table.7.1) показывает наборы, применяемые в Соединенных Штатах. Мы не включили наборы, которые используются для экспорта. Обратите внимание, что в списке находятся не все комбинации смены ключей, целостности сообщения и установления подлинности сообщения.

Каждый набор начинается термином "SSL", сопровождаемым алгоритмом смены ключей. Слово "WITH" отделяет алгоритм смены ключей от алгоритмов шифрования и хэширования.

Например,

SSL\_DHE\_RSA\_WITH\_DES\_CBC\_SHA

определяет DHE\_RSA (кратковременный метод Диффи-Хеллмана ( Diffie-Hellman ephemeral)) с цифровой подписью RSА для смены ключей, DES\_CBC - в качестве алгоритма шифрования и SHA - как *алгоритм хэширования*.

Обратите внимание, что сокращения DH (Diffie-Hellman) - это фиксированный метод Диффи-Хеллмана, DHE (Diffie-Hellman Ephemeral) - это кратковременный метод Диффи-Хеллмана и DH-anon (anonymous Diffie-Hellman) - это *анонимный метод Диффи-Хеллмана*.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 7.1. Список набора шифров SSL | | | |
| ***Набор шифров*** | **Смена ключей** | **Шифрование** | **Хэш** |
| SSL-NULL-WITH-NULL- NULL  SSL\_RSA\_WITH\_NULL\_MD5  SSL\_RSA\_WITH\_NULL\_SHA  SSL\_RSA\_WITH\_RC4\_128\_MD5  SSL\_RSA\_WITH\_RC4\_128\_SHA  SSL\_RSA\_WITH\_IDEA\_CBC\_SHA  SSL\_RSA\_WITH\_DES\_CBC\_SHA  SSL RSA WITH 3DES EDE CBC SHA | NULL  RSA  RSA  RSA  RSA  RSA  RSA  RSA | NULL  NULL  NULL  RC4  RC4  IDEA  DES  3DES | NULL  MD5  SHA-1  MD5  SHA-1 SHA-1  SHA-1  SHA-1 |
| SSL\_ DH\_anon\_ WITH\_ RC4\_128\_ MD5 | DH anon | RC4 | MD5 |
| SSL\_ DH\_ anon WITH\_ DES\_ CBC\_ SHA | DH anon | DES | SHA-1 |
| SSL \_DH\_ anon\_ WITH\_ 3DES\_ EDE\_ CBC SHA | DH anon | 3DES | SHA-1 |
| SSL\_ DHE\_ RSA\_ WITH\_ DES\_ CBC\_ SHA | DHE RSA | DES | SHA-1 |
| SSL\_ DHE\_RSA\_ WITH\_3DES\_ EDE CBC SHA | DHE RSA | 3DES | SHA-1 |
| SSL\_DHE\_DSS\_ WITH\_ DES\_ CBC\_ SHA | DHE DSS | DES | SHA-1 |
| SSL\_ DHE\_ DSS\_ WITH\_ 3DES\_ EDE\_ CBC\_ SHA | DHE DSS | 3DES | SHA-1 |
| SSL\_ DH\_ RSA\_ WITH\_ DES\_ CBC\_ SHA | DH RSA | DES | SHA-1 |
| SSL\_ DH\_ RSA\_ WITH\_ 3DES\_ EDE\_ CBC\_ SHA | DH RSA | 3DES | SHA-1 |
| SSL\_ DH\_ DSS\_ WITH\_ DES\_ CBC\_ SHA | DH DSS | DES | SHA-1 |
| SSL\_ DH\_ DSS\_ WITH\_ 3DES\_ EDE\_ CBC\_ SHA | DH DSS | 3DES | SHA-1 |
| SSL\_FORTEZZA\_DMS\_WITH\_NULL\_SHA  SSL\_FORTEZZA\_DMS\_WITH\_FORTEZZA\_CBC\_SHA  SSL\_FORTEZZA\_DMS\_WITH\_RC4\_128\_SHA | Fortezza  Fortezza  Fortezza | NULL  Fortezza  RC4 | SHA-1  SHA-1  SHA-1 |

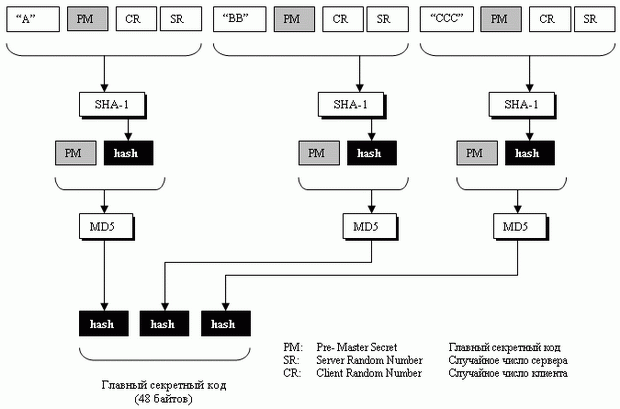
**Алгоритмы сжатия**

Как мы уже говорили, сжатие является дополнительной услугой в SSLv3. Для SSLv3 не определен алгоритм сжатия. Поэтому заданным по умолчанию методом сжатия служит NULL. Однако система может использовать любой алгоритм сжатия по выбору сторон.

**Генерирование криптографических параметров**

Чтобы обеспечить целостность и конфиденциальность сообщения, в SSL необходимо иметь: шесть криптографических объектов секретности, четыре ключа и два инициализирующих вектора (IV). Клиенту нужно: один ключ для передачи сообщения установления подлинности (HMAC - HASH-BASED *MESSAGE AUTHENTICATION* CODE), один ключ для шифрования и один IV для шифрования блока. Сервер нуждается в том же самом. SSL требует, чтобы ключи для одного направления отличались от ключей для другого направления. Если будет атака в одном направлении, она не затронет другое направление. Для генерации параметров используют следующую процедуру:

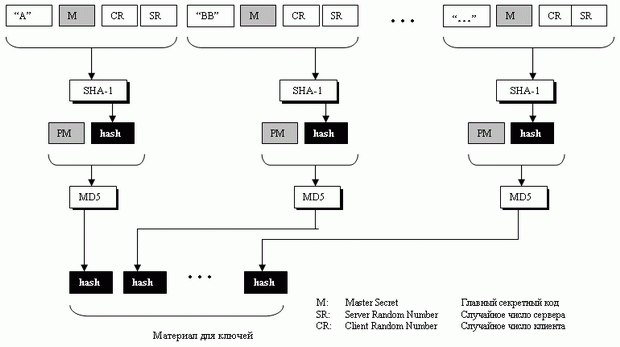
1. Клиент и сервер обмениваются двумя случайными числами, одно из которых создано клиентом, а другое - сервером.
2. Клиент и сервер обмениваются одним *предварительным главным секретным кодом*, используя один из алгоритмов смены ключей, которые мы обсуждали раньше.
3. Создается 48-байтовый *главный секретный код* **(master secret)** из *предварительного главного секретного кода* **(pre-master secret)**, с применением *хэш-функций (SHA*-1 и MD5), как это показано на рис. 7.8.

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_08.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_08.gif)  
**Рис. 7.8.**Вычисление главного секретного кода из предварительного главного секретного кода

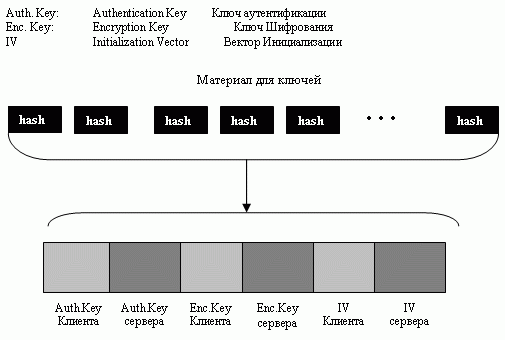
1. *Главный секретный код* используется для того, чтобы создать *материал для ключей* **(key material)**, который имеет переменную длину. Для этого применяют то же самое множество хэш-функций, что и в предыдущем случае, и подставляют спереди различные константы, как это показано на [рис. 7.9](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#image.7.9). Алгоритм повторяется, пока не получится материал для ключа адекватного размера.

Обратите внимание, что длина блока *материала для ключей* зависит от выбранного набора шифра и размера ключей, необходимых для этого набора.

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_09.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_09.gif)  
**Рис. 7.9.**Вычисление материала для ключей из главного секретного кода

1. Из *материала для ключей* извлекаются шесть различных ключей, как показано на [рис. 7.10](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#image.7.10).



**Рис. 7.10.**Извлечение криптографических секретных кодов из материала

**Сеансы и соединение**

SSL отличает *соединение* от **сеанса.**Давайте рассмотрим эти два термина. Сеанс - связь между клиентом и сервером. После того как сеанс установлен, эти две стороны имеют общую информацию, такую как идентификатор сеанса, сертификат, подтверждающий подлинность каждого из них (в случае необходимости), метод сжатия (если необходимо), *набор шифров* и *главный секретный код*. Эта информация используется для того, чтобы создать ключи для сообщения, содержащего шифр установления подлинности.

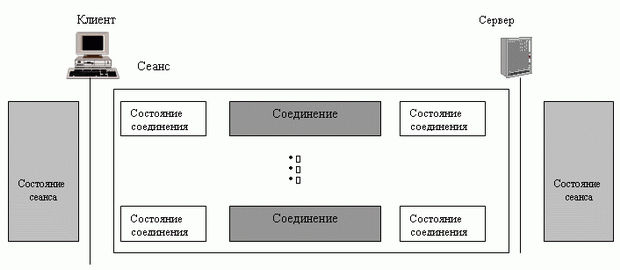
Для двух объектов, чтобы начать обмен данными, установление сеанса необходимо, но не достаточно; они должны создать между собой соединение. Эти два объекта обмениваются двумя случайными числами и создают, используя *главный секретный код*, ключи и параметры, необходимые для того, чтобы обмениваться сообщениями, включая установление подлинности и секретность.

Сеанс может состоять из многих соединений. Соединение между двумя сторонами может быть закончено и восстановлено в пределах одного и того же сеанса. Когда соединение закончено, эти две стороны могут также закончить сеанс, но это необязательно. Сеанс может быть приостановлен и продолжен снова.

Чтобы создавать новый сеанс, эти две стороны должны пройти процесс переговоров. Чтобы возобновлять старый сеанс и создавать только новое соединение, эти две стороны могут пропустить часть переговоров, что уменьшает время вхождения в связь. Не надо создавать *главный секретный код*, когда сеанс продолжается.

Разделение сеанса от соединения предотвращает высокую стоимость создания *главного секретного кода*. Если мы разрешаем приостановления и продолжения сеанса, процесс вычисления *главного секретного кода* может быть устранен. [рис. 7.11](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#image.7.11) иллюстрирует идею сеанса и соединения в этом сеансе.

**В сеансе одна сторона играет роль клиента и другая - роль сервера. При соединении обе стороны имеют равные роли, они равны по уровню.**

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_11.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_11.gif)  
**Рис. 7.11.**Сеанс и соединение

**Состояние сеанса**

Сеанс определяется состоянием сеанса - это множество параметров, установленных между сервером и клиентом. [табл. 7.2](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#table.7.2) показывает список параметров для *состояния сеанса*.

|  |  |
| --- | --- |
| Таблица 7.2. Параметры *состояния сеанса* | |
| **Параметры** | **Описание** |
| ID сеанса | Случайное 8-битовое число, выбранное сервером и определяющее сеанс |
| Сертификат уровня | Сертификат типа X509 .v.3. Этот параметр может быть пустым (null) |
| Метод сжатия | Метод сжатия |
| *Набор шифров* | Согласованный *набор шифров* |
| *Главный секретный код* | 48-байтовый секретный код |
| Возможность повторения | Флаг "Да, Нет", который разрешает новое соединение в старом сеансе |

Состояние соединения

*Подключение* определяется *состоянием соединения* - это множество параметров, установленных между двумя равными по уровню объектами. [табл. 7.3](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#table.7.3) показывает список параметров для состояния соединения.

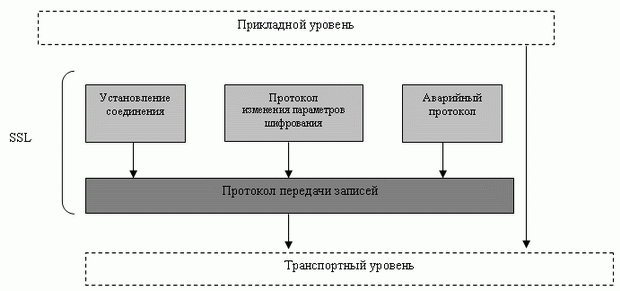
SSL использует два признака, чтобы отличить криптографическую секретность: *писать* и *читать.* Термин *писать* определяет ключ, используемый для того, чтобы подписать или зашифровать исходящее сообщение.Термин *читать* определяет ключ, используемый для того, чтобы подтвердить или расшифровывать прибывающие сообщения. Обратите внимание: *писать-*ключ клиента - тот же самый, что и ключ- *читать*сервера; ключ- *читать*клиента - тот же самый, что и ключ- *писать* сервера.

**Клиент и сервер имеют шесть различных криптографических объекта: три объекта секретности читать и три писать. Секретность читать для клиента та же самая, что и секретность писать для сервера, и наоборот.**

|  |  |
| --- | --- |
| Таблица 7.3. Параметры состояния соединения | |
| **Параметры** | **Описание** |
| Случайные числа клиента и сервера | Последовательность байтов, выбранная для каждого соединения серверу и клиенту. |
| Записанный сервером секретный код подлинности сообщения | Ключ *кода установления подлинности сообщения* исходящего сервера для сохранения целостности сообщения.. Используется сервером для подписи, а клиентом для верификации. |
| Записанный клиентом секретный код установления подлинности сообщения | Ключ *кода установления подлинности сообщения* исходящего сервера для сохранения целостности сообщения.. Используется сервером для подписи, а клиентом для верификации. |
| Секретный код, записанный сервером | Ключ шифрования исходящего сервера для сохранения целостности сообщения |
| Секретный код, записанный клиентом | Ключ шифрования исходящего сервера для сохранения целостности сообщения |
| Вектор инициализации | Блочные шифры в режиме "цепочки блочных шифров" I (*CBC*) используют векторы инициализации. (IV). Для каждого *шифровального ключа* путем переговоров определен один вектор инициализации, который используется первым блоком обмена ключами. Зашифрованный текст из блока используется как вектор инициализации ( IV) для следующего блока. |
| Порядковый номер | Каждая сторон имеет порядковый номер. Он начинается с 0 и увеличивается на 1. Он не должен быть больше 264 - 1. |

**7.2. Четыре протокола**

Мы обсудили идею относительно *SSL*, не показав, как *SSL* выполняет свои задачи. *SSL* содержит четыре протокола на двух уровнях, как это изображено на [рис. 7.12](https://intuit.ru/studies/courses/553/409/lecture/9387?page=3#image.7.12). Протокол передачи записей - переносящий информацию. Он переносит на *транспортный уровень* сообщения от трех других протоколов, а также данные, поступающие от прикладного уровня. Сообщения из *протокола записей* - это полезная нагрузка для транспортного уровня, обычно *TCP*. *Протокол установления соединения* обеспечивает параметры безопасности для *Протокола записей*. Он устанавливает *набор шифров* и задает ключи и параметры безопасности.

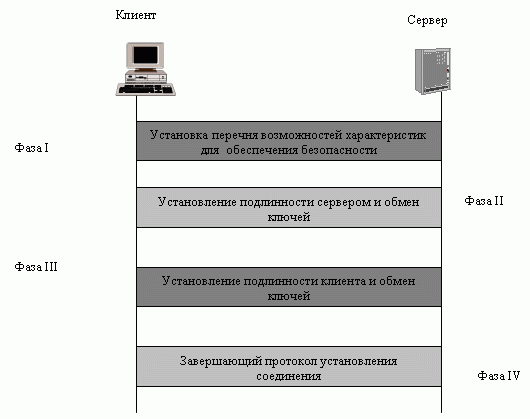
[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_12.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_12.gif)  
**Рис. 7.12.**Четыре протокола SSL

Он также подтверждает, если необходимо, подлинность сервера клиенту и подлинность клиента серверу. Протокол изменения параметров шифрования используется, чтобы передавать сигналы для подготовки к криптографической безопасности. *Аварийный протокол* нужен, чтобы известить о ситуациях, отклоняющихся от нормы. Все эти протоколы мы кратко рассмотрим в этой секции.

