

ТЕМА 5. ОРГАНИЗАЦИЯ БЕЗОПАСНОСТИ В КОРПОРАТИВНЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ

В данной теме рассматриваются следующие вопросы:

- аутентификация, авторизация и аудит;
- маркер защиты пользователя;
- списки контроля доступа (ACL);
- утверждения;
- служба сертификации Active Directory Certification Services;
- применение сертификатов X509.3;
- аутентификация в IIS.

Лекции – 2 часа, лабораторные занятия – 2 часа, самостоятельная работа – 6 часов.

Минимальный набор знаний:

Понятия аутентификации, авторизации, аудита

Понятие SID и RID

UAC: что означает флаг Deny в маркере защиты администратора?

Списки контроля доступа (ACL)

Сертификаты X509.3 (структура, основные поля, цепочки доверия, СОС)

Шаблоны сертификатов

ОГЛАВЛЕНИЕ

5.1. Аутентификация, авторизация и аудит

5.1.1. Аутентификация

5.1.2. Авторизация

5.1.2. Аудит

5.2. Подсистема безопасности Windows

5.2.1. маркер защиты пользователя;

5.2.2. списки контроля доступа (ACL);

5.2.3. Определение уровня доступа

5.2.4. Утверждения;

5.3. Инфраструктура открытого ключа

5.3.1. Сертификаты X509.3.

5.3.2. Построение цепочки сертификатов

5.3.3. Обзор возможностей службы Active Directory Certification Services

5.4. Методы аутентификации в службе Internet Information Services

5.1. Аутентификация, авторизация и аудит

5.1.1. Аутентификация

В основе информационной защиты лежит одна очень важная концепция *идентичности* (identity).

В защищенной системе идентификатор представляет каждого пользователя. Идентичность в компьютерном мире представляет собой набор данных, который однозначно описывает человека или объект, иногда называемый субъектом или сущностью, и содержит информацию о взаимоотношениях субъекта с другими объектами. В операционной системе Windows пользователь учетной записи представляет личность человека или услуги. Хранилище идентификаторов поддерживает учетные записи для одного или нескольких пользователей, который также известен как база данных каталога. AD DS — один из примеров базы данных каталога. В доменных службах Active Directory идентификация обычно представляется с помощью принципа безопасности. Принципы безопасности идентифицируются однозначно с помощью атрибута, называемого идентификатором безопасности (SID).

Одной из самых больших проблем в глобальной цифровой связи является надежность идентификации; нет надежного и надежного способа подтвердить чью-либо цифровую идентификацию. Даже если атрибуты связаны с цифровой идентичностью человека, эти атрибуты или даже идентификаторы могут быть изменены, замаскированы или удалены, и могут быть созданы новые атрибуты или идентификаторы. Чтобы назначить цифровое представление сущности, сторона, присваивающая атрибут, должна доверять тому, что требование атрибута, такое как имя, местоположение, роль как сотрудника или возраст, является правильным и связано с человеком или предмет, представляющий этот атрибут.

На другой стороне системы находится ресурс, к которому пользователь требует доступа. Разрешения (permissions) защищают ресурс, и каждое разрешение определяет сопряжение определенного уровня доступа с идентификатором. Многие ресурсы на базе Windows, включая файлы и папки на томах файловой системы NTFS, защищены дескриптором безопасности, который содержит дискреционный список управления доступом (DACL — Discretionary Access Control List), в котором каждое разрешение принимает форму записи управления доступом (ACE — Access Control Entry).

Аутентификация — это процесс проверки личности пользователя, компьютера, группы, устройства, службы или процесса. В реальной жизни процесс аутентификации происходит очень часто. Например, при международном перелете вы должны представить свой паспорт в аэропорту для пограничного контроля. Делая это, вы выполняете процесс аутентификации. Сотрудник пограничной службы доверяет организации, выдавшей ваш паспорт. Таким образом, вы удостоверяете свою личность надежным способом. Однако предъявление паспорта по-прежнему не дает вам права доступа к самолету, являющемуся ресурсом в этом примере.

