**Лекции №15: Применение шифрования**

**Цель:** ознакомить студентов с принципами и методами шифрования данных в операционных системах Windows и веб-сервере IIS. Рассмотреть различные способы защиты информации с помощью шифрования, объяснить важность шифрования для обеспечения конфиденциальности и целостности данных, а также научить использовать эти методы в реальных приложениях.

В *Windows* и *IIS* реализовано несколько методов шифрования.

* Аутентификация с помощью Kerberos или других поддерживаемых методов (см. ["Приложение D. Методы аутентификации Microsoft IIS"](https://www.intuit.ru/studies/courses/1002/122/lecture/24710)).
* Шифрование файлов в разделе или на жестком диске NTFS.
* Виртуальные частные сети (VPN) с использованием протоколов *PPTP*, IPSec или *L2TP*.
* Использование шифрования SSL или следующего поколения стандарта – TLS.

*Шифрование*, используемое в аутентификации *Windows 2000*, должно контролироваться администратором. *Аутентификация* *Windows 2000* представляет собой самодостаточную систему, и вам необходимо только настроить параметры аутентификации на сервере либо изменить эффективность шифрования в реестре. *По* умолчанию Microsoft в настоящее время использует 128-битное *шифрование*, что подходит для большинства *интернет*-систем; *шифрование* с большей степенью требуется только для специализированных приложений, связанных с безопасностью.

*Шифрование* файлов *NTFS* (метод защиты данных, применяемый в разделах *NTFS*) можно настроить под ваши нужды. Во многих организациях *шифрование* *NTFS* используется на файловых серверах *интранет* для усиления контроля при работе с документами высокой степени секретности. Не следует использовать этот метод на сайте с большим разделом, так как *шифрование* абсолютно всех данных сайта сильно замедляет его работу даже при наличии мощной аппаратной платформы. *Шифрование* файлов используется, как правило, для отдельных элементов данных, например, для номеров кредитных карт в базе данных клиента. Во многих случаях данные хранятся и контролируются на другом сервере, и *база данных* содержит свой собственный *алгоритм* шифрования для обеспечения *защиты файла*.

На сервере *Windows 2000* можно включить поддержку *VPN*, для чего необходимо настроить его на маршрутизацию и удаленный *доступ*. В ["Подготовка и укрепление веб-сервера"](https://www.intuit.ru/studies/courses/1002/122/lecture/3521)настоятельно рекомендовалось отключить маршрутизацию и удаленный *доступ* на сервере, так как эти функции представляют угрозу несанкционированного проникновения злоумышленника в *сеть*. Виртуальная частная *сеть* *VPN*, используемая для удаленного управления сервером, должна располагаться на отдельном сервере для *Windows 2000*, либо следует использовать сетевой экран или аппаратное оборудование *VPN*. (См. ["Особенности процесса разработки"](https://www.intuit.ru/studies/courses/1002/122/lecture/3527), ["Сторонние средства обеспечения безопасности"](https://www.intuit.ru/studies/courses/1002/122/lecture/3533))

На веб-сайте чаще всего используется *шифрование* *SSL*/*TLS* при отправке данных. *SSL* применяется как в интернете, так и в сетях *интранет*, если требуется соблюдение неприкосновенности и конфиденциальности информации. В данной лекции мы обсудим применение и настройку шифрования *IIS* *SSL*/*TLS* для защиты веб-сайтов.

### Основы шифрования

Вначале рассмотрим несколько важных принципов шифрования. *Шифрование* включает в себя *кодирование* данных, чтобы их мог прочитать только определенный получатель. *Шифрование* – это древняя наука, первые сведения о ней относятся еще к ХХ веку до нашей эры (Египет). Римляне использовали *шифрование* сообщений в процессе обмена информацией между войсковыми подразделениями. Сегодня системы шифрования широко используются для передачи секретных цифровых данных через *компьютерные сети*.

#### Ключи и шифры

Системы шифрования делают данные нечитабельными и, наоборот, доступными для чтения с использованием алгоритмов, называемых шифрами. Шифр представляет собой определенный метод кодирования и декодирования данных. Во времена Юлия Цезаря римляне использовали простой шифр со смещением букв алфавита. Для расшифровки римского сообщения получателю требовался ключ; т.е. ему необходимо было знать, на какое расстояние смещены буквы алфавита и в каком направлении. В настоящее время в системах шифрования используются сложные математические формулы и большие двоичные числа, но принцип остается все тем же.