**Протокол установления соединения**

*Протокол установления соединения* используется при передаче сообщений, чтобы договориться, если это необходимо, о составе шифров от сервера к клиенту и от клиента к серверу и обменяться информацией, для того чтобы обеспечить криптографическую безопасность. Процедура установления связи происходит в 4 фазы; она проиллюстрирована на рис. 7.13.



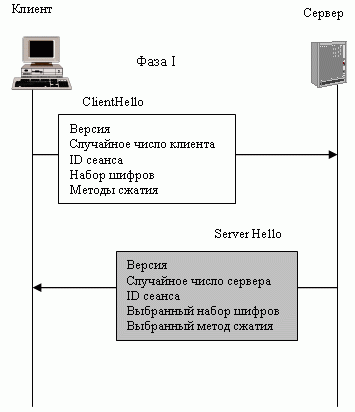
**Рис. 7.13.**Протокол установления соединения

**Фаза 1: Установление характеристик для обеспечения безопасности**

В Фазе 1 клиент и сервер объявляют свои характеристики безопасности, которые нужны и удобны для обоих. В этой фазе устанавливается и выбирается ID сеанса. Стороны согласуют конкретный метод сжатия. Наконец, выбирают два случайных числа: одно выбирается клиентом и другое - сервером, чтобы создать *главный секретный код*. В этой фазе стороны обмениваются двумя сообщениями: ClientHello и ServerHello. [рис. 7.14](https://intuit.ru/studies/courses/553/409/lecture/9387?page=3#image.7.14) содержит дополнительные детали Фазы 1.

**ClientHello.**Клиент посылает сообщение. Оно содержит следующую информацию:

* Самый высокий номер версии SSL, которую может поддержать клиент.
* 32-байтовое случайное число (от клиента), которое будет использоваться для генерации *главного секретного кода* (мастер кода).
* ID сеанса, который определяет сеанс.
* *Набор шифров*, который определяет список алгоритмов, поддерживаемых клиентом.
* Список методов сжатия, которые клиент может поддержать.



**Рис. 7.14.**Протокол установления соединения, Фаза I

**ServerHello.**Сервер отвечает клиенту сообщением ServerHello, оно содержит:

* Номер версии SSL. Это два номера версии:

наиболее высокий номер, поддерживаемый клиентом, и наиболее высокий, поддерживаемый сервером.

* 32-байтовое случайное число (от сервера), которое будет использоваться для генерации *главного секретного кода* (мастер кода).
* ID сеанса, который определяет сеанс.
* Выбранный шифр из списка клиента.
* Выбранный метод сжатия из списка клиента.

**После Фазы 1 клиент и сервер знают следующее: Версия SSL**

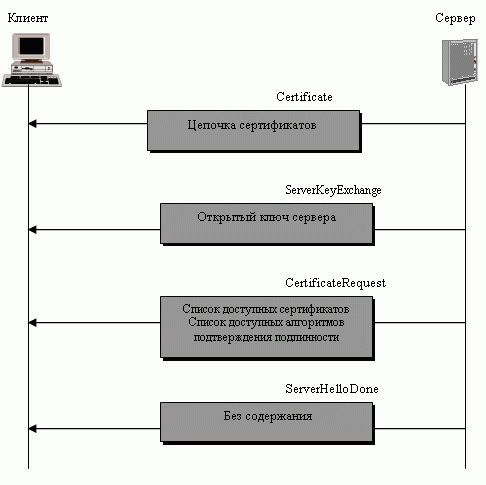
**Алгоритмы для смены ключей, установления подлинности сообщения и шифрования**

**Метод сжатия**

**Два случайных числа для генерации ключей**

**Фаза II: Смена ключей сервера и установление подлинности**

В фазе II сервер, если необходимо, подтверждает свою подлинность. Передатчик может передать свой открытый ключ и может также запросить сертификат клиента. В конце сервер объявляет, что процесс serverHello окончен. [рис. 7.15](https://intuit.ru/studies/courses/553/409/lecture/9387?page=3#image.7.15) дает дополнительные сведения о Фазе II.



**Рис. 7.15.**Протокол установления соединения, Фаза II

**Certificate (сертификат)** - если это требуется, сервер передает сообщение **certificate**, чтобы подтвердить свою подлинность. Сообщение включает список сертификатов типа X.509. Если алгоритм смены ключей - *анонимный Диффи-Хеллман* (Diffie-Hellman), то сертификат не нужен.

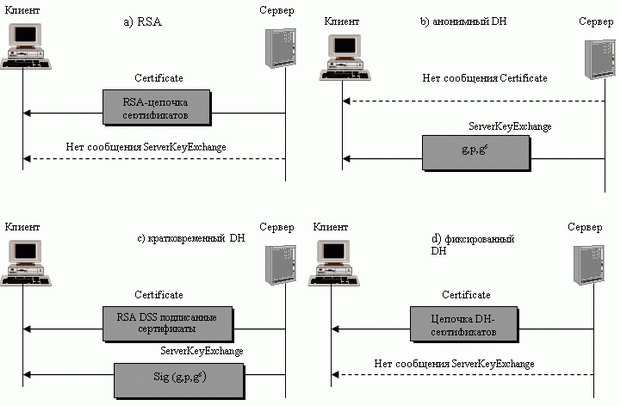
**ServerKeyExchange.** После сообщения **Certificate** сервер передает сообщение ServerKeyExchange, которое включает в себя его вклад в *предварительный главный секретный код*. Если метод смены ключей - RSA или метод "*фиксированный Диффи-Хеллман*", то такого сообщение не требуется.

**CertificateRequest** - сервер может потребовать, чтобы клиент подтвердил свою подлинность. В этом случае сервер передает CertificateRequest сообщение в Фазе II, в котором запрашивает от клиента подтверждение в Фазе III. Сервер не может запросить сертификат от клиента, если клиент использует метод "*анонимный Диффи-Хеллман*".

**ServerHelloDone** - последнее сообщение в Фазе II. Оно является сигналом клиенту, что Фаза II закончена и что клиент должен запустить Фазу III.

**После Фазы II Клиенту подтверждена подлинность сервера. Если требуется, то клиент знает открытый ключ сервера.**

Давайте тщательно рассмотрим установление подлинности сервера и смену ключей в этой фазе. Первые два сообщения здесь базируются на методе смены ключей. [рис. 7.16](https://intuit.ru/studies/courses/553/409/lecture/9387?page=3#image.7.16) показывает четыре из шести методов обмена ключей, которые мы обсуждали раньше. Мы не включили Нулевой (NULL) метод, потому что в нем нет никакого обмена. Мы не включили метод *Fortezza*, потому что в этой книге мы его глубоко не рассматривали.

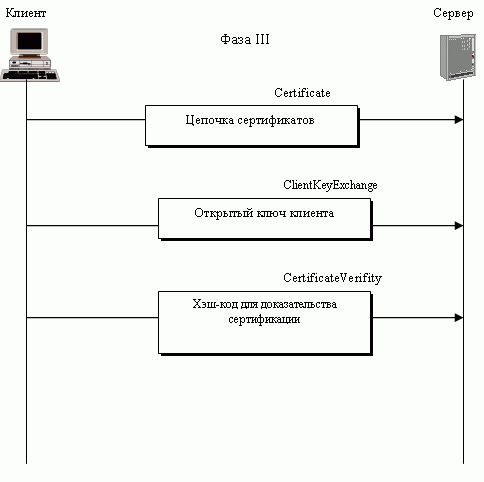
[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_16.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_16.gif)  
**Рис. 7.16.**Четыре случая Фазы II

1. **RSA.** В этом методе в первом сообщении сервер передает сертификат своего открытого ключа RSА шифрования/дешифрования. Второе сообщение, однако, пустое, потому что не сгенерирован предварительный *главный секретный код* - он передается клиентом в следующей фазе. Обратите внимание, что сертификат открытого ключа подтверждает подлинность сервера клиенту. Когда сервер получает предварительный *главный секретный код*, он расшифровывает его секретным ключом. Владение секретным ключом для сервера - доказательство, что сервер - это тот объект, который находился в сертификате открытого ключа, переданном в первом сообщении.
2. **Анонимный DH (DHanonym).** В этом методе не требуется сообщения **Certificate**. Анонимный объект не имеет сертификата. В ServerKeyExchange сообщении сервер передает параметры метода Диффи-Хеллмана и свой полуключ. Обратите внимание, что здесь сервер не подтверждает свою подлинность.
3. **Кратковременный DH (DHE).** В этом методе сервер передает либо RSA-, либо DSS-сертификат цифровой подписи. Секретный ключ, связанный с сертификатом, позволяет серверу подписать сообщение; открытый ключ позволяет получателю проверить подпись. Во втором сообщении сервер передает параметры Диффи-Хеллмана и полуключ, подписанный его секретным ключом. В этом методе сервер подтверждает клиенту свою подлинность не потому, что он передает сертификат, а потому, что подписывает параметры и ключи своим секретным ключом. Владение секретным ключом - доказательство того, что сервер - объект, который находится в сертификате. Если самозванец копирует и передает сертификат клиенту, симулируя, что он - сервер, запрошенный в сертификате, он не сможет подписать второе сообщение, потому что не имеет секретного ключа.
4. **Фиксированное DH (DH).** В этом методе сервер передает либо RSA-, либо DSS-сертификат цифровой подписи, в который включает свой зарегистрированный полуключ DH. Второе сообщение - пустое.
5. Сертификат подписан секретным ключом CA и может быть проверен клиентом, использующим открытый ключ CA. Другими словами, CA подтвердил клиенту подлинность и заверил, что полуключ принадлежит серверу.

**Фаза III: Смена ключей клиента и установление его подлинности**

Фаза III предназначена подтвердить подлинность клиента. От клиента серверу можно передать до трех сообщений, как показано на [рис. 7.17](https://intuit.ru/studies/courses/553/409/lecture/9387?page=4#image.7.17).



**Рис. 7.17.**Протокол установления соединения, Фаза III

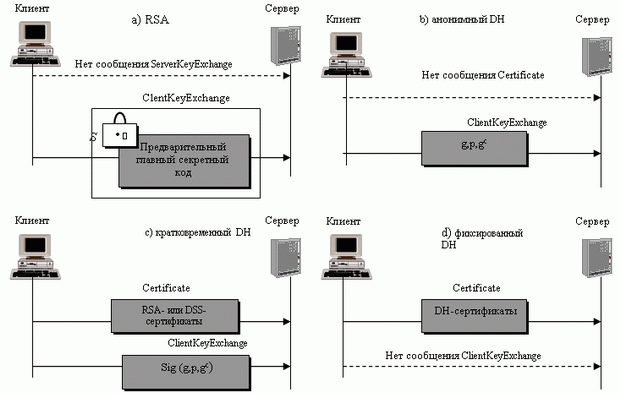
**Сертификат.** Чтобы сертифицировать себя на сервере, клиент передает сообщение Certificate. Обратите внимание, что его формат - тот же самый, что и у сообщения Certificate, передаваемого сервером в Фазе II, но содержание различно. В данном случае Certificate включает цепочку сертификатов, которые сертифицируют клиента. Такое сообщение передают, только если сервер запросил сертификат в Фазе II. Если есть запрос и клиент не имеет сертификата, чтобы передать его, тогда передается аварийное сообщение (часть *аварийного протокола*, который будет обсуждаться позже) с предупреждением, что сертификата нет. Сервер может продолжить сеанс или может решить прервать его.

**ClientKeyExchange.** После передачи сообщения Certificate клиент передает сообщение ClientKeyExchange, которое включает в себя вклад в предварительный *главный секретный код*. Содержание этого сообщения базируется на используемом алгоритме смены ключей. Если метод - RSA, клиент создает полный предварительный *главный секретный код* и зашифровывает его открытым ключом RSA сервера. Если метод - *анонимный* или *кратковременный Диффи-Хеллман*, клиент передает свой полуключ. Если метод - *Fortezza*, клиент передает параметры *Fortezza*. Содержание этого сообщения пусто, если метод - "*фиксированный Диффи-Хеллман*".

**Верификация сертификата.** Если клиент передал сертификат, объявляющий, что он имеет открытый ключ в сертификате, он должен доказать, что он знает соответствующий секретный ключ. Это необходимо, чтобы сорвать попытки самозванца, который передает сертификат и утверждает, что он исходит от клиента. Доказательство владения секретным ключом он представляет, создавая сообщение и подписывая его секретным ключом. Сервер может проверить сообщение переданным ему открытым ключом, чтобы гарантировать, что сертификат фактически принадлежит клиенту. Обратите внимание, что это возможно, если в сертификат включены необходимые подписанные полномочия; включая пару ключей - открытый и секретный. Сертификат для *фиксированного Диффи-Хеллмана* не может быть проверен таким путем.

**После Фазы III Клиент проверен на подлинность сервером. Клиент и сервер знают***предварительный главный секретный код*.

Рассмотрим более подробно установление подлинности клиента и смену ключей в этой фазе. Основой данного метода здесь являются три сообщения. [рис. 7.18](https://intuit.ru/studies/courses/553/409/lecture/9387?page=4#image.7.18) показывает четыре из шести методов, которые мы рассматривали раньше. Опять мы исключили метод NULL и метод *Fortezza*.

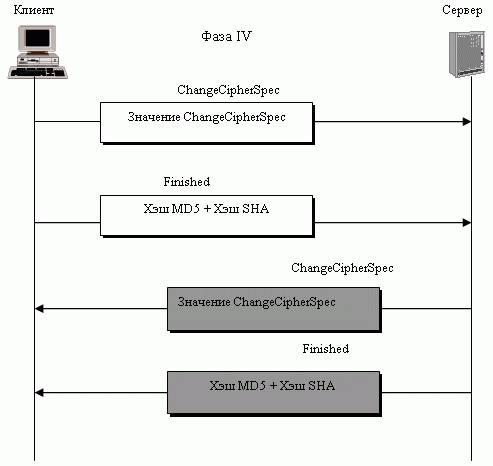
[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_18.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_18.gif)  
**Рис. 7.18.**Четыре случая Фазы III

1. **RSA.** В этом случае не передается сообщение Certificate, если сервер явно не запросил его в Фазе II. ClientKeyExchange включает в себя *предварительный главный секретный код*, который зашифрован открытым ключом RSА, полученным в Фазе II.
2. **Анонимный DH.** В этом методе не передается сообщение Certificate. Сервер не имеет права запросить сертификат в Фазе II, потому что и клиент и сервер - анонимны. В ClientKeyExchange-сообщении сервер передает параметры Диффи-Хеллмана и свой полуключ. Обратите внимание, что в этом методе клиент не проверен на подлинность сервером.
3. **Кратковременный DH.** В этом методе клиент обычно имеет сертификат. Сервер должен передать его RSA- или DSS-сертификат (на основе согласованного множества шифров). В ClientKeyExchange сообщении клиент подписывает параметры **DH**, а также и свой полуключ, и передает их. Подлинность клиента подтверждается серверу подписью второго сообщения. Если клиент не имеет сертификата, а сервер запрашивает его, клиент передает аварийное сообщение. Если это приемлемо для сервера, клиент передает параметры DH и ключ в исходном тексте. Конечно, в этой ситуации клиент не проверен сервером на подлинность.
4. **Фиксированный DH.** В этом методе клиент обычно передает сертификат **DH** в первом сообщении. Обратите внимание, что здесь второе сообщение пустое. Клиент подтверждает свою подлинность серверу, посылая сертификат **DH**.

**Фаза IV: Завершение и окончание**

В Фазе IV клиент и сервер передают сообщения, чтобы изменить спецификацию шифра и закончить процедуру установления связи. В этой фазе происходит обмен четырьмя сообщениями, как это показано на [рис. 7.19](https://intuit.ru/studies/courses/553/409/lecture/9387?page=4#image.7.19).



**Рис. 7.19.**Протокол установления соединения, Фаза IV

**ChangeCipherSpec.** Клиент передает сообщение ChangeCipherSpec, чтобы показать, что он передал весь *набор шифров* и параметры для перехода из состояния ожидания в активное состояние. Это сообщение - фактически часть *протокола ChangeCipherSpec*, который мы обсудим позже.

**Finished**. Это сообщение также передает клиент. Сообщение Finished объявляет об окончании процедуры установления связи клиентом.