В компьютерных средах аутентификация обычно состоит из предоставления набора учетных данных. Пользователь предоставляет учетные данные,

содержащие как минимум два компонента: имя входа и секрет, известный только пользователю и системе, например, пароль. Система сравнивает предъявленные учетные данные со своей информацией. Помимо этого типа аутентификации, пользователь также может удостоверить свою личность, используя смарт-карты, токены или сканирование отпечатков пальцев или сетчатки. Выполнение аутентификации с использованием нескольких факторов, таких как смарт-карта с PIN-кодом, делает аутентификацию более безопасной и надежной.

Существует два типа аутентификации компьютера: локальная и удаленная. Локальный или интерактивный вход в систему происходит, когда пользователь входит в систему непосредственно на компьютере. Удаленный или сетевой вход в систему происходит при подключении к другому компьютеру (например, файловому или почтовому серверу) для получения файлов или других типов ресурсов.

В бизнес-средах, использующих операционную систему Windows Server 2012, пользователи и компьютеры аутентифицируются в отношении доменных служб Active Directory, используя в основном аутентификацию протокола Kerberos версии 5. Аутентификация Kerberos работает на основе билетов, которые позволяют компьютерам, обменивающимся по сети, доказать свою личность друг другу безопасным образом. Он разработан в основном для архитектуры клиент/сервер и обеспечивает взаимную аутентификацию — как пользователь, так и сервер проверяют личность друг друга. Помимо аутентификации Kerberos, операционная система Windows также может работать с протоколом проверки подлинности NTLM, а в некоторых случаях может даже использовать обычную проверку подлинности. Хотя NTLM считается безопасным методом аутентификации, обычная проверка подлинности работает, отправляя имя пользователя и пароль в виде открытого текста. Мы не рекомендуем использовать обычную проверку подлинности для производственных сред, за исключением случаев, когда она дополнительно защищена протоколом Secure Sockets Layer (SSL).

В автономной конфигурации систем на базе Windows, также называемой рабочей группой, каждый компьютер поддерживает одно и только одно доверенное хранилище идентификаторов: локальный список пользователей и групп, хранящийся в реестре, который называется базой данных диспетчера учетных записей безопасности (SAM). В отличие от централизованной аутентификации в домене, в рабочей группе отсутствует система распределенной аутентификации, поскольку каждый компьютер имеет собственный SAM.

Но использование рабочих групп становится затруднительным даже в сетях с десятком компьютеров. Поэтому в корпоративных информационных системах используются централизованные решения.

Принимая во внимание то, что было сказано об аутентификации; давайте посмотрим на процесс присоединения компьютера к домену. Как было сказано ранее, пользователи могут получить доступ к локальным ресурсам на компьютере, выполнив вход на компьютер с помощью локальной учетной записи, и в этом случае пользователь аутентифицируется локальным SAM или с помощью доменной учетной записи из AD DS, и в этом случае пользователь аутентифицируется в AD DS. Пользователи могут войти в систему, используя

учетную запись домена, только если компьютер, к которому они подключаются, подключен к домену AD DS. Делая это, компьютер начинает доверять AD DS как надежной службе аутентификации, поэтому каждый доменный пользователь может получить доступ к локальным ресурсам на этом компьютере. Присоединение компьютера к домену фактически устанавливает процесс доверия между доменом AD DS и компьютером. Это очень похоже на установление доверия между доменами; когда вы устанавливаете доверие между двумя доменами, вы разрешаете пользователям из одного домена выполнять аутентификацию и доступ к ресурсам в другом домене. Присоединившись к компьютеру к домену, вы разрешаете пользователям из этого домена войти на этот компьютер и получить доступ к ресурсам этого компьютера.

5.1.2. Авторизация

После аутентификации пользователей им, вероятно, потребуется получить доступ к ресурсу на локальном компьютере или удаленном сервере. Как уже упоминалось ранее, аутентификация не гарантирует вам права доступа к ресурсу, так же, как представление вашего паспорта в аэропорту не дает вам права садиться на самолет. Пользователям необходимо выполнить дополнительный процесс, называемый **авторизацией**, для доступа к любому ресурсу. Вы не можете быть авторизованы без предварительной аутентификации; однако аутентификация не гарантирует успешного доступа к ресурсу. Авторизация — это процесс, который определяет, предоставлять или отклонять запрашиваемый пользователем уровень доступа к ресурсу.