Компьютеризированные системы шифрования создаются таким образом, чтобы шифруемые данные с использованием внутреннего ключа обрабатывались программой, называемой *системой шифрования*. Как правило, для расшифровки данных компьютеру требуются две вещи: шифр и ключ. В Windows 2000 и IIS система шифрования встроена в саму операционную систему. Это позволяет службам и приложениям операционной системы выполнять шифрование в реальном времени. Например, если сервер Windows 2000 настроен на шифрование файлов в папке, шифрование происходит при сохранении файла. Если сервер настроен на работу с VPN или использует протоколы SSL/TLS, данные шифруются в момент отправки их в сеть и расшифровываются после получения.

#### Шифрование на симметричном (секретном) ключе

Исторически сложилось так, что системы шифрования являются системами на симметричных *(секретных) ключах*. Система шифрования на симметричном ключе использует один и тот же ключ как для зашифровки, так и для расшифровки данных. Когда римляне отправляли сообщения, зашифрованные посредством смещения алфавита, ключом была величина смещения и его направление. Используя эту информацию, получатель мог расшифровать сообщение посредством сдвига букв сообщения в обратном направлении для размещения их на исходных местах. В такой системе ключом являлось число (меньшее, чем число букв в римском алфавите) и направление (влево или вправо). Недостатком данной системы шифрования было то, что легко угадывался ключ и раскрывался текст сообщения.

Современные компьютеризированные системы шифрования позволяют использовать очень длинные ключи. Если стороннее лицо узнает алгоритм шифрования, то для угадывания ключа длиной 128 бит понадобится очень много времени, даже при использовании компьютеризированных методов, так как количество возможных комбинаций равно 3,4 x 1038. Поэтому шифры, используемые в промышленности и в государственных стандартах безопасности, открыты для всеобщего доступа и хорошо известны. Это может показаться слишком легкомысленным, однако открытое распространение шифров, на самом деле, предназначено для повышения их защищенности, так как многие специалисты в области шифрования имеют возможность оценки, дополнения и совершенствования шифров. Примерами хорошо известных систем шифрования, алгоритмы которых доступны любому желающему, являются DES, Triple DES и новый стандарт AES. Некоторые коммерческие *симметричные алгоритмы*, такие как *RC4*, сделаны общедоступными для повышения надежности посредством проверки алгоритмов пользователями. Ниже приведен перечень современных симметричных шифров.

* **DES** – стандарт шифрования (*Data Encryption Standard*), используемый правительством США.
* **Skipjack** – секретный алгоритм симметричного ключа, применяемый в оборудовании, совместимом с *FORTEZZA* (применяется в правительстве США).
* **Triple-DES** – стандарт DES, применяемый трижды (для повышения сложности взлома).
* **RC2, RC4, RC5 и RC6** – эти коммерческие шифры создаются и лицензируются компанией RSA Security, Inc. Microsoft использует *RC4* по умолчанию в протоколах *PPTP* и *L2TP*, SSL/TLS и при поддержке протокола IPSec (можно изменить в системном реестре).
* **AES** – стандарт AES является новым стандартом обработки информации, который применяется для определения криптографического алгоритма в организациях правительства США для защиты важной (секретной) информации.

**Важно**. Стандарт DES разработан в начале 1970-х годов. Он десятилетиями использовался для шифрования финансовых транзакций между банками и финансовыми институтами. Сейчас этот стандарт устарел, так как длина его ключа ограничена 56 битами. Новые стандарты не имеют этого ограничения. Используя мощь современных микропроцессоров, 56-битный ключ DES может быть взломан за несколько часов, в то время как на взлом 128-битного ключа потребуются многие месяцы работы самых мощных компьютеров. Вывод: не следует использовать алгоритмы шифрования с ключами, длина которых меньше 128 бит.

Преимущество шифрования на симметричном ключе заключается в быстрой и эффективной работе, что делает его подходящим для приложений, требующих шифрования в реальном времени, в отличие от других методов, отрицательно влияющих на производительность систем.