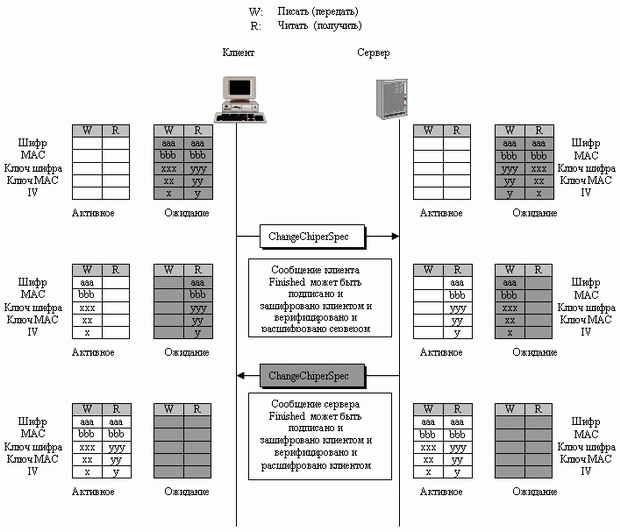
**ChangeCipherSpec.** Сервер передает ChangeCipherSpec-сообщение, чтобы показать, что он также обменялся *набором шифров* и параметрами для перехода из состояния ожидания в активное состояние. Это сообщение - часть *протокола ChangeCipherSpec*, который будет рассмотрен позже.

**Finished.** Наконец, сервер передает сообщение Finished, чтобы показать, что процедура установления связи полностью закончена.

**Протокол ChangeCipherSpec**

Мы видели, что переговоры о *наборе шифров* и генерация криптографической секретности формируются постепенно по ходу выполнения *протокола установления соединения*. Возникает вопрос: когда две стороны могут использовать эти параметры секретности? SSL утверждает, что стороны не могут использовать эти параметры или секретность, пока они не передали или не получили специальное сообщение: ChangeCipherSpec - сообщение, которым обмениваются клиент и сервер в ходе выполнения *протокола установления соединения* и которое определено в *протоколе ChangeCipherSpec*. По этой причине объекты не посылают или не получают сообщение. Передатчик и приемник нуждаются не в одном, а в двух состояниях. Одно - состояние ожидания, при котором сохраняются параметры и секретности. Другое - активное состояние, при котором параметры и секретность используются *протоколом передачи записей*, чтобы подписаться/проверить или зашифровать/расшифровывать сообщения. Кроме того, каждое состояние содержит два множества значений: *читать* (входящее), *писать* (исходящее).

*Протокол ChangeCipherSpec* определяет процесс перемещения значений между состоянием ожидания и активным состоянием. [рис. 7.20](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#image.7.20) показывает гипотетическую ситуацию, с гипотетическими значениями, чтобы только проиллюстрировать концепцию.

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_20.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_20.gif)  
**Рис. 7.20.**Передвижение параметров из состояния ожидания в активное состояние

На рисунке показано только несколько параметров. Перед обменом сообщениями ChangeCipherSpec в состоянии ожидания заполнены только значения столбцов.

Сначала клиент передает ChangeCipherSpec-сообщение. После этого клиент перемещает параметры "писать" (исходящие) из состояния ожидания в активное состояние. Теперь он может использовать эти параметры, чтобы подписаться или зашифровать исходящее сообщение. После получения этого сообщения сервер перемещает параметры "читать" (входящие) из состояния ожидания в активное состояние. Теперь сервер может верифицировать и расшифровывать сообщение. Это означает, что сообщение Finished, передаваемое клиентом, может быть подписано клиентом и расшифровано сервером.

Сервер передает сообщение ChangeCipherSpec после получения от клиента сообщения Finished. После передачи этого сообщения он перемещает параметры "писать" (входящие) из состояния ожидания в активное состояние. Сервер может теперь использовать эти параметры для того, чтобы подписать или зашифровать исходящие сообщения. После того как клиент получает это сообщение, он перемещает первый абзац (входящий) из состояний ожидания в активное состояние. Теперь клиент может проверить (верифицировать) и расшифровать сообщения.

Конечно, после обмена сообщениями Finished обе стороны могут работать в обоих направлениях, используя активные параметры для чтения/записи.

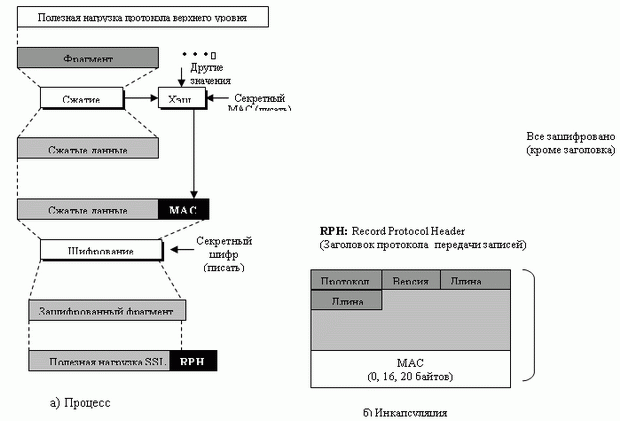
**Аварийный протокол**

SSL использует *аварийный протокол* для того, чтобы известить об ошибках и ненормальном состоянии устройств. Имеется только один тип аварийного сообщения, которое описывает проблему и ее уровень (опасное или полный выход из строя). [табл. 7.4](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#table.7.4) показывает типы аварийных сообщений, определенных для SSL.

|  |  |  |
| --- | --- | --- |
| Таблица 7.4. Аварийные сообщения, определенные для SSL | | |
| ***Цифровое обозначение*** | ***Описание*** | ***Значение*** |
| 0 | *CloseNotify* | Передатчик больше не будет посылать сообщений |
| 10 | *UnexpectedMessage* | Получено несоответствующее сообщение |
| 20 | *BadRecordMAC* | Получено некорректное MAC |
| 30 | *DecompressionFailure* | Невозможно получить несжатый текст |
| 40 | *HandshakeFailure* | Передатчик не может закончить установление соединения |
| 41 | *NoCertificate* | Клиент не сертифицировал послание |
| 42 | *BadCertificate* | Полученный сертификат искажен |
| 43 | *UnsupportedCertificate* | Тип полученного сертификата не поддерживается |
| 44 | *CertificateRevoked* | Подписавший аннулирует сертификат |
| 45 | *CertificateExpired* | Сертификат просрочен |
| 46 | *CertificateUnknown* | Сертификат неизвестен |
| 47 | *Illegal Parameter* | Поле не может быть обработано |

**Протокол передачи записей**

*Протокол передачи записей* доставляет сообщения от верхнего уровня (*протокол установления соединения*, *ChangeCipherSpec-протокол*, *аварийный протокол*) или прикладных уровней. Сообщения фрагментированы и произвольно сжаты; MAC добавляется к сжатому сообщению, используя согласованный алгоритм хэш. Сжатый фрагмент и MAC зашифрованы с применением согласованного алгоритма шифрования. В конце к зашифрованному сообщению добавляется заголовок SSL. [рис. 7.21](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#image.7.21) иллюстрирует этот процесс в передатчике. Процесс в приемнике имеет обратный порядок.

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_21.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_21.gif)  
**Рис. 7.21.**Процесс работы протокола передачи записей

Обратите внимание, однако, что этот процесс может быть выполнен, только когда криптографические параметры находятся в активном состоянии. Сообщения, передаваемые перед перемещением из состояния ожидания в активное состояние, не подписаны и не зашифрованы.

В следующих секциях мы увидим некоторые другие сообщения в *протоколе установления соединения*, которые используются некоторыми определенными типами хэширования для обеспечения целостности сообщения.

**Фрагментация/Объединение**

В передатчике сообщение от прикладного уровня фрагментируется в блоки по 2 байта, при этом последний блок может быть меньше. В приемнике фрагменты объединяются вместе, чтобы создать точную копию первоначального сообщения.

**Сжатие/Расширение**

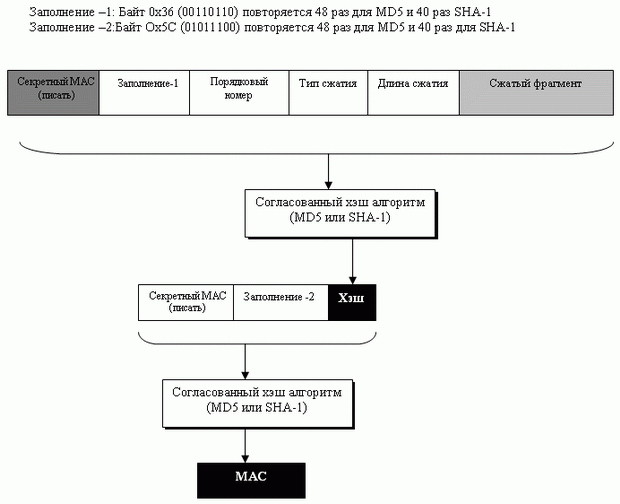
В передатчике все фрагменты прикладного уровня сжаты с использованием метода сжатия, о котором договариваются во время процедуры установления связи. Метод сжатия не должен вносить потери (расширенный фрагмент должен быть точной копией первоначального фрагмента). Размер фрагмента не должен превысить 1024 байта. Некоторые методы сжатия работают только с заранее заданными размерами блоков, и если размер блока меньше, то блок дополняется. Поэтому размер сжатого фрагмента может быть больше, чем размер первоначального фрагмента. В приемнике сжатый фрагмент расширяется, чтобы создать точную копию оригинала. Если размер расширенного фрагмента превышает 214, вырабатывается аварийное сообщение "неправильное расширение" (fatal *decompression*). Обратите внимание, что сжатие/расширение в SSL - дополнительные функции.

**Подписание/Подтверждение**

В передатчике метод подтверждения подлинности определяется в течение установления соединения (NULL, MD5 или SHA-1) и создает подпись (MAC), как это показано на [рис. 7.22](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#image.7.22).

*Алгоритм хэширования* применяется дважды. Сначала хэширование создается путем последовательных соединений (конкатенации) следующих значений:

* секретный MAC (писать) (ключ подтверждения подлинности для исходящих сообщений);
* заполнение 1, которое представляет байт 0x36, повторяемый 48 раз для MD5 и 40 раз для SHA-1;
* порядковый номер этого сообщения;

[](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_22.gif)

[увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_22.gif)  
**Рис. 7.22.**Вычисление MAC

* тип сжатия, который определяет протокол верхнего уровня, обеспечившего сжатие фрагмента;
* длина сжатия, которая указывает длину сжатого фрагмента;
* сжатый фрагмент.

Второй раз завершающее хэширование (MAC) создается последовательным соединением (конкатенацией) следующих значений:

* секретный MAC (писать);
* заполнение 2, которое представляет байт 0x5C, повторяемый 48 раз для MD5 и 40 раз для SHA-1;
* создание хэша из результата первого шага.

В приемнике проводится верификация - вычисление нового хэша и сравнение его с полученным хэшированием.

**Шифрование/Дешифрование**

В передатчике сжатый фрагмент и хэш зашифрованы с применением секретного шифра (писать). В приемнике полученное сообщение расшифровывается с использованием секретного шифра (читать). При шифровании к блоку добавляется заполнение, чтобы сделать размер сообщения пригодным для шифрования - кратным числом размера блока.

**Создание кадра**

В передатчике после шифрования добавляется заголовок *протокола передачи записей*. Заголовок удаляется в приемнике перед дешифрованием.

**7.3. Форматы сообщения SSL**

Мы уже знаем, что сообщения из трех протоколов и данные от прикладного уровня инкапсулируются (вставляются) в сообщения *протокола передачи записей*. Другими словами, сообщение в *протокол передачи записей* инкапсулируется из четырех различных источников на стороне передатчика. На стороне приемника *протокол передачи записей* извлекает сообщения и доставляет их различным пунктам назначения. *Протокол передачи записей* имеет общий заголовок, который добавляется к каждому сообщению, прибывающему от источников, как показано на [рис. 7.23](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.23).



**Рис. 7.23.**Общий заголовок протокола передачи записей

Ниже приводится описание полей.

* **Протокол.** Это 1-байтовое поле определяет источник или пункт назначения инкапсулированного сообщения. Оно используется для мультиплексирования и демультиплексирования. Значения - 20 (*ChangeCipherSpec-протокол*), 21 (*аварийный протокол*), 22 (*протокол установления соединения*) и 23 (данные от прикладного уровня).
* **Версия.** Это 2-байтовое поле определяет версию SSL; один байт - первая цифра версии и другой - вторая. Текущая версия SSL - 3.0.
* **Длина.** Это 2-байтовое поле определяет размер сообщения (без заголовка) в байтах.

**ChangeCipherSpec-протокол**

Как мы говорили раньше, *протокол ChangeCipherSpec* имеет одно сообщение: ChangeCipherSpec. Это сообщение содержит только один байт, инкапсулированный в сообщение *протокола передачи записей* со значением 20, как показано на [рис. 7.24](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.24).



**Рис. 7.24.**Сообщение ChangeCipherSpec (CCS)

Однобайтовое поле в сообщении названо *CCS*, и его значение в настоящее время равно 1.

**Аварийный протокол**

*Аварийный протокол*, как мы говорили прежде, имеет одно сообщение - рапорт об ошибках в процессе. [рис. 7.25](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.25) показывает инкапсуляцию этого единственного сообщения в *Протоколе передачи записей* со значением 21.



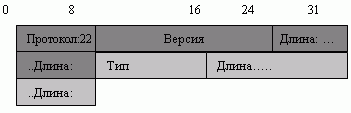
**Рис. 7.25.**Аварийное сообщение

Два поля аварийного сообщения разъясняются ниже.

* **Уровень.** Это однобайтовое поле определяет уровень ошибки. В настоящее время определены два уровня: предупреждение и полный отказ.
* **Описание.** Однобайтовое описание определяет тип ошибки.

**Протокол установления соединения**

Несколько сообщений были определены для *протокола установления соединения*. Все эти сообщения имеют четырехбайтовый типовой заголовок, показанный на [рис. 7.26](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.26). Рисунок приводит заголовок *протокола передачи записей* и типовой заголовок для *протокола установления соединения*. Обратите внимание, что значение поля протокола - 22.



**Рис. 7.26.**Типовой заголовок в протоколе установления соединения

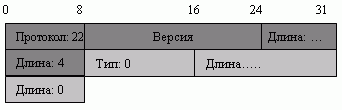
* **Тип.** Это однобайтовое поле определяет тип сообщения. В настоящее время определены десять типов, которые перечислены в [табл. 7.5](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#table.7.5).

|  |  |
| --- | --- |
| Таблица 7.5. Типы сообщений установления соединения | |
| **Тип** | **Сообщение** |
| 0 | HelloRequest |
| 1 | ClientHello |
| 2 | ServerHello |
| 11 | Certificate |
| 12 | ServerKeyExchange |
| 13 | CertificateRequest |
| 14 | ServerHelloDone |
| 15 | CertificateVerify |
| 16 | ClientKeyExchange |
| 20 | Finished |

* **Длина.** Это трехбайтовое поле определяет длину сообщения (исключая длину типа и поля длины). Читатель может задать вопрос: почему мы нуждаемся в двух полях длины - одном в общем заголовке *протокола передачи записей* и одном в типовом заголовке для сообщений установления соединения? Ответ: сообщение *протокола передачи записей* может доставить два сообщения установления соединения в одно и то же время, если нет потребности передать другое сообщение между ними.

**Сообщение HelloRequest**

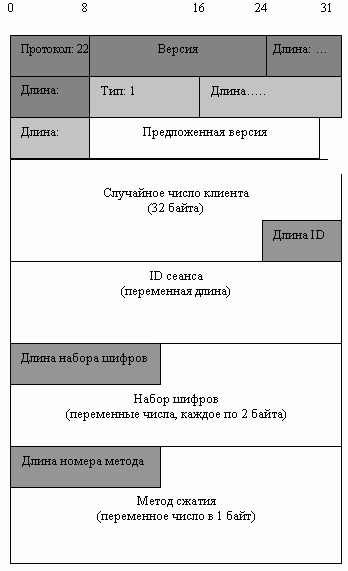
Сообщение HelloRequest, которое используется редко, является запросом от сервера к клиенту для перезапуска сеанса. Это может быть необходимо, если в сервере обнаружены сбои и необходим новый сеанс. Например, если сеанс становится таким длинным, что это угрожает его безопасности, сервер может передать рассматриваемое сообщение. Клиент тогда должен передать сообщение ClientHello и договориться о параметрах безопасности. [рис. 7.27](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.27) показывает формат такого сообщения. Это - четыре байта со значением типа 0. Он не имеет никакого текстового блока, так что значение поля длины - также 0.