После аутентификации пользователя локальная служба безопасности (LSA) генерирует маркер защиты (security token), который содержит идентификаторы безопасности SID пользователя и всех групп, к которым пользователь принадлежит. Кроме того, маркер защиты содержит привилегии и права пользователя на данном компьютере, например, право отключать систему или входить в систему в интерактивном режиме.

Важно помнить, что маркер защиты генерируется и остается локально на компьютере, который аутентифицировал пользователя. Когда пользователь регистрируется на свой компьютер локально или интерактивно, система создает маркер безопасности, и, если пользователь имеет право входа в систему в интерактивном режиме, происходит вызов процесса explorer.exe, который и отображает рабочий стол.

Когда пользователь подключается к серверу для доступа к файлу, сервер аутентифицирует пользователя и генерирует маркер защиты на сервере, который имитирует пользователя с идентификатором SID пользователя и SID всех групп, к которым принадлежит этот пользователь. Маркер доступа на сервере отличается от маркера защиты на рабочем столе пользователя. Маркер защиты никогда не передается по сети, а LSA операционной системы Windows никогда не будет принимать маркер защиты, сгенерированный другим LSA.

Конечно, это должно быть так, потому что пользователь, вероятно, принадлежит к различным локальным группам на сервере и на своем компьютере. И почти наверняка пользователь имеет другие привилегии и права пользователя на сервере.

Когда пользователь пытается получить доступ к ресурсу, делается запрос доступа, который указывает ресурс и уровень доступа пользователя, а затем создается маркер защиты, который представляет пользователя. Подсистема безопасности проверяет ACL ресурса, сравнивая SID в ACE с SID в маркере защиты. ACE, который соответствует как SID'у в маркере защиты, так и желаемому типу доступа, определяет, разрешен или запрещен доступ пользователю к ресурсу. Если совпадение не найдено, доступ запрещен. Если более одного ACE соответствует пользователю, применяются кумулятивные разрешения.

В Windows Server 2012 появился еще один тип авторизации, который может применяться к файлам и папкам. Вместо сравнения SID в маркере защиты с идентификаторами SID, указанными в ACE, пользователь может разрешить путем оценки значений определенных атрибутов пользователей. Этот метод авторизации называется Dynamic Access Control (DAC) и будет обсуждаться в пункте 5.2.6.

5.1.3. Аудит безопасности

У слова аудит имеется много различных значений, в рамках данной темы под аудитом будем понимать только регистрацию в журнале безопасности событий, имеющих отношение к безопасности информационной системы, например, регистрация в системе, получение доступа к объекту, изменение объекта и т.п.

Менеджер объектов может генерировать события аудита в результате проверки доступа, а функции Windows, доступные для пользовательских приложений, могут генерировать их напрямую. Код режима ядра всегда позволяет генерировать событие аудита. Две привилегии, SeSecurityPrivilege и SeAuditPrivilege, относятся к аудиту. Процесс должен иметь привилегию SeSecurityPrivilege для управления журналом событий безопасности и для просмотра или установки SACL объекта. Однако процессы, которые вызывают службы аудита, должны иметь привилегию SeAuditPrivilege для успешного создания записи аудита.

Политика аудита локальной системы контролирует решение о проверке определенного типа событий безопасности. Политика аудита, также называемая локальной политикой безопасности, является частью политики безопасности, которую LSASS поддерживает в локальной системе, и она настроена с помощью редактора политики локальной безопасности, как показано на рис. 5.1.

Конфигурация политики аудита (как базовые настройки в разделе «Локальные политики», так и расширенная конфигурация политики аудита, которые будут описаны ниже) хранится в реестре как битовое значение в ключе HKEY_LOCAL_MACHINE\SECURITY\Policy\PolAdtEv.