Недостатком шифрования с использованием симметричного ключа является то, что ключи должны согласовываться между отправителем и получателем заранее, т.е. им необходимо договориться о ключах. При обмене ключами нужно соблюдать особые меры предосторожности, так как если ключи станут известны третьему лицу, то он легко расшифрует текст. Если количество получателей текста невелико, этот процесс можно осуществить с относительной легкостью, но при увеличении числа получателей его сложность возрастает в геометрической прогрессии. Следовательно, несмотря на возможность автоматизации, обмен ключами является очень ответственным процессом на веб-сайтах, с которыми работает большое число клиентов.

#### Шифрование на открытом ключе (ассиметричное шифрование)

В 1970-х годах появилась новая система шифрования, называемая *шифрованием на ассиметричном (открытом) ключе*. Она называется ассиметричной, потому что не требует использования идентичных ключей отправителем и получателем шифрованного сообщения. Она является системой с открытым ключом, так как один из ключей не содержится в секрете.

Давайте остановимся на этом поподробнее. Шифрование на открытом ключе использует два различных ключа, составляющих пару, но не идентичных. В шифровании с симметричным ключом каждый ключ является уникальным. Пара ключей открытый/секретный работает сообща: один ключ предназначен для шифрования данных, а другой – для расшифровки, и наоборот. Секретный ключ должен содержаться в секретности в целях безопасности, а открытый ключ может передаваться по небезопасному соединению без угрозы для системы. Следовательно, система шифрования на открытом ключе решает одну из главных проблем старых систем шифрования, заключающуюся в безопасном способе передачи ключа шифрования другой стороне.

Как правило, открытые ключи используются только для зашифровки данных. Расшифровать их сможет только тот пользователь, чей компьютер содержит соответствующий секретный ключ. Эта система построена на математических принципах, используемых в шифрах с открытыми ключами и обеспечивающих существование одного и только одного уникального секретного ключа, соответствующего уникальному открытому ключу. Следовательно, если выполняется шифрование данных пользователя на общем ключе, можете быть уверены, что только пользователь, владеющий второй, секретной, половиной ключа, сможет их расшифровать.

Первым коммерческим алгоритмом шифрования на открытом ключе был *алгоритм RSA* (сокращение представляет собой первые буквы фамилий трех специалистов, которые разработали шифр и впоследствии создали свою собственную компанию RSA Security, Inc.). Данный алгоритм использовался в Netscape в качестве компонента первой версии SSL (это был единственный шифр, используемый в Netscape), который в итоге фактически стал частью стандарта, когда Netscape открыли SSL для общего пользования. Microsoft изначально использовал шифрование RSA в операционной системе Windows NT, оно используется и в Windows 2000. Ключи RSA являются основными возможностями шифрования в Windows 2000/IIS.

#### Комбинирование методов шифрования

Несмотря на то, что шифрование на открытом ключе имеет преимущество с точки зрения безопасности обмена ключами, его недостатком является скорость работы. Системы шифрования с ассиметричными ключами работают гораздо медленнее, чем системы с симметричными ключами. Они применяются только для шифрования небольших объемов данных, их не рекомендуется использовать в случае больших объемов информации при шифровании в реальном времени в сеансах безопасного соединения.

Стандарты SSL/TLS и IPSec, применяемые для шифрования данных в режиме реального времени в интернете, комбинируют алгоритмы ассиметричного и симметричного шифрования для использования преимуществ каждого. Например, на практике шифрование на открытом ключе не используется для шифрования данных в сессии безопасного соединения, так как оно работает медленно в сравнении с шифрованием на секретном ключе такой же длины.

Вместо этого для аутентификации, шифрования и отправки ключа для симметричного шифра *RC4* или AES (объем данных относительно невелик) используется шифрование с открытым ключом. Симметричный шифр используется для шифрования данных сеанса (данные, объем которых измеряется мегабайтами). На принимающей стороне программное обеспечение сначала расшифровывает симметричный ключ, после чего использует его для расшифровки полезных данных.

Комбинирование методов шифрования повышает общий уровень безопасности системы шифрования. Посредством ассиметричного шифрования для доставки *ключей сессии* симметричного шифрования система может использовать отдельный симметричный ключ для каждого сеанса. Разумеется, это можно делать как при помощи SSL, так и IPSec. Программное обеспечение выбирает симметричный ключ случайным образом. Таким образом, раскрытие одного *ключа сеанса* связи (что происходит крайне редко) не влияет на следующую установку сеанса.