**Рис. 7.27.**Сообщение HelloRequest

**Сообщение ClientHello**

Сообщение ClientHello - первое сообщение в обмене при установлении соединения. На [рис. 7.28](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.28) показан формат сообщения.



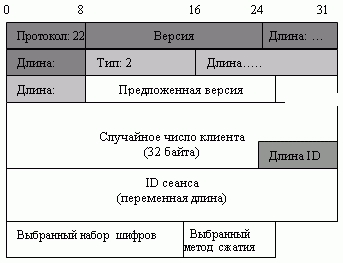
**Рис. 7.28.**Сообщение ClientHello

Поле "тип" и поле длины предварительно были уже обсуждены. Ниже приводится краткое описание других полей.

* **Версия.** Это 2-байтовое поле показывает версии используемого SSL. Версия - 3.0 для SSL и 3.1 - для TLS. Обратите внимание, что значение версии, например, 3.0, сохраняется в двух байтах: 3 в первом байте и 0 - во втором.
* **Случайное число клиента.** Это 32-байтовое поле используется клиентом, чтобы передать случайное число, которое создает параметры безопасности.
* **Длина ID сеанса.** Это 1-байтовое поле определяет длину ID сеанса (следующее поле). Если нет ID сеанса, значение этого поля - 0.
* **ID сеанса.** Значение этого поля переменной длины - 0, когда клиент начинает новый сеанс. ID сеанса инициируется сервером. Однако если клиент хочет возобновить предварительно остановленный сеанс, он может включить предварительно определенный ID сеанса в этом поле. Протокол определяет для ID сеанса максимум 32 байта.
* **Длина***набора шифров*. Это 2-байтовое поле определяет длину предложенного клиентом списка *набора шифров* (следующее поле).
* **Список***набора шифров*. Это поле переменной длины дает список *набора шифров*, поддерживаемых клиентом. Поле перечисляет *набор шифров* от наиболее предпочтительных к наименее предпочтительным. Каждый *набор шифров* кодируется как двухбайтовое число.
* **Длина методов сжатия.** Это 1-байтовое поле определяет длину списка предложенных клиентом методов сжатия (следующее поле).
* **Список методов сжатия.** Это поле переменной длины дает список методов сжатия, которые поддерживает клиент. Поле перечисляет методы от наиболее предпочтительных к наименее предпочтительным. Каждый метод кодируется как однобайтовое число. Сейчас единственный метод - метод NULL ("нет сжатия"). В этом случае значение длины методов сжатия - 1, и список метода сжатия имеет только один элемент со значением 0.

**Сообщение ServerHello**

Cообщение ServerHello - ответ сервера на сообщение ClientHello. Формат подобен сообщению ClientHello, но с меньшим количеством полей. [рис. 7.29](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.29) показывает формат сообщения.



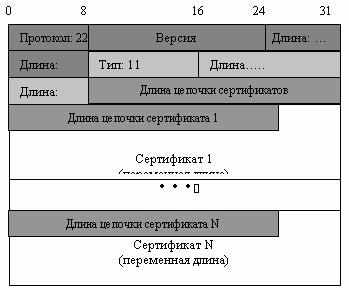
**Рис. 7.29.**Сообщение ServerHello

Поле версии - то же самое, что и в сообщении ClientHello. Поле случайного числа сервера определяет значение, выбранное сервером. Длина ID сеанса и поле ID сеанса - те же самые, что и в сообщении ClientHello. Однако ID сеанса - обычно пробел (и длина обычно устанавливается на 0), если сервер не возобновляет старый сеанс. Другими словами, если сервер позволяет возобновление сеанса, он вставляет соответствующее значение в поле ID сеанса, которое используется клиентом (в сообщении ClientHello), когда клиент желает повторно открыть старый сеанс.

Поле выбранного *набора шифров* определяет единственный *набор шифров*, который выбран сервером из списка, посланного клиентом. Поле методов сжатия определяет выбранный сервером метод из списка, посланного клиентом.

**Сообщение Certificate**

Сообщение Certificate может быть передано клиентом или сервером, чтобы передать список сертификатов открытого ключа. [рис. 7.30](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.30) показывает формат.



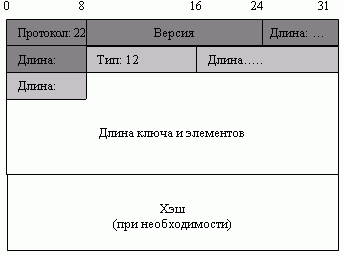
**Рис. 7.30.**Сообщение Certificate

Значение поля "тип" - 11. Текстовый блок сообщения включает следующие поля.

* **Длина цепочки сертификатов.** Это трехбайтовое поле показывает длину цепочки сертификатов. Это поле избыточно, потому что его значение - всегда на 3 меньше, чем значение поля длины.
* **Цепочка сертификатов.** Это поле переменной длины содержит цепочку сертификатов открытого ключа, который использует клиент или сервер. Для каждого сертификата есть два дополнительных поля:
  + трехбайтовое поле длины;
  + сам сертификат - поле переменного размера.

**Сообщение ServerKeyExchange**

ServerKeyExchange-сообщение передают от сервера к клиенту. [рис. 7.31](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.31) показывает общий формат.

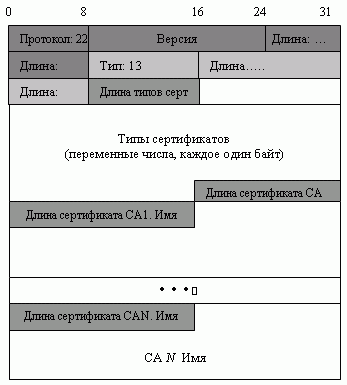


**Рис. 7.31.**Сообщение ServerKeyExchange

Сообщение содержит ключи, сгенерированные сервером. Формат сообщения зависит от *набора шифров*, выбранного в предыдущем сообщении. Клиент, который получает сообщение, должен интерпретировать его согласно предыдущей информации. Если сервер передал сообщение сертификата, то сообщение содержит подписанный параметр.

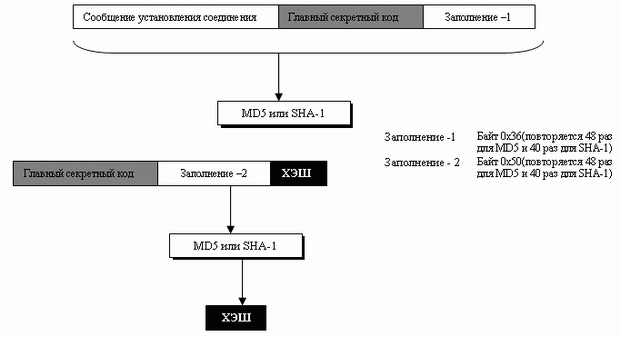
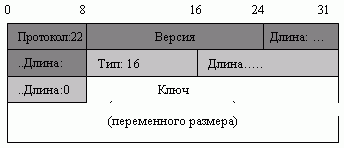
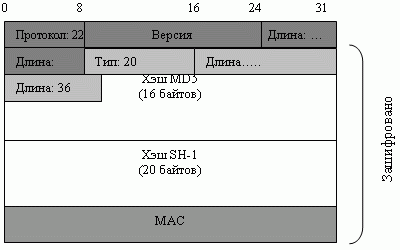
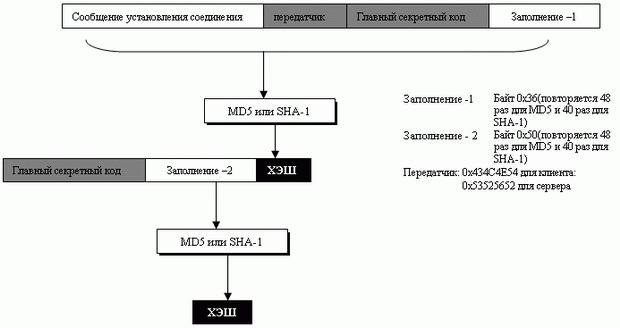
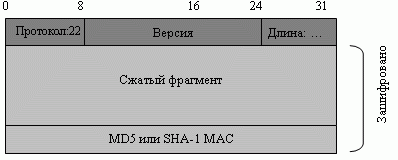
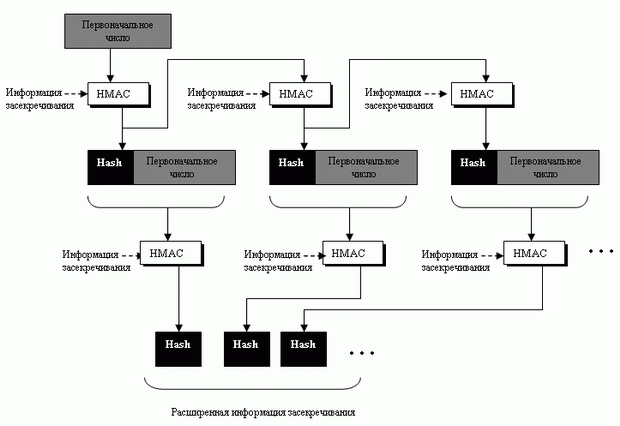
**Сообщение CertificateRequest**

CertificateRequest-сообщение передают от сервера к клиенту. Сообщение просит, чтобы клиент подтвердил серверу свою подлинность, а также подлинность одного из используемых сертификатов и одной из удостоверяющих администраций, названной в сообщении. [рис. 7.32](https://intuit.ru/studies/courses/553/409/lecture/9387?page=6#image.7.32) показывает общий формат.

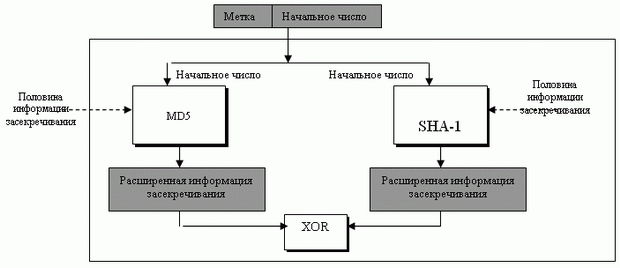
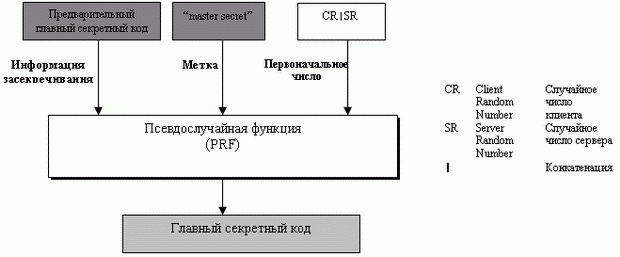
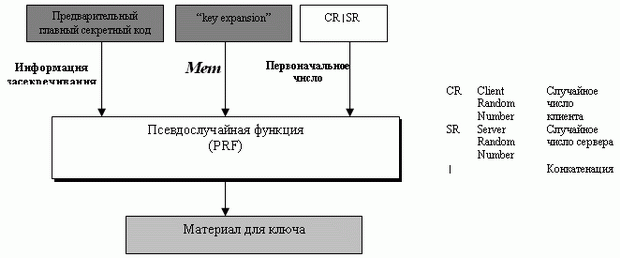


**Рис. 7.32.**Сообщение CertificateRequest

Значение поля типа - 13. Текстовый блок сообщения включает следующие поля:

* **Длина типов сертификатов** (**Len of Cert Types).** Это однобайтовое поле показывает длину типов сертификатов.
* **Типы сертификатов (Certificate Types).** Это поле переменной длины содержит список сертификата открытого ключа, который принят сервером. Каждый тип - один байт.
* **Длина сертификатов подтверждения подлинности (Length of CAs).**Это двухбайтовое поле показывает длину списка сертификатов, удостоверяющих подлинность (остальная часть пакета).
* **Длина** **сертификата подлинности x.** **Имя. (Length of CA x. Name).**Это двухбайтовое поле определяет длину *x*-ого сертификата и название администрации. *x* может принимать значение от 1 до N.
* **Сертификат x, подтверждающий подлинность. Имя.** ( **CA x. Name**). Это поле переменной длины определяет название **x**-ого сертификата администрации. x может принимать значение от 1 до N.
* **Сообщение ServerHelloDone**
* ServerHelloDone-сообщение - последнее сообщение, посылаемое во второй фазе процедуры установления связи. Сообщение сигнализирует, что Фаза II не доставляет никакой дополнительной информации. [рис. 7.33](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.33) показывает формат.
* 
* **Рис. 7.33.**Сообщение ServerHelloDone
* **Сообщение CertivicateVerify**
* Сообщение CertivicateVerify - это последнее сообщение Фазы III. В этом сообщении клиент доказывает, что он фактически имеет секретный ключ, связанный с его сертификатом открытого ключа. Чтобы доказать это, клиент создает хэш всех сообщений установления соединения, посланных перед этим сообщением, и подписывает их, используя алгоритм MD5 или SHA-1, основанный на типе сертификата клиента. [рис. 7.34](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.34) показывает формат.
* 
* **Рис. 7.34.**Сообщение CertivicateVerify
* Если секретный ключ клиента связан с DSS-сертификатом, то хэширование базируется только на алгоритме SHA-1, и длина хэш - 20 байтов. Если секретный ключ клиента связан с сертификатом RSА, то есть два (конкатенированных) хэша: один - основанный на MD5 и другой - основанный на SHA-1. Полная длина - 16 + 20 = 36 байтов. [рис. 7.35](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.35) показывает вычисления хэша.
* [](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_35.gif)
* [увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_35.gif)  
  **Рис. 7.35.**Вычисление хэш для сообщения CertivicateVerify
* **Сообщение ClientKeyExchange**
* ClientKeyExchange - второе сообщение, посылаемое в течение третьей фазы процедуры установления связи. В этом сообщении клиент обеспечивает ключи. Формат сообщения зависит от заданных алгоритмов смены ключей, выбранных двумя сторонами. [рис. 7.36](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.36) показывает общую идею формата.
* 
* **Рис. 7.36.**Сообщение ClientKeyExchange
* **Сообщение Finished**
* Cообщение Finished показывает, что переговоры закончены. Оно содержит все сообщения, которыми обменивались в течение процедуры установления связи, сопровождаемой указанием типа передатчика (клиент или сервер) и дополнением *главного секретного кода*. Точный формат зависит от типа используемого набора шифра. Общий формат показан на [рис. 7.37](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.37) - последовательное соединение (конкатенация) двух хэшей. На [рис. 7.38](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.38) изображено, как вычисляется каждый из них. Обратите внимание, что когда клиент или сервер передают сообщение Finished, они уже передали сообщение ChangedCipherSpec. Другими словами, криптографическая секретная информация (писать) находится в активном состоянии, клиент или сервер могут ее обработать. Сообщение Finished подобно фрагменту данных, прибывающему от прикладного уровня. Сообщение Finished может быть заверено (используя MAC из *набора шифров*) и зашифровано (используя алгоритм шифрования *набора шифров*).
* 
* **Рис. 7.37.**Сообщение Finished
* [](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_38.gif)
* [увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_38.gif)  
  **Рис. 7.38.**Вычисление хеша для сообщения Finished
* **Прикладные данные**
* *Протокол передачи записей* добавляет подпись в конце фрагмента (возможно, сжатый MAC), прибывающий от прикладного уровня, и затем зашифровывает фрагмент и MAC.
* После добавления общего заголовка со значением протокола 23 сообщение передачи записи передано. Обратите внимание, что общий заголовок не зашифрован. [рис. 7.39](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.39) показывает общий формат.
* 
* **Рис. 7.39.**Сообщение протокола передачи записей
* *Безопасность транспортного уровня* (*TLS* - *Transport Layer* *Security*) - протокол IEFT, стандартная версия протокола *SSL*. Эти два протокола очень похожи, но имеют небольшие отличия. Вместо того чтобы описывать *TLS* полностью, в этой секции мы только отметим отличия между протоколами *TLS* и *SSL*.
* **Версии**
* Первое отличие - номер версии (основное, но незначительное). Текущая версия SSL - 3.0; текущая версия TLS - 1.0. Другими словами, SSLv3.0 совместим с TLSv 1.0.
* **Набор шифров**
* Другое незначительное отличие между SSL и TLS - отсутствие поддержки *Fortezza*. TLS не поддерживает *Fortezza* для смены ключей или для шифрования/дешифрования. [табл. 7.6](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#table.7.6) показывает *набор шифров* для TLS.
* **Генерация криптографической секретности**
* Генерация криптографической секретности в TLS более сложная, чем в SSL. TLS сначала определяет две функции: *функцию расширения данных* и *псевдослучайную функцию*. Рассмотрим их.
* **Функция расширения данных**
* **Функция расширения данных** использует заранее заданный код аутентификации на основе хэширования (HMAC-HASH-BASED *MESSAGE AUTHENTICATION* CODE), или MD5, или SHA-1 для того, чтобы расширить информацию засекречивания. Эту функцию можно рассматривать как функцию, содержащую множество секций, где каждая секция создает одно значение хэширования. Расширенная секретность - последовательное соединение значений хэширования. Каждая секция использует два HMAC, информацию засекречивания и начальное число. *Функция расширения данных* - это формирование цепочки в виде многих секций. Однако чтобы сделать следующую секцию зависимой от предыдущей, второе начальное число - фактически выход первого HMAC предыдущей секции, как это показано на [рис. 7.40](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.40).
* [](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_40.gif)
* [увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_40.gif)  
  **Рис. 7.40.**Функция расширения данных
* *Псевдослучайная функция*
* TLS определяет *псевдослучайную функцию (PRF - PseudoRandom Function)*, чтобы получить комбинацию двух *функций расширения данных*: одна из них использует MD5 и другая - SHA-1. На *PRF* поступает три части информации: секретный код, метка и начальное число.