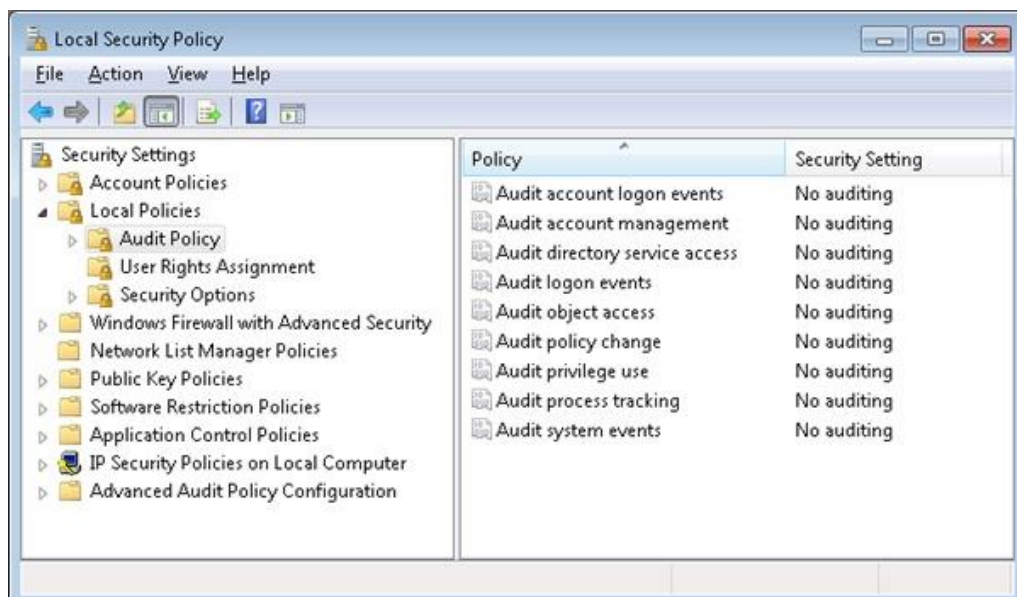


Рис. 5.1. Настройка политики аудита в редакторе локальной политики безопасности

Аудит доступа к объектам

Важное использование механизма аудита во многих средах — это поддерживать журнал доступа к защищенным объектам, в частности файлам. Для этого необходимо включить политику доступа к объектам аудита, и в списках контроля доступа к системе должны быть аудиты ACE, которые позволяют проводить аудит для рассматриваемых объектов.

Когда процесс пытается открыть дескриптор объекта, монитор состояния защиты (SRM) сначала определяет, разрешена или запрещена попытка. Если включен аудит доступа к объектам, SRM затем сканирует системный ACL объекта.

Записи аудита доступа к объектам включают не только факт разрешенного или запрещенного доступа, но также и причину успеха или неудачи. Эта отчетность «причина для доступа» обычно принимает форму записи контроля доступа, указанной в SDDL (языке определения дескриптора безопасности) в записи аудита. Это позволяет диагностировать сценарии, в которых разрешен доступ к объекту, которому вы считаете, доступ, или наоборот, путем идентификации конкретной записи контроля доступа, которая привела к попытке доступа к успешному завершению или сбою.

Расширенные настройки политики аудита

В дополнение к параметрам политики аудита, описанным ранее, редактор политики локальной безопасности предлагает гораздо более детальный набор элементов управления аудитом в разделе конфигурации расширенной политики аудита, как показано на рис. 5.2.