#### Цифровые подписи и инфраструктура открытого ключа

Шифрование на общем ключе привело к введению цифровых сертификатов, используемых для аутентификации на веб-сайтах с протоколами SSL/TLS и IPSec. *Цифровой сертификат* представляет собой цифровой документ (небольшой файл), заверяющий подлинность и статус владельца для пользователя или компьютерной системы. Например, бизнес-сертификат подтверждает тот факт, что компания обладает определенным открытым ключом. Цифровые сертификаты помогают автоматизировать распределение открытых ключей в протоколе шифрования с открытым ключом. Когда другому компьютеру необходимо произвести обмен данными с вашей системой, он осуществляет доступ к цифровому сертификату, содержащему ваш открытый ключ.

Набор продуктов и процессов, необходимых для безопасного создания, управления и распределения цифровых сертификатов, называется *инфраструктурой открытого ключа* (*PKI*). Одним из компонентов *PKI* является компьютер, называемый *сервером сертификатов*. Объединение, включающее *сервер сертификатов* и создающее сертификаты, называется *бюро сертификатов* (CA). CA несет ответственность за подтверждение подлинности сертификата и принадлежности его физическому лицу или организации перед созданием сертификата.

Существуют компании, например, Verisign и SSL.com, являющиеся коммерческими CA. Эти организации за определенную плату выпускают сертификаты для отдельных лиц и компаний. Если организация взяла на себя роль своего собственного бюро сертификатов и выпускает свои сертификаты, то необходим программный продукт Microsoft Sertificate Server, но придется установить этот компонент и управлять им на отдельном сервере в целях безопасности. Руководство многих компаний предпочитает оплатить услугу по получению сертификатов вместо покупки своего собственного оборудования и выделения необходимых ресурсов для самостоятельного *выпуска сертификатов*.

Стандарт цифровых сертификатов X.509 обеспечивает совместимость с протоколами SSL/TLS и IPSec, используемыми в Microsoft Windows 2000 и IIS. Согласно этому стандарту цифровой *сертификат X.509 v3* должен содержать четыре объекта.

* *Отличительное имя* (Distinguished Name, *DN*) организации, от которой получен сертификат (т.е. имя, введенное в поле Name сертификата).
* Открытый ключ отдельного лица или организации, идентифицируемый сертификатом.
* Цифровую подпись, полученную от секретного ключа бюро сертификатов, предусмотренного соответствующим *сервером сертификатов*.
* Отметки о дате, означающие даты выпуска и срок действия сертификата.

Имея эту информацию, два узла в виртуальной частной сети или веб-сервер и правильно настроенный веб-браузер могут отправлять и получать потоки данных, которые будут расшифрованы только ими.

Установление доверия к бюро сертификатов осуществляется по решению персонала или руководства компании. В интернете, как правило, принимается сертификат Verisign. После вынесения решения о доверии корневой сертификат СА нужно установить на серверах и клиентах, осуществляющих взаимную аутентификацию. Корневым сертификатом называется сертификат, содержащий открытый ключ CA, которому будут сопоставляться отдельно выпущенные и подписанные сертификаты с применением концепции открытого ключа для подтверждения подлинности сертификатов. Например, браузер авторизует цифровой сертификат веб-сайта, сопоставляя подпись сертификата с открытым ключом корневого сертификата CA, установленного в браузере. С помощью этой проверки браузер определяет, что сайт на самом деле принадлежит компании или организации, которую он представляет (в силу доверия к бюро сертификатов).

**Вопросы для закрепления:**

1. Что такое шифрование и зачем оно важно для защиты данных?
2. Какие методы аутентификации поддерживает IIS, и какова роль шифрования в этих методах?
3. Объясните, как работает шифрование NTFS и в каких случаях оно используется.
4. В чем заключаются преимущества и недостатки использования VPN в Windows 2000?
5. Чем шифрование SSL/TLS отличается от других методов шифрования, и как оно используется для защиты веб-сайтов?
6. Что такое симметричное шифрование, и какие преимущества оно имеет для защиты данных?
7. Какой стандарт шифрования применяется в современных системах, и почему алгоритмы с ключами менее 128 бит не рекомендуется использовать?
8. Что такое шифрование на открытом ключе (ассиметричное шифрование), и в чем его основное отличие от симметричного шифрования?
9. Какова роль цифровых подписей и инфраструктуры открытого ключа (PKI) в обеспечении безопасности данных?
10. Что такое цифровой сертификат X.509, и какие объекты он должен содержать для подтверждения подлинности владельца?