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица 7.6. Набор шифров для TLS | | | |
| **Набор шифров** | **Замена ключей** | **Шифрование** | **Хэш** |
| TLS\_NULL\_WITH\_NULL\_NULL | NULL | NULL | NULL |
| TLS\_RSA\_WITH\_NULL\_MD5 | RSA | NULL | MD5 |
| TLS\_RSA\_WITH\_NULL\_SHA | RSA | NULL | SHA-1 |
| TLS\_RSA\_W1TH\_RC4\_128\_MD5 | RSA | RC4 | MD5 |
| TLS\_RSA\_WITH\_RC4\_128\_SHA | RSA | RC4 | SHA-! |
| TLS\_RSA\_WITH\_IDEA\_CBC\_SHA | RSA | IDEA | SHA-1 |
| TLS\_RSA\_WITH\_DES\_CBC\_SHA | RSA | DES | SHA-1 |
| TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA | RSA | 3DES | SHA-1 |
| TLS\_DH\_anon\_WITH-RC4\_l 28\_MD5 | DH \_anon | RC4 | MD5 |
| TLS\_DH\_anon\_WITH\_DES\_CBC\_SHA | DH\_anon | DES | SHA-1 |
| TLS\_DH\_anon\_WITH\_3DES\_EDE\_CBC\_SHA | DH\_anon | 3DES | SHA-1 |
| TLS\_ DHE\_ RSA\_ WITH\_ DES\_ CBC\_ SHA | DHE\_RSA | DES | SHA-1 |
| TLS\_DHE\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA | DHE\_RSA | 3DES | SHA-1 |
| TLS\_ DHE\_ DSS\_ WITH\_ DES\_ CBC\_ SHA | DHE\_.DSS | DES | SHA-1 |
| TLS\_DHE\_DSS\_WITH\_3DES\_EDE\_CBC\_SHA | DHE\_DSS | 3DES | SHA-1 |
| TLS\_DH\_RSA\_WITH\_DES\_CBC\_SHA | DH\_RSA | DES | SHA-1 |
| TLS\_DH\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA | DH\_RSA | 3DES | SHA-1 |
| TLS\_ DH\_ DSS\_ WITH\_ DES\_ CBC\_ SHA | DH\_DSS | DES | SHA-1 |
| TLS\_DH\_DSS\_WITH\_3DES\_EDE\_CBC\_SHA | DH\_DSS | 3DES | SHA-1 |

* Метка и начальное число связаны и служат начальным числом для каждой *функции расширения данных*. Информация засекречивания разделена на две части; каждая часть используется как информация засекречивания для каждой *функции расширения данных*. Выходы двух *функций расширения данных* складывают по модулю два, чтобы создать конечную расширенную информацию засекречивания. Обратите внимание, что поскольку хэш создается MD5 и SHA-1, он имеет различные размеры, поэтому должны быть созданы дополнительные секции функций на базе MD5, чтобы сделать два вывода с одинаковым размером. [рис. 7.41](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.41) показывает идею применения *PRF*.
* [](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_41.gif)
* [увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_41.gif)  
  **Рис. 7.41.**PRF
* *Главный секретный код*
* TLS использует функцию *PRF*, чтобы создать *главный секретный код* от *предварительного главного секретного кода*. Это можно сделать, используя предварительный *главный секретный код* как информацию засекречивания, строку "*главный секретный код*" - как метку и последовательное соединение информации (конкатенацию) случайного числа клиента и случайное число сервера - как начальное число. Обратите внимание, что метка - фактически код ASCII строки "*главного секретного кода*". Другими словами, метка определяет выход для создания *главного секретного кода*. [рис. 7.42](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.42) иллюстрирует идею.
* [](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_42.gif)
* [увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_42.gif)  
  **Рис. 7.42.**Генерация главного секретного кода
* *Материал для ключей*
* TLS использует функцию *PRF*, чтобы создать *материал для ключей* от *главного секретного кода*. На сей раз информация засекречивания содержит: *главный секретный код*; метку - это строка "расширение ключа"; и начальное число - конкатенацию случайного числа сервера и случайного числа клиента, как это показано на [рис. 7.43](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.43).
* [](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_43.gif)
* [увеличить изображение](https://intuit.ru/EDI/18_11_15_6/1447798956-19904/tutorial/577/objects/7/files/07_43.gif)  
  **Рис. 7.43.**Генерация материала для ключа
* **Аварийный протокол**
* TLS поддерживает все аварийные сигналы, определенные в SSL, за исключением NoCertificate. TLS также добавляет к списку SSL некоторые новые. [табл. 7.7](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#table.7.7) показывает полный список аварийных сигналов, поддерживаемых TLS.

|  |  |  |
| --- | --- | --- |
| Таблица 7.7. Аварийные сигналы, определенные для TLS | | |
| ***Значение*** | ***Описание*** | ***Содержание*** |
| 0 | *CloseNotify* | Передатчик не будет посылать сообщений |
| 10 | *UnexpectedMessage* | Получено несоответствующее сообщение |
| 20 | *BadRecordMAC* | Получен некорректный MAC |
| 21 | *DecryptionFailed* | Дешифрованное сообщение недействительно |
| 22 | *RecordOverFlow* | Размер сообщения больше чем 214 + 2048 |
| 30 | *DecampressionFailure* | Невозможно соответствующее расширение сообщения |
| 40 | *HandsHakeFailure* | Передатчик не может завершить установление соединения |
| 42 | *BadCertificate* | Полученный сертификат искажен |
| 43 | *UnsupportedCertificate* | Полученный тип сертификата не поддерживается |
| 44 | *CertificateRevoked* | Пописавший аннулировал сертификат |
| 45 | *CertificateExpired* | Срок сертификата истек |
| 46 | *CertificateUnknown* | Сертификат неизвестен |
| 47 | *IllegalParameter* | Поле выходит за допустимые пределы или не соответствует им другим |
| 48 | *UnknownCA* | CA не может быть идентифицировано |
| 49 | *AcessDenied* | Нежелательно для продолжения переговоров |
| 50 | *Decode Error* | Полученное сообщение не может быть декодировано |
| 51 | *DecryptError* | Расшифровка зашифрованного текста недействительна |
| 60 | *ExportRestriction* | Проблемы согласования с ограничениями в США |
| 70 | *ProtocolVersion* | Эта версия протокола не поддерживается |
| 71 | *InsufficientSecurity* | Требуется больший набор секретных шифров |
| 80 | *InternalError* | Местная ошибка |
| 90 | *UserCanceled* | Данная сторона хочет прекратить переговоры |
| 100 | *NoRenegotiation* | Сервер не может снова начать переговоры по установлению соединения |

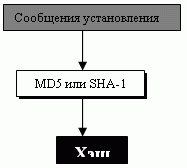
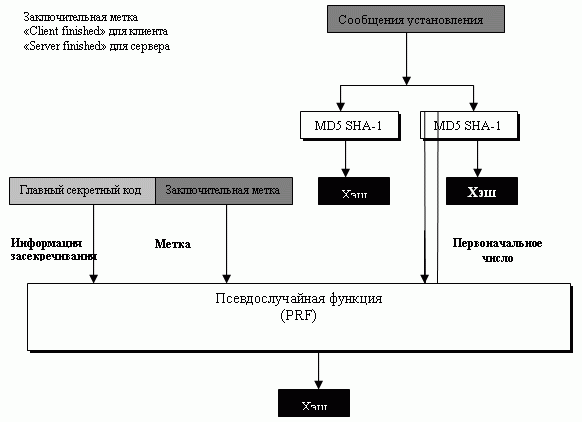
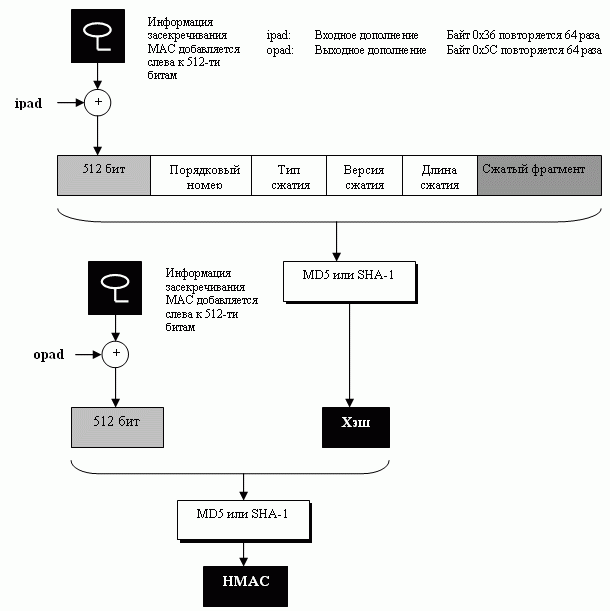
**Упражнения**

1. Какова длина *материала для ключей*, если *набор шифров* - один из перечисленных ниже:
   * SSL\_RSA\_WITH\_NULL\_MD5
   * SSL\_RSA\_WITH\_NULL\_SHA
   * TLS\_RSA\_WITH\_DES\_CBC\_SHA
   * TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
   * TLS\_DHE\_RSA\_WITH\_DES\_CBC\_SHA
   * TLS\_DH \_RSA \_ 3DES \_EDE\_CBC\_ SHA
2. Покажите число повторных модулей, необходимых для каждого случая в Упражнении 1 (см. [рис. 7.9](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#image.7.9)).
3. Сравните вычисление *главного секретного кода* в SSL с таким же процессом в TLS. В SSL предварительный главный код применяется в вычислении три раза, в TLS - только единожды. Какое вычисление более эффективно по объему и по времени?
4. Сравните вычисление *материала для ключей* в SSL и TLS. Ответьте на следующие вопросы:
   * Какое вычисление обеспечивает большую безопасность?
   * Какое вычисление более эффективно по объему и времени?
5. Вычисление *материала для ключей* в SSL требует нескольких итераций, в TLS этого не делается. Как может TLS вычислять *материал для ключей* переменной длины?
6. Когда сеанс продолжается с новым соединением, SSL не требует проведения полной процедуры установления связи. Покажите сообщения, которыми необходимо будет обменяться в частичной процедуре установления связи.
7. Когда сеанс продолжен, какая из следующей криптографической информации для засекречивания должна быть повторно вычислена?
   * *предварительный главный секретный код*
   * *главный секретный код*
   * ключи подтверждения подлинности
   * ключи шифрования
   * IV (первоначальный вектор)
8. Что случится в процессе на [рис. 7.20](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#image.7.20), если сервер передает сообщение ChangeCipherSpec, а клиент не передает? Какие сообщения могут быть переданы в *протоколе установления соединения*? Какие не могут быть переданы?
9. Сравните вычисление MAC в SSL и TLS (см. [рис. 7.22](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#image.7.22) и [рис. 7.46](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.46)). Какое из них более эффективно?
10. Сравните вычисление хэширования для сообщения CertificateVerify в SSL и TLS (см. [рис. 7.35](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.35) и [рис. 7.44](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.44)). Какое более эффективно?
11. Сравните вычисление хэширования для сообщения Finished в SSL и TLS (см. [рис. 7.38](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.38) и [рис. 7.45](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.45)). Ответьте на следующие вопросы:
    * Которое из них более безопасно?
    * Которое из них более эффективно?
12. TLS использует *PRF* для всех вычислений хэша кроме сообщения CertificateVerify. Объяснить это исключение.
13. Большинство протоколов записывают в виде формулы, чтобы показать вычисления криптографической секретности и хэширования. Например, в SSL вычисление *главного секретного кода* (см. [рис. 7.8](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#image.7.8)) отображается следующим образом (последовательная конкатенация записывается в виде линеек):
14. Master Secret = MD5 (pre-master | SHA-1 ("A" |pre-master I CR I SR)) |
15. MD5 (pre-master| SHA-1 ("A" | pre-master | CR I SR)) |
16. MD5 (pre-master |SHA-1 ("A" | pre-master | CR | SR))
17. Master Secret Главный секретный код
18. pre-master Предварительный главный секретный код
19. | Конкатенация
20. CR Случайное число клиента
21. SR Случайное число сервера

Упражнения

Запишите в виде формул следующие процессы:

* + *Материал для ключей* в SSL ([рис. 7.9](https://intuit.ru/studies/courses/553/409/lecture/9387?page=2#image.7.9))
  + MAC в SSL ([рис. 7.22](https://intuit.ru/studies/courses/553/409/lecture/9387?page=5#image.7.22))
  + Вычисление хэширования для сообщения CertificateVerify в SSL (рис. 7.35)
  + Вычисление хэширования для сообщения Finished в SSL ([рис. 7.38](https://intuit.ru/studies/courses/553/409/lecture/9387?page=7#image.7.38))
  + Расширение данных в TLS ([рис. 7.40](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.40))
  + *PRF* в TLS ([рис. 7.41](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.41))
  + *Главный секретный код* в TLS ([рис. 7.42](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.42))
  + *Материал для ключей* в TLS ([рис. 7.43](https://intuit.ru/studies/courses/553/409/lecture/9387?page=8#image.7.43))
  + Вычисление хэширования для сообщения CertificateVerify в TLS ([рис. 7.44](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.44))
  + Вычисление хэширования для Finished сообщения в TLS ([рис. 7.45](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.45))
  + MAC в TLS ([рис. 7.46](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.46))

1. Покажите, как SSL или TLS реагируют на атаку воспроизведения. То есть покажите, как SSL или TLS отвечает нападавшему, который пытается имитировать одно или более сообщений установления соединения (предварительно записав сообщение).
2. Покажите, как SSL или TLS реагируют на атаку грубой силы. Может ли злоумышленник использовать исчерпывающий компьютерный поиск и найти ключ шифрования в SSL или TLS? Какой протокол более безопасен в этом отношении - SSL или TLS?
3. Каков риск использования ключей короткой длины в SSL или TLS? Какую атаку злоумышленник может применить, если ключи коротки?
4. Действительно ли SSL или TLS относительно безопасны к атаке "посредника"? Может ли злоумышленник создать *материал для ключей* между клиентом и самим собой и между собой и сервером?
5. **Протокол установления соединения**
6. TLS вносит некоторые изменения в *протокол установления соединения*. Были специально изменены детали сообщения CertificateVerify и сообщения Finished.
7. **Сообщение CertificateVerify (сертификат верифицирован)**
8. В SSL хэширование, используемое в .Сообщении CertificateVerify . - это хэширование с двумя шагами, а именно: сообщение установления соединения плюс заполнение и *главный секретный код*. TLS упростил процесс: в нем хэшируется только сообщение установления соединения, как показано на [рис. 7.44](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.44).
9. 
10. **Рис. 7.44.**Хэш для сообщения CertificateVerify в TLS
11. **Сообщение Finished**
12. Было также изменено вычисление хэш для сообщения Finished. TLS использует *PRF*, чтобы вычислить два хэша, применяемых для сообщения Finished, как это показано на [рис. 7.45](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.45).
13. 
14. **Рис. 7.45.**Хэш для сообщения Finished в TLS
15. **Протокол передачи записей**
16. Единственное изменение в *протоколе передачи записей* - использование HMAC, чтобы подписать сообщение. TLS применяет MAC, как это определено в лекции 11, для того чтобы создать HMAC. TLS также добавляет версию протокола (названную сжатой версией) к тексту, который будет подписан. [рис. 7.46](https://intuit.ru/studies/courses/553/409/lecture/9387?page=9#image.7.46) показывает, как формируется HMAC.
17. 
18. **Рис. 7.46.**HMAC для TLS

Основные угрозы конфиденциальности

Конфиденциальную информацию можно разделить на: - предметную; - служебную. Служебная информация (например, пароли пользователей) не относится к определенной предметной области, в информационной системе она играет техническую роль, но ее раскрытие особенно опасно, поскольку оно чревато получением несанкционированного доступа ко всей информации, в том числе предметной. Даже если информация хранится в компьютере или предназначена для компьютерного использования, угрозы ее конфиденциальности могут носить некомпьютерный и вообще нетехнический характер. Если для доступа к различным системам используются многоразовые пароли или иная конфиденциальная информация, то наверняка эти данные будут храниться не только в голове, но и в записной книжке, которые пользователь часто оставляет на рабочем столе, а то и попросту теряет. Невозможно помнить много разных паролей и как результат применение несложных схем чередования или использование двух-трех легко запоминающихся (и столь же легко угадываемым) паролей. Описанный класс уязвимых мест можно назвать размещением конфиденциальных данных в среде, где им не обеспечена (зачастую - и не может быть обеспечена) необходимая защита. Угроза же состоит в том, что кто-то не откажется узнать секреты, которые сами просятся в руки. Помимо паролей, хранящихся в записных книжках пользователей, в этот класс попадает передача конфиденциальных данных в открытом виде (в разговоре, в письме, по сети), которая делает возможным перехват данных. Для атаки могут использоваться разные технические средства (подслушивание или прослушивание разговоров, пассивное прослушивание сети и т.п.), но идея одна - осуществить доступ к данным в тот момент, когда они наименее защищены. Весьма опасной угрозой являются... выставки, на которые многие организации, недолго думая, отправляют оборудование из производственной сети, со всеми хранящимися на них данными. Остаются прежними пароли, при удаленном доступе они продолжают передаваться в открытом виде. Это плохо даже в пределах защищенной сети организации; в объединенной сети выставки - это слишком суровое испытание честности всех участников. Еще один пример изменения, о котором часто забывают, - хранение данных на резервных носителях. Для защиты данных на основных носителях применяются развитые системы управления доступом; копии же нередко просто лежат в шкафах и получить доступ к ним могут многие. 61 Перехват данных - очень серьезная угроза, и если конфиденциальность действительно является критичной, а данные передаются по многим каналам, их защита может оказаться весьма сложной и дорогостоящей. Технические средства перехвата хорошо проработаны, доступны, просты в эксплуатации, а установить их, например на кабельную сеть, может кто угодно, так что эту угрозу нужно принимать во внимание по отношению не только к внешним, но и к внутренним коммуникациям. Кражи оборудования являются угрозой не только для резервных носителей, но и для компьютеров, особенно портативных. Часто ноутбуки оставляют без присмотра на работе или в автомобиле, иногда просто теряют. Опасной нетехнической угрозой конфиденциальности являются методы морально-психологического воздействия, такие как маскарад - выполнение действий под видом лица, обладающего полномочиями для доступа к данным. К неприятным угрозам, от которых трудно защищаться, можно отнести злоупотребление полномочиями. На многих типах систем привилегированный пользователь (например системный администратор) способен прочитать любой (незашифрованный) файл, получить доступ к почте любого пользователя и т.д. Другой пример - нанесение ущерба при сервисном обслуживании. Обычно сервисный инженер получает неограниченный доступ к оборудованию и имеет возможность действовать в обход программных защитных механизмов. Таковы основные угрозы, которые наносят наибольший ущерб субъектам информационных отношений.

ВРЕДОНОСТНЫЕ ПРОГРАММЫ И КОМПЬЮТЕРНЫЕ ВИРУСЫ 13.1 Основные понятия Для хранения информации на любом компьютере используются два вида памяти - постоянные запоминающие устройства (ПЗУ) и постоянные запоминающие устройства (ПЗУ). К первому типу относятся жесткие диски (винчестеры), дискеты, CD-ROM и другие мобильные носители информации, ко второму - оперативная память, то есть микросхема в системном блоке. Поэтому ПЗУ также называют внешней долговременной памятью, а ОЗУ - внутренней. Главное отличие оперативной памяти от внешней состоит в том, что информация, записанная на ОЗУ, может храниться только во время работы компьютера, при выключении или перезагрузке она теряется. Как следствие, большинство устройств ПЗУ предназначены для хранения значительно большего объема информации, чем оперативная память. Для организации хранилища информации на ПЗУ используются файлы. Файл - это логический блок информации, хранимой на носителях информации. Файл обязательно имеет имя и может содержать произвольный объем информации. Максимальная длина имени и максимальный объем файла определяются файловой системой. Файловая система - это совокупность правил, определяющих систему хранения информации: различные атрибуты файлов, такие как максимальная длина имени, максимальный допустимый размер файла. Примеры файловых систем - FAT, FAT32, NTFS, EXT2, ISO9660. Компьютерная программа- это последовательность инструкций (команд) для выполнения компьютером определенных действий. Программы записываются при помощи специальных языков программирования или машинного кода. Примеры компьютерных программ - программа чтения и записи данных на дискету, программа воспроизведения музыки с диска, записная книжка в мобильном телефоне, Microsoft Word. Передача программе пользовательских данных может осуществляться с помощью графического интерфейса, командной строки, конфигурационного файла или косвенно через другие программы. Конфигурационный файл представляет собой текстовый файл с последовательным перечнем данных и команд, которые необходимо передать программе. При взаимодействии же двух программ между собой пользователь как правило явного участия не принимает. Вызов компьютерной программы, то есть запуск программы на выполнение, производится путем последовательной загрузки содержимого соответствующего ей файла в оперативную память, после чего компьютер начинает выполнять последовательность заложенных в эту программу действий. 153 Запустить программу можно также непрямым методом. Например, при доступе к любому файлу, содержащему текстовую информацию, должна запускаться программа, позволяющая его прочесть, то есть преобразовывающая машинный код, содержащийся в текстовом файле, в буквы, которые пользователь прочитает на экране. Таким образом, практически все программы помимо основных функций, выполняют ряд дополнительных, служебных действий, не видимых обычному пользователю. Вредоносная программа - это программа, наносящая какой-либо вред компьютеру, на котором она запускается, или другим подключенным к нему компьютерам. Одним из способов для вредоносной программы оставаться незамеченной на компьютере является дописывание своего кода к файлу другой известной программы. При этом возможно как полное перезаписывание файлов (но в этом случае вредоносная программа обнаруживает себя при первом же запуске, поскольку ожидаемые действия полностью заменены), так и внедрение в начало, середину или конец файла. Пример. CIH - вирус, который в ходе заражения записывает свои копии во все запускаемые пользователем программные файлы (PE EXE). Внедрение может происходить как одним куском, так и путем деления вредоносного кода на блоки и записи их в разных частях заражаемого файла. При этом инфицированная программа может дальше выполнять свои основные функции и вирус в ней никак себя не обнаруживает. Однако в определенный момент времени происходит уничтожение всей информации на жестком диске. Поскольку самая известная версия CIH срабатывала 26 апреля, то он получил второе имя - «Чернобыль». 13.2 Способы распространения вредоносных программ В настоящее время имеется четыре основных способа передачи вредоносного ПО. 1. Мобильные носители. К мобильным носителям можно отнести все виды энергонезависимых ПЗУ. То есть таких устройств, которые позволяют достаточно долго хранить информацию и при этом не требуют дополнительного питания от компьютера. Это дискеты, компакт диски, flash-накопители, перфокарты и перфоленты. Мобильные носители - достаточно распространенный способ для размножения компьютерных вирусов. Однако по скорости распространения этот путь существенно уступает компьютерным сетям. 2. Локальная вычислительная сеть (ЛВС)2) - это компьютерная сеть, покрывающая относительно небольшую территорию (дом, школу, институт, микрорайон). 154 Вредоносные программы в полной мере используют преимущества ЛВС - фактически, почти все современные вирусы имеют встроенные процедуры инфицирования по локальным сетям и как следствие высокие темпы распространения. Инфицирование обычно происходит в такой последовательности. Зараженный компьютер с заданным интервалом инициирует соединение поочередно со всеми другими компьютерами сети и проверяет наличие на них открытых для общего доступа файлов. Если такие есть, происходит инфицирование. 3. Глобальная вычислительная сеть (ГВС) - это компьютерная сеть, покрывающая большие территории - города, страны, континенты. Самая большая и самая известная на сегодняшний день глобальная вычислительная сеть - это всемирная сеть Интернет. Наличие сети такого масштаба делает возможным всемирные эпидемии компьютерных вирусов. Пример. 30 апреля 2004 года были обнаружены первые экземпляры вируса Sasser - в течение дня им было атаковано около 4 тысяч компьютеров, что вызвало серьезные сбои в работе таких компаний как Postbank, Delta Air Lines, Goldman Sachs. Впоследствии было поражено более 8 млн. компьютеров, а убытки от Sasser были оценены в 979 млн. долларов США. 4. Электронная почта - это способ передачи информации в компьютерных сетях, основанный на пересылке пакетов данных, называемых электронными письмами. На сегодняшний день электронная почта выступает основным путем распространения вирусов. Это происходит потому, что время доставки письма очень мало (обычно исчисляется минутами) и практически все пользователи Интернет имеют как минимум один почтовый ящик. При этом для того, чтобы доставить пользователю на компьютер зараженный файл, не нужно его принуждать куда-либо обратиться и скопировать к себе вирус. Достаточно лишь прислать на его электронный адрес инфицированное письмо и заставить адресата его открыть. Часто для инфицирования даже не требуется запускать вложение - существуют методы, позволяющие заражать даже при обычном прочтении письма. Пример 1. Horillka распространяется через Интернет в виде файлов, прикрепленных к зараженным письмам с такими параметрами: заголовок - «Внимание!», текст: «Выпущено новое vbs обновление для поиска вирусов в памяти ОС Windows! Оно помогает бороться с вирусами, рассылающимися по почте. Антивирусный модуль написан на скрипт-языке, что помогает перехватывать vb и js вирусы, прежде чем они начнут деструктивную деятельность. Достаточно открыть файл и программа по устранению вирусов проведет поиск вредоносных программ в памяти компьютера». Во вложении находится файл с именем «WinSys32dll.vbs», после его запуска происходит заражение компьютера. Как результат, 11 декабря каждого года на экран 155 выдается сообщение «COOOOOOOOL» и после следующей перезагрузки уничтожаются все данные на жестком диске С. Пример 2. LoveLetter в мае 2000 года в течение всего нескольких часов заразил миллионы компьютеров по всему миру. Такому успеху способствовала удачно выбранная тема, интригующий текст и имя вложенного файла - «ILOVEYOU», «kindly check the attached LOVELETTER coming from me» и «LOVE-LETTER-FOR-YOU.TXT.vbs. После заражения происходила кража конфиденциальной информации и искажение содержимого некоторых файлов на жестком диске. 13.3 Операционная система. Уязвимости и заплаты Все программы можно разделить на два типа - прикладные и системные. Прикладное программное обеспечение (прикладные программы) - это программы, предназначенные для выполнения определенных пользовательских задач и рассчитанные на непосредственное взаимодействие с пользователем. Прикладные программы часто называют приложениями. Системное программное обеспечение используется для обеспечения работы компьютера самого по себе и выполнения прикладных программ. В персональном компьютере под прикладными программами понимаются различные текстовые редакторы, игры, почтовые программы, электронные словари. Роль базового системного программного обеспечения играет операционная система. Операционная система (ОС) - это комплекс программ, который обеспечивает управление физическими устройствами компьютера, доступ к файлам, ввод и вывод данных, выполнение и взаимодействие пользовательских программ. Наличие автозагрузки дает возможность вредоносным вирусам практически незаметно выполнять свои функции. Для этого во время заражения в список автозагрузки добавляется ссылка на программу, которая загружает вирус в оперативную память при каждой загрузке операционной системы. То есть фактически активация вируса происходит без участия пользователя при каждом включении компьютера. Уязвимость (или брешь в системе безопасности) - это место в программном коде, которое теоретически или реально может быть использовано для несанкционированного доступа к управлению программой. Уязвимости могут появляться как в системном, так и в прикладном программном обеспечении. После обнаружения уязвимости, производители программ обычно стараются как можно скорее выпустить дополнения, которые бы исправляли исходный код и закрывали брешь. Заплата или патч (от англ. patch - латать, ставить заплаты) - это программный код, используемый для модификации используемой программы. 156 Другими словами заплата - это дополнительная программа, которую следует запустить на выполнение, если в уже используемой программе обнаружилась ошибка или уязвимость. При этом часто можно устанавливать патч без удаления основной программы и даже без завершения ее работы - в первую очередь это касается операционных систем. Пример. В январе 2003 года началась эпидемия Slammer, заражающего сервера под управлением операционной системы Microsoft SQL Server 2000. Вирус использовал брешь в системе безопасности SQL Server, заплата к которой вышла еще в июле 2002. После проникновения Slammer начинал в бесконечном цикле посылать свой код на случайно выбранные адреса в сети - только за первые 10 минут было поражено около 90% (120 000 единиц) всех уязвимых серверов, при этом пять из тринадцати главных серверов Интернет вышли из строя. 13.4 Последствия заражений вредоносной программой Последствия инфицирования компьютера вредоносной программой могут быть как явными, так и неявными. К неявным последствиям обычно относят заражения программами, которые по своей сути являются вирусами, однако из-за ошибок в своем коде или нестандартному программному обеспечению целевого компьютера, вредоносную нагрузку выполнить не могут. При этом свое присутствие в системе они никак не выражают. Класс явных последствий постоянно увеличивается. К ним можно отнести: 1. Несанкционированная рассылка электронных писем. Ряд вирусов после заражения компьютера ищут на жестком диске файлы, содержащие электронные адреса и без ведома пользователя начинают рассылку по ним инфицированных писем. Пример. Sircam рассылал себя с зараженных компьютеров в виде файлов, вложенных в письма электронной почты. Для этого случайным образом на жестком диске выбирался файл, к которому прикреплялся вирусный код (дописывался в конец файла). Таким образом отсылаемые письма содержали вложение, состоящее из двух частей: вирус и файлприманку. Имя вложения формировалось на основе выбранного файла - например, если исходный файл назывался photos.zip, то имя вложения было - photos.zip.pif, photos.zip.lnk, photos.zip.bat или photos.zip.com. Адреса получателей выбирались из найденных на зараженном компьютере, а текст писем составлялся так, чтобы внушить как можно меньше подозрений и заставить адресата запустить полученный файл. Побочным эффектом такого способа распространения является утечка с зараженного компьютера конфиденциальных документов. 157 2. Кража конфиденциальной информации. После инфицирования вирус ищет файлы, содержащие конфиденциальную информацию (номера кредитных карт, различные пароли, секретные документы), для кражи которой он предназначен, и передает ее хозяину. Это может происходить путем отправки выбранных данных в электронном сообщении на определенный адрес или прямой пересылки их на удаленный сервер. 3. Несанкционированное использование сетевых ресурсов. Существуют вирусы, которые после заражения без ведома пользователя подключаются к различным платным службам с использованием личных данных, найденных на компьютере. Впоследствии жертве приходится оплачивать не заказанные ею услуги, а злоумышленник обычно получает процент от этого счета. Пример. Dialer - после попадания на компьютер, этот вирус начинал дозвон на международные телефонные номера для подключения к платным сервисам. Через некоторое время пользователю приходил огромный телефонный счет и доказать в подавляющем большинстве случаев что он никуда не звонил не представлялось возможным. 4. Удаленное управление компьютером. После того, как произошло заражение, некоторые вирусы передают своему хозяину инструменты для удаленного управления инфицированным компьютером - открывают бекдоры (от англ. backdoor - черный ход). Обычно это выражается в возможности удаленно запускать размещенные на нем программы, а также загружать из Интернет по желанию злоумышленника любые файлы. Свое присутствие такие программы обычно выражают только в использовании части ресурсов зараженного компьютера для своих нужд - в основном процессора и оперативной памяти. Такие компьютеры часто называют машинами-зомби. 5. Ботнеты. Группа компьютеров, которыми централизованно управляет один злоумышленник, называется ботнетом. Число таких компьютеров в Интернет на сегодняшний день достигает нескольких миллионов и продолжает увеличиваться каждый день. Пример. Bagle - вирус, распространяющийся в виде вложения в электронные письма. Адрес отправителя и имя вложения - произвольные, тема - «Hi», текст - «Test =)». После заражения он копирует себя на жесткий диск под именем bbeagle.exe и регистрирует этот файл в автозапуске операционной системы. Далее происходят попытки соединиться с несколькими удаленными серверами. При этом злоумышленнику предоставляется возможность загружать на зараженный компьютер любые файлы и запускать их на выполнение. Первый вирус из этой серии, Bagle.a, был обнаружен 18 января 2004, однако по замыслу автора уже через 10 дней он перестал размножаться и вскоре появились новые, более совершенные 158 модификации Bagle. В результате автор получил огромную сеть подконтрольных ему компьютеров. Bagle-ботнет - одна из самых масштабных и известных сетей машин-зомби. 6. Несанкционированная атака на чужой сервер. Последнее время вирусописатели используют ботнеты для организации так называемых DoSатак. DoS (от англ. Denial of Service) - это построенное на принципе отказа в обслуживании нападение на удаленный сайт. Это означает, что каждый инфицированный компьютер периодически (с интервалом обычно порядка 1 секунды) посылает произвольный запрос на получение информации с заданного злоумышленником сайта. Все веб-сайты рассчитаны на определенное число запросов в единицу времени, поэтому резкое увеличение нагрузки практически всегда выводит сервер из строя. Атака, которая производится одновременно с большого количества компьютеров, называется распределенной DoS-атакой или DDoS (от англ. Distributed Denial of Service). Пример.Одна из самых известных DDoS-атак была предпринята в июле 2001 года. Объектом нападения стал веб-сайт Белого дома в США (www.whitehouse.gov). В атаке участвовало около 12000 (по другим данным - до 200000) компьютеров, зараженных во время прошедшей незадолго до этого эпидемии вируса CodeRed. 7. Рассылка спама. Под этим термином обычно понимается ненужная, нежелательная, не запрошенная получателем корреспонденция. Спам может приходить как по электронной почте, так и в виде других сообщений, например на мобильный телефон в виде SMS. Поскольку электронных адресов в Интернет очень много, рассылка спама занимает много ресурсов. Поэтому злоумышленники часто используют для этих целей ботнеты. 8. Фишинг. Фактически фишинг - это метод кражи чужой информации, суть которого заключается в подделке известного сайта и рассылке электронных писем-приглашений зайти на него и ввести свою конфиденциальную информацию. Например, создается точная копия сайта какого-либо банка и с помощью спам-технологий рассылается письмо, максимально похожее на настоящее, с уведомлением о сбое в программном обеспечении и просьбой зайти на сайт и заново ввести свои данные. Тут же, в письме приводится адрес сайта - естественно, поддельный, но также максимально похожий на правду. Существует международная организация, ведущая учет фишинговым инцидентам - Anti-Phishing Working Group (www.antiphishing.org). 9. Уничтожение информации. Большинство современных вредоносных программ если и несут в себе процедуры уничтожения 159 информации на компьютере-жертве, то только в качестве дополнительной, не основной функции. Однако для многих пользователей это наиболее явное и болезненное последствие - удаленным и не подлежащим восстановлению может оказаться любой файл на жестком диске, как детские фотографии, так и только что законченная курсовая работа или книга. 10. Мистификации. Иногда на электронную почту или по другим каналам приходят так называемые предупреждения о новых вирусах. Обычно они содержат призывы не ходить по приведенным ссылкам, проверить свой компьютер на наличие на нем вируса указанным в сообщении методом или предостережение не принимать почту с определенными параметрами. Чаще всего это просто мистификация. Вреда, если не предпринимать указанные действия и не пересылать всем друзьям и знакомым, нет. Пример. В апреле 2004 года произошла массовая рассылка предупреждения о якобы опасном вирусе, основным признаком присутствия которого на компьютерах под управлением операционной системы Microsoft Windows заявлялось наличие файла jdbgmgr.exe, который и содержит саму вредоносную программу. В действительности же этот файл является стандартной программой, входящей в большинство версий Microsoft Windows. Удаление или изменение содержимого jdbgmgr.exe влечет непредсказуемые последствия в работоспособности операционной системы. 13.5 Классификация вредоностных программ Все вредоносные программы в соответствии со способами распространения и вредоносной нагрузкой можно разделить на четыре основные типа: - компьютерные вирусы, - черви, - трояны - другие программы. Рассмотрим основные особенности указанных типов подробнее. 13.5.1 Вирусы Основная черта компьютерного вируса - это способность к саморазмножению. Компьютерный вирус- это программа, способная создавать свои дубликаты (не обязательно совпадающие с оригиналом) и внедрять их в вычислительные сети и/или файлы, системные области компьютера и прочие выполняемые объекты. При этом дубликаты сохраняют способность к дальнейшему распространению. Условно жизненный цикл любого компьютерного вируса можно разделить на пять стадий: - Проникновение на чужой компьютер. 160 - Активация. - Поиск объектов для заражения. - Подготовка копий. - Внедрение копий. Путями проникновения вируса могут служить как мобильные носители, так и сетевые соединения - фактически, все каналы, по которым можно скопировать файл. Однако в отличие от червей, вирусы не используют сетевые ресурсы - заражение вирусом возможно, только если пользователь сам каким-либо образом его активировал. Например, скопировал или получил по почте зараженный файл и сам его запустил или просто открыл. После проникновения следует активация вируса. Это может происходить несколькими путями. В соответствии с выбранным методом активации вирусы делятся на следующие виды: 1. Загрузочные вирусы заражают загрузочные сектора жестких дисков и мобильных носителей. Примеры. Вредоносная программа Virus.Boot.Snow.a записывает свой код в MBR жесткого диска или в загрузочные сектора дискет. При этом оригинальные загрузочные сектора шифруются вирусом. После получения управления вирус остается в памяти компьютера (резидентность) и перехватывает прерывания INT 10h, 1Ch и 13h. Иногда вирус проявляет себя визуальным эффектом - на экране компьютера начинает падать снег. Другой загрузочный вирус Virus.Boot.DiskFiller также заражает MBR винчестера или загрузочные сектора дискет, остается в памяти и перехватывает прерывания - INT 13h, 1Ch и 21h. При этом, заражая дискеты, вирус форматирует дополнительную дорожку с номером 40 или 80 (в зависимости от объема дискеты он может иметь 40 либо 80 дорожек с номерами 0-39 или 0-79 соответственно). Именно на эту нестандартную дорожку вне поля обычной видимости вирус записывает свой код, добавляя в загрузочный сектор лишь небольшой фрагмент - головную часть вируса. 2. Файловые вирусы - заражают файлы. Отдельно по типу среды обитания в этой группе также выделяют следующие типы. 2.1 Классические файловые вирусы - они различными способами внедряются в исполняемые файлы (внедряют свой вредоносный код или полностью их перезаписывают), создают файлы-двойники, свои копии в различных каталогах жесткого диска или используют особенности организации файловой системы. Пример. Самый известный файловый вирус всех времен и народов — Virus.Win9x.CIH, известный также как «Чернобыль». Имея небольшой размер - около 1 кб - вирус заражает PE-файлы (Portable Executable) на 161 компьютерах под управлением операционных систем Windows 95/98 таким образом, что размер зараженных файлов не меняется. Для достижения этого эффекта вирус ищет в файлах «пустые» участки, возникающие из-за выравнивания начала каждой секции файла под кратные значения байт. После получения управления вирус перехватывает IFS API, отслеживая вызовы функции обращения к файлам и заражая исполняемые файлы. 26 апреля срабатывает деструктивная функция вируса, которая заключается в стирании Flash BIOS и начальных секторов жестких дисков. Результатом является неспособность компьютера загружаться вообще (в случае успешной попытки стереть Flash BIOS) либо потеря данных на всех жестких дисках компьютера. 2.2 Макровирусы, которые написаны на внутреннем языке, так называемых макросах какого-либо приложения. Подавляющее большинство макровирусов используют макросы текстового редактора Microsoft Word. Пример. Одними из наиболее разрушительных макровирусов являются представители семейства Macro.Word97.Thus. Эти вирусы содержат три процедуры Document\_Open, Document\_Close и Document\_New, которыми подменяет стандартные макросы, выполняющиеся при открытии, закрытии и создании документа, тем самым обеспечивая заражение других документов. 13 декабря срабатывает деструктивная функция вируса - он удаляет все файлы на диске C:, включая каталоги и подкаталоги. 2.3 Скрипт-вирусы, написанные в виде скриптов для определенной командной оболочки - например, bat-файлы для DOS или VBS и JS - скрипты для Windows Scripting Host (WSH). Пример. Virus.VBS.Sling написан на языке VBScript (Visual Basic Script). При запуске он ищет файлы с расширениями .VBS или .VBE и заражает их. При наступлении 16-го июня или июля вирус при запуске удаляет все файлы с расширениями .VBS и .VBE, включая самого себя. Дополнительным отличием вирусов от других вредоносных программ служит их жесткая привязанность к операционной системе или программной оболочке, для которой каждый конкретный вирус был написан. Это означает, что вирус для Microsoft Windows не будет работать и заражать файлы на компьютере с другой установленной операционной системой, например Unix. Точно также макровирус для Microsoft Word 2003 скорее всего не будет работать в приложении Microsoft Excel 97. При подготовке своих вирусных копий для маскировки от антивирусов могут применяться такие технологии как: - Шифрование — вирус состоит из двух функциональных кусков: собственно вирус и шифратор. Каждая копия вируса состоит из шифратора, случайного ключа и собственно вируса, зашифрованного этим ключом. 162 - Метаморфизм — создание различных копий вируса путем замены блоков команд на эквивалентные, перестановки местами кусков кода, вставки между значащими кусками кода «мусорных» команд, которые практически ничего не делают. Сочетание этих двух технологий приводит к появлению следующих типов вирусов классифицируемых по технологии защиты от обнаружения: 1. Шифрованный вирус — вирус, использующий простое шифрование со случайным ключом и неизменный шифратор. Такие вирусы легко обнаруживаются по сигнатуре шифратора. 2. Метаморфный вирус — вирус, применяющий метаморфизм ко всему своему телу для создания новых копий. 3. Полиморфный вирус — вирус, использующий метаморфный шифратор для шифрования основного тела вируса со случайным ключом. При этом часть информации, используемой для получения новых копий шифратора также может быть зашифрована. Например, вирус может реализовывать несколько алгоритмов шифрования и при создании новой копии менять не только команды шифратора, но и сам алгоритм. Пример. Одним из наиболее сложных и относительно поздних полиморфных вирусов является Virus.Win32.Etap. При заражении файла вирус перестраивает и шифрует собственный код, записывает его в одну из секций заражаемого файла, после чего ищет в коде файла вызов функции ExitProcess и заменяет его на вызов вирусного кода. Таким образом, вирус получает управление не перед выполнением исходного кода зараженного файла, а после него. Основные цели любого компьютерного вируса - это распространение на другие ресурсы компьютера и выполнение специальных действий при определенных событиях или действиях пользователя (например, 26 числа каждого четного месяца или при перезагрузке компьютера). Специальные действия нередко оказываются вредоносными. 13.5.2 Черви В отличие от вирусов черви - это вполне самостоятельные программы. Главной их особенностью также является способность к саморазмножению, однако при этом они способны к самостоятельному распространению с использованием сетевых каналов. Для подчеркивания этого свойства иногда используют термин «сетевой червь». Червь (сетевой червь) - это вредоносная программа, распространяющаяся по сетевым каналам и способная к самостоятельному преодолению систем защиты компьютерных сетей, а также к созданию и 163 дальнейшему распространению своих копий, не обязательно совпадающих с оригиналом. Пример. Классическими сетевыми червями являются представители семейства Net-Worm.Win32.Sasser. Эти черви используют уязвимость в службе LSASS Microsoft Windows. При размножении, червь запускает FTPслужбу на TCP-порту 5554, после чего выбирает IP-адрес для атаки и отсылает запрос на порт 445 по этому адресу, проверяя, запущена ли служба LSASS. Если атакуемый компьютер отвечает на запрос, червь посылает на этот же порт эксплойт уязвимости в службе LSASS, в результате успешного выполнения которого на удаленном компьютере запускается командная оболочка на TCP-порту 9996. Через эту оболочку червь удаленно выполняет загрузку копии червя по протоколу FTP с запущенного ранее сервера и удаленно же запускает себя, завершая процесс проникновения и активации. Жизненный цикл червей состоит из таких стадий: - Проникновение в систему. - Активация. - Поиск объектов для заражения. - Подготовка копий. - Распространение копий. В зависимости от способа проникновения в систему черви делятся на типы: - сетевые черви используют для распространения локальные сети и Интернет; - почтовые черви - распространяются с помощью почтовых программ; - IM-черви используют системы мгновенного обмена сообщениями; - IRC-черви распространяются по каналам IRС; - P2P-черви - при помощи пиринговых файлообменных сетей. После проникновения на компьютер, червь должен активироваться - иными словами запуститься. По методу активации все черви можно разделить на две большие группы. 1. Требующие активного участия пользователя. Отличительная особенность таких является использование обманных методов. Это проявляется, например, когда получатель инфицированного файла вводится в заблуждение текстом письма и добровольно открывает вложение с почтовым червем, тем самым его активируя. 2. Не требующие активного участия пользователя. Активация сетевого червя без участия пользователя всегда означает, что червь использует бреши в безопасности программного обеспечении 164 компьютера. Это приводит к очень быстрому распространению червя внутри корпоративной сети с большим числом станций, существенно увеличивает загрузку каналов связи и может полностью парализовать сеть. Именно этот метод активации использовали черви Lovesan и Sasser. В последнее время наметилась тенденция к совмещению этих двух технологий - такие черви наиболее опасны и часто вызывают глобальные эпидемии. Сетевые черви могут кооперироваться с вирусами - такая пара способна самостоятельно распространяться по сети (благодаря червю) и в то же время заражать ресурсы компьютера (функции вируса). 13.5.3 Троянские программы Трояны или программы класса троянский конь, в отличие от вирусов и червей, не обязаны уметь размножаться. Троян (троянский конь) - программа, основной целью которой является вредоносное воздействие по отношению к компьютерной системе путем выполнения несанкционированных пользователем действий: кражи, порчи или удаления конфиденциальных данных, нарушения работоспособности компьютера или использования его ресурсов в неблаговидных целях. Некоторые трояны способны к самостоятельному преодолению систем защиты компьютерной системы, с целью проникновения в нее. Однако в большинстве случаев они проникают на компьютеры вместе с вирусом либо червем - то есть такие трояны можно рассматривать как дополнительную вредоносную нагрузку, но не как самостоятельную программу. Нередко пользователи сами загружают троянские программы из Интернет. Жизненный цикл троянов состоит всего из трех стадий: - Проникновение в систему. - Активация. - Выполнение вредоносных действий. Как уже говорилось выше, проникать в систему трояны могут двумя путями - самостоятельно и в кооперации с вирусом или сетевым червем. В первом случае обычно используется маскировка, когда троян выдает себя за полезное приложение, которое пользователь самостоятельно копирует себе на диск (например, загружает из Интернет) и запускает. При этом программа действительно может быть полезна, однако наряду с основными функциями она может выполнять действия, свойственные трояну. Пример. Trojan.SymbOS.Hobble.a является архивом для операционной системы Symbian (SIS-архивом). При этом он маскируется под антивирус Symantec и носит имя symantec.sis. После запуска на смартфоне троян подменяет оригинальный файл оболочки FExplorer.app на поврежденный 165 файл. В результате при следующей загрузке операционной системы большинство функций смартфона оказываются недоступными. Для проникновения на компьютер, трояну необходима активация и здесь он похож на червя - либо требует активных действий от пользователя или же через уязвимости в программном обеспечении самостоятельно заражает систему. Поскольку главная цель написания троянов - это производство несанкционированных действий, они классифицируются по типу вредоносной нагрузки. 1. Клавиатурные шпионы, постоянно находясь в оперативной памяти, записывают все данные, поступающие от клавиатуры с целью последующей их передачи своему автору. 2. Похитители паролей предназначены для кражи паролей путем поиска на зараженном компьютере специальных файлов, которые их содержат. Пример. Trojan-PSW.Win32.LdPinch.kw собирает сведения о системе, а также логины и пароли для различных сервисов и прикладных программ - мессенджеров, почтовых клиентов, программ дозвона. Часто эти данные оказываются слабо защищены, что позволяет трояну их получить и отправить злоумышленнику по электронной почте. 3. Утилиты скрытого удаленного управления - это трояны, которые обеспечивают несанкционированный удаленный контроль над инфицированным компьютером. Перечень действий, которые позволяет выполнять тот или иной троян, определяется его функциональностью, заложенной автором. Обычно это возможность скрыто загружать, отсылать, запускать или уничтожать файлы. Такие трояны могут быть использованы как для получения конфиденциальной информации, так и для запуска вирусов, уничтожения данных. Пример. Backdoor.Win32.Netbus.170 предоставляет полный контроль над компьютером пользователя, включая выполнение любых файловых операций, загрузку и запуск других программ, получение снимков экрана и т. д. 4. Люки (backdoor) — трояны предоставляющие злоумышленнику ограниченный контроль над компьютером пользователя. От утилит удаленного управления отличаются более простым устройством и, как следствие, небольшим количеством доступных действий. Тем не менее, обычно одними из действий являются возможность загрузки и запуска любых файлов по команде злоумышленника, что позволяет при необходимости превратить ограниченный контроль в полный. 166 Пример. Троян Backdoor.win32.Wootbot.gen использует IRC-канал для получения команд от «хозяина». По команде троян может загружать и запускать на выполнение другие программы, сканировать другие компьютеры на наличие уязвимостей и устанавливать себя на компьютеры через обнаруженные уязвимости. 5. Анонимные SMTP-сервера и прокси-сервера - разновидность троянов, которые на зараженном компьютере организовывают несанкционированную отправку электронной почты, что часто используется для рассылки спама. Пример. Трояны из семейства Trojan-Proxy.Win32.Mitglieder распространяются с различными версиями червей Bagle. Троян запускается червем, открывает на компьютере порт и отправляет автору вируса информацию об IP-адресе зараженного компьютера. После этого компьютер может использоваться для рассылки спама. 6. Утилиты дозвона - в скрытом от пользователя режиме инициируют подключение к платным сервисам Интернет. 7. Модификаторы настроек браузера меняют стартовую страницу в браузере, страницу поиска или еще какие-либо настройки, открывают дополнительные окна, имитируют нажатия на рекламные баннеры и т. п. 8. Логические бомбы характеризуются способностью при срабатывании заложенных в них условий (в конкретный день, время суток, определенное действие пользователя или команды извне) выполнять какоелибо действие, например, удаление файлов. Пример. Virus.Win9x.CIH, Macro.Word97.Thus Отдельно отметим, что существуют программы из класса троянов, которые наносят вред другим, удаленным компьютерам и сетям, при этом не нарушая работоспособности инфицированного компьютера. Яркие представители этой группы - организаторы DDoS-атак. 13.5.4 Другие вредоносные программы Среди множества других вредоносных программ, для которых нельзя привести общий критерий, можно выделить следующие небольшие группы. 1. Условно опасные программы, то есть такие, о которых нельзя однозначно сказать, что они вредоносны. Такие программы обычно становятся опасными только при определенных условиях или действиях пользователя. 167 К условно опасным программам относятся: - Riskware - вполне легальные программы, которые сами по себе не опасны, но обладают функционалом, позволяющим злоумышленнику использовать их с вредоносными целями. К riskware относятся обычные утилиты удаленного управления, которыми часто пользуются администраторы больших сетей, клиенты IRC, программы для загрузки файлов из Интернет, утилиты восстановления забытых паролей и другие. - Рекламные утилиты (adware) - условно-бесплатные программы, которые в качестве платы за свое использование демонстрируют пользователю рекламу, чаще всего в виде графических баннеров. После официальной оплаты и регистрации обычно показ рекламы заканчивается и программы начинают работать в обычном режиме. Проблема adware кроется в механизмах, которые используются для загрузки рекламы на компьютер. Кроме того, что для этих целей часто используются программы сторонних и не всегда проверенных производителей, даже после регистрации такие модули могут автоматически не удаляться и продолжать свою работу в скрытом режиме. - Pornware - к этому классу относятся утилиты, так или иначе связанные с показом пользователям информации порнографического характера. На сегодняшний день это программы, которые самостоятельно дозваниваются до порнографических телефонных служб, загружают из Интернет порнографические материалы или утилиты, предлагающие услуги по поиску и показу такой информации. Отметим, что к вредоносным программам относятся только те утилиты класса pornware, которые устанавливаются на компьютер пользователя несанкционированно - через уязвимость в операционной системы или браузера или при помощи троянов. Обычно это делается с целью насильственного показа рекламы платных порнографических сайтов или служб. 2. Хакерские утилиты - К этому виду программ относятся программы скрытия кода зараженных файлов от антивирусной проверки (шифровальщики файлов), автоматизации создания сетевых червей, компьютерных вирусов и троянских программ (конструкторы вирусов), наборы программ, которые используют хакеры для скрытного взятия под контроль взломанной системы (RootKit) и другие подобные утилиты. То есть такие специфические программы, которые обычно используют только хакеры. 3. Злые шутки - программы, которые намеренно вводят пользователя в заблуждение путем показа уведомлений о, например, форматировании диска или обнаружении вирусов, хотя на самом деле ничего не происходит.

Признаки и диагностика заражений через браузер Явные проявления обычно заражений через Браузер выражаются в неожиданно появляющихся рекламных сообщениях и баннерах - обычно это следствие проникновения на компьютер рекламной утилиты. Поскольку их главная цель - это привлечь внимание пользователя к рекламируемой услуге или товару, то им сложно оставаться незаметными. Также явные проявления могут вызывать ряд троянских программ, например утилиты несанкционированного дозвона к платным сервисам. Они вынуждены быть явными, поскольку используемые ими приложения сложно использовать незаметно от пользователя. 14.1.2 Подозрительные процессы Одним из основных проявлений вредоносных программ является наличие в списке запущенных процессов (в ОС семейства Windows вызывается через CTRL+ALT+DEL, рис 14.1) подозрительных программ. Исследуя этот список и особенно сравнивая его с перечнем процессов, которые были запущены на компьютере сразу после установки системы, то есть до начала работы, можно сделать достаточно достоверные выводы об инфицировании. Это часто помогает при обнаружении вредоносных программ, имеющих лишь только скрытые или косвенные проявления. Рис. 14.1 – Просмотр запущенных в системе Windows процессов 182 14.1.3 Сетевая активность Неожиданно возросшая сетевая активность может служить ярким свидетельством работы на компьютере подозрительный программы. Но при этом нужно не забывать, что ряд вполне легальных приложений также имеют свойство иногда связываться с сайтом фирмы-производителя, например для проверки наличия обновлений или более новых версий. Поэтому, прежде чем отключать сеть необходимо уметь определять какие программы и приложения вызвали эту подозрительную активность. Рис. 14.3 Рис. 14.2 Рис. 14.4 Изучить и проанализировать сетевую активность можно с помощью встроенных в операционную систему инструментов или же воспользовавшись специальными отдельно устанавливаемыми приложениями. В этом задании это предлагается сделать с помощью Диспетчера задач Windows (рис. 14.2) и встроенной утилиты netstat (рис. 14.3, 14.4), которая выводит на экран мгновенную статистику сетевых соединений. 14.1.4 Элементы автозапуска Для того, чтобы прикладная программа начала выполняться, ее нужно запустить. Следовательно, и вирус нуждается в том, чтобы его запустили. 183 Оптимальным с точки зрения вируса вариантом служит запуск одновременно с операционной системой - в этом случае запуск практически гарантирован. Вредоносная программа может вносить изменения в системные файлы win.ini и system.ini. Следует также отметить, что в файле system.ini кроме секции [boot] вредоносные программы могут использовать секцию [Drivers]. Вредоносные программы могут вносить изменения в следующие ветки реестра: - HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows\CurrentVersion в ключи Run, RunOnce, RunOnceEx, RunServices, RunServicesOnce - для того чтобы система запускала созданные червем файлы - HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion в ключ Run. Кроме выше перечисленных ветвей и ключей реестра вредоносные программы могут вносить изменения и в другие ветки и ключи реестра, например: - HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\WOW\boot - HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\Drivers32 - HKEY\_LOCAL\_MACHINE\Software\Microsoft\Windows NT\ CurrentVersion\WinLogon - HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services - HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Active Setup\ Installed Components - HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows NT\ CurrentVersion\AeDebug Диагностика элементов автозапуска возможно путем изучения секций: SYSTEM.INI (рис. 14.5); WIN.INI (рис. 14.6); «Автозапуск» (рис. 14.7) и «Службы» (рис. 14.8), в утилите конфигурирования msconfig. Удаление запускаемого вируса из автозагрузке возможно путем использования утилиты regedit. Рис. 14.5 Рис. 14.6 184 Рис. 14.7 Рис. 14.8 14.2 Основы функционирования антивирусного программного обеспечения Антивирус - программное средство, предназначенное для борьбы с вирусами. Основными задачами антивируса является: - Препятствование проникновению вирусов в компьютерную систему. - Обнаружение наличия вирусов в компьютерной системе. - Устранение вирусов из компьютерной системы без нанесения повреждений другим объектам системы. - Минимизация ущерба от действий вирусов. 14.2.1 Технологии обнаружения вирусов Технологии, применяемые в антивирусах, можно разбить на две группы 1. Технологии сигнатурного анализа. 2. Технологии вероятностного анализа: 2.1.Эвристический анализ. 2.2.Поведенческий анализ. 2.3.Анализ контрольных сумм. Сигнатурный анализ - метод обнаружения вирусов, заключающийся в проверке наличия в файлах сигнатур вирусов. Сигнатурный анализ является наиболее известным методом обнаружения вирусов и используется практически во всех современных антивирусах. Для проведения проверки антивирусу необходим набор вирусных сигнатур, который хранится в антивирусной базе. Ввиду того, что сигнатурный анализ предполагает проверку файлов на наличие сигнатур вирусов, антивирусная база нуждается в периодическом обновлении для поддержания актуальности антивируса. 185 Недостатки сигнатурного анализа определяют границы его функциональности - возможность обнаруживать лишь уже известные вирусы - против новых вирусов сигнатурный сканер бессилен. Достоинством сигнатурного анализа является то, что наличие сигнатур вирусов предполагает возможность лечения инфицированных файлов, обнаруженных при помощи сигнатурного анализа. Однако, лечение допустимо не для всех вирусов - трояны и большинство червей не поддаются лечению по своим конструктивным особенностям, поскольку являются цельными модулями, созданными для нанесения ущерба. Грамотная реализация вирусной сигнатуры позволяет обнаруживать известные вирусы со стопроцентной вероятностью. Эвристический анализ - технология, основанная на вероятностных алгоритмах, результатом работы которых является выявление подозрительных объектов. В процессе эвристического анализа проверяется структура файла, его соответствие вирусным шаблонам. Наиболее популярной эвристической технологией является проверка содержимого файла на предмет наличия модификаций уже известных сигнатур вирусов и их комбинаций. Достоинством эвристического анализа является то, что он может определять гибриды и новые версии ранее известных вирусов без дополнительного обновления антивирусной базы. Недостатком – то, что эвристический анализ не предполагает лечения. Данная технология не способна на 100% определить вирус перед ней или нет, и как любой вероятностный алгоритм грешит ложными срабатываниями. Поведенческий анализ - технология, в которой решение о характере проверяемого объекта принимается на основе анализа выполняемых им операций. Поведенческий анализ весьма узко применим на практике, так как большинство действий, характерных для вирусов, могут выполняться и обычными приложениями. Наибольшую известность получили поведенческие анализаторы скриптов и макросов, поскольку соответствующие вирусы практически всегда выполняют ряд однотипных действий. Помимо этого поведенческие анализаторы могут отслеживать попытки прямого доступа к файлам, внесение изменений в загрузочную запись дискет, форматирование жестких дисков и т. д. Поведенческие анализаторы не используют для работы дополнительных объектов, подобных вирусным базам и, как следствие, неспособны различать известные и неизвестные вирусы - все подозрительные программы априори считаются неизвестными вирусами. Аналогично, особенности работы средств, реализующих технологии поведенческого анализа, не предполагают лечения. Например, средства защиты, вшиваемые в BIOS, также можно отнести к поведенческим анализаторам. При попытке внести изменения в MBR компьютера, анализатор блокирует действие и выводит соответствующее уведомление пользователю. 186 Анализ контрольных сумм - это способ отслеживания изменений в объектах компьютерной системы. На основании анализа характера изменений - одновременность, массовость, идентичные изменения длин файлов - можно делать вывод о заражении системы. Анализаторы контрольных сумм (также используется название «ревизоры изменений») как и поведенческие анализаторы не используют в работе дополнительные объекты и выдают вердикт о наличии вируса в системе исключительно методом экспертной оценки. Чаще подобные технологии применяются в сканерах при доступе - при первой проверке с файла снимается контрольная сумма и помещается в кэше, перед следующей проверкой того же файла сумма снимается еще раз, сравнивается, и в случае отсутствия изменений файл считается незараженным. 14.2.2 Классификация антивирусного программного обеспечения Помимо используемых технологий, антивирусы отличаются друг от друга условиями эксплуатации. Уже из анализа задач можно сделать вывод о том, что препятствование проникновению вредоносного кода должно осуществляться непрерывно, тогда как обнаружение вредоносного кода в существующей системе - скорее разовое мероприятие. Следовательно, средства, решающие эти две задачи должны функционировать по-разному. Таким образом, антивирусы можно разделить на две большие категории: - Предназначенные для непрерывной работы - к этой категории относятся средства проверки при доступе, почтовые фильтры, системы сканирования проходящего трафика Интернет, другие средства, сканирующие потоки данных. - Предназначенные для периодического запуска - различного рода средства проверки по запросу, предназначенные для однократного сканирования определенных объектов. К таким средствам можно отнести сканер по требованию файловой системы в антивирусном комплексе для рабочей станции, сканер по требованию почтовых ящиков и общих папок в антивирусном комплексе для почтовой системы (в частности, для Microsoft Exchange). Антивирусный комплекс - набор антивирусов, использующих одинаковое антивирусное ядро или ядра, предназначенный для решения практических проблем по обеспечению антивирусной безопасности компьютерных систем. В антивирусный комплекс также в обязательном порядке входят средства обновления антивирусных баз. Антивирусное ядро - реализация механизма сигнатурного сканирования и эвристического анализа на основе имеющихся сигнатур вирусов. 187 Исходя из текущей необходимости в средствах защиты выделяют следующие типы антивирусных комплексов: 1. Антивирусный комплекс для защиты рабочих станций - предназначен для обеспечения антивирусной защиты рабочей станции, на которой он установлен. Состоит, как и указывалось ранее из средств непрерывной работы и предназначенных для периодического запуска, а также средств обновления антивирусных баз. 2. Антивирусный комплекс для защиты файловых серверов - предназначен для обеспечения антивирусной защиты сервера, на котором установлен. Указание на файловый сервер в названии является скорее данью истории, корректней будет звучать термин «сетевой». Определение того, насколько нуждается в антивирусной защите сервер, осуществляется не только исходя из его назначения (является сервер файловым, почтовым, либо выполняет другую функцию), а и из используемой на нем платформы. 3. Антивирусный комплекс для защиты почтовых систем, назаначение комплекса назначение - препятствовать доставке зараженных сообщений пользователям сети, но он не предназначен для защиты почтовой системы от поражения вирусами. Как уже указывалось ранее, сегодня одним из главных средств доставки вирусов в локальную сеть является именно электронная почта. Поэтому, при наличии в локальной сети специализированного узла, обрабатывающего входящую и исходящую из сети почтовую корреспонденцию (почтового сервера), логично будет использовать средство централизованной проверки всего почтового потока на наличие вирусов) 4. Антивирусный комплекс для защиты шлюзов - предназначен для проверки на наличие вирусов данных, через этот шлюз передаваемых. Как правило в его состав входят: 4.1.Сканер HTTP-потока — предназначен для проверки данных, передаваемых через шлюз по протоколу HTTP. 4.2.Сканер FTP-потока — предназначен для проверки данных, передаваемых через шлюз по протоколу FTP. В случае использования FTP over HTTP FTP-запросы будут проверяться сканером HTTPпотока. 4.3.Сканер SMTP-потока — предназначен для проверки данных, передаваемых через шлюз по SMTP