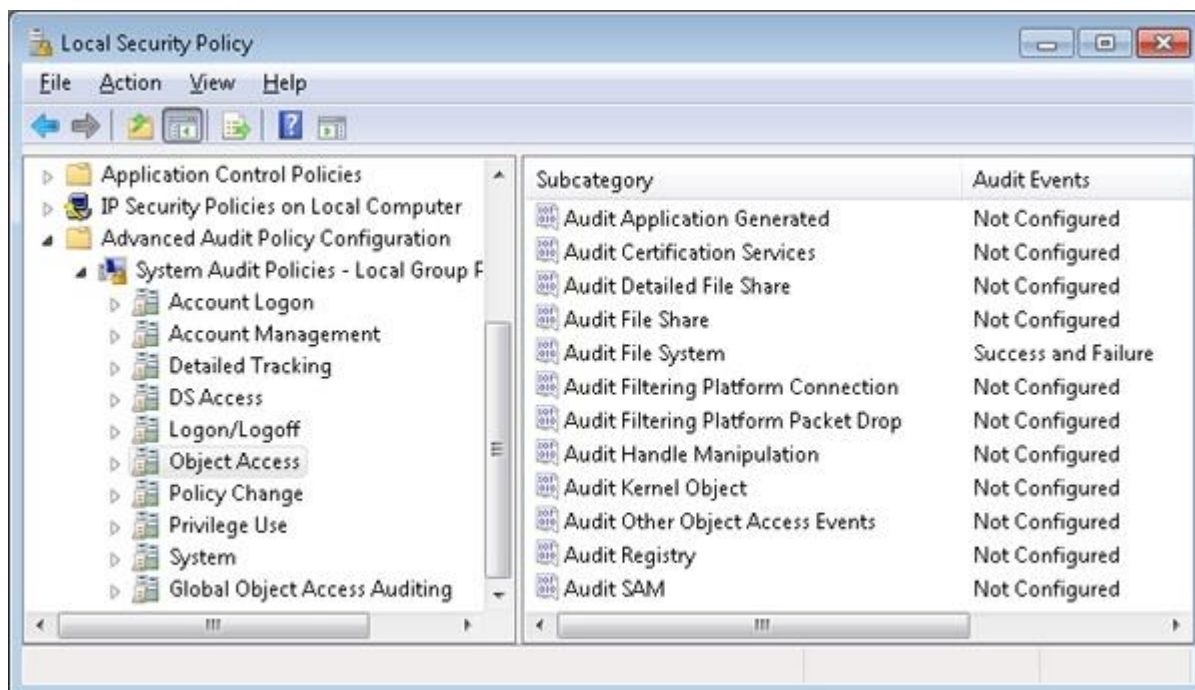


Рис. 5.2. Настройка расширенной политики аудита в редакторе локальной политики безопасности

Каждый из девяти параметров политики аудита в разделе «Локальные политики», как показано на рисунке 5.1, отображает группу параметров, которые обеспечивают более подробный контроль. Например, хотя параметры доступа к объектам аудита в разделе «Локальные политики» позволяют получить доступ ко всем объектам, подлежащим аудиту, параметры здесь позволяют контролировать доступ к отдельным видам объектов индивидуально. Включение одного из параметров политики аудита в разделе «Локальные политики» неявно включает все соответствующие расширенные события политики аудита, но, если требуется более точный контроль над содержимым журнала аудита, дополнительные параметры можно установить индивидуально. Затем стандартные настройки становятся результатом расширенных настроек; однако это не видно в редакторе политики локальной безопасности. Попытки указать параметры аудита, используя как базовые, так и расширенные параметры, могут привести к неожиданным результатам.

5.2. Подсистема безопасности Windows

5.2.1. Предоставление доступа к объекту в Windows

Подсистема безопасности Windows состоит из многих компонентов, в данном курсе мы ограничимся рассмотрением только двух из них.

- **Монитор состояния защиты (Security Reference Monitor, SRM).** Компонент исполнительной системы (\Windows\System32\Ntoskrnl.exe), отвечающий за определение структуры данных маркера доступа для представления контекста защиты, за проверку прав доступа к объектам, манипулирование привилегиями (правами пользователей) и генерацию сообщений аудита безопасности.

- **Подсистема локальной аутентификации (Local security authentication subsystem, LSASS).** Процесс пользовательского режима (\Windows\System32\Lsass.exe), который отвечает за политику безопасности в локальной системе (например, список пользователей, имеющих право на вход в систему, правила, связанные с паролями, привилегии, выдаваемые пользователям и их группам, параметры аудита безопасности системы), а также за аутентификацию пользователей и передачу сообщений аудита безопасности в журнал событий Event Log. Основную часть этой функциональности реализует сервис локальной аутентификации Lsassrv (\Windows\System32\Lsassrv.dll) — DLL-модуль, загружаемый Lsass.

Защита объектов и протоколирование обращений к ним находится в центре управления избирательным доступом и аудита. Защищаемые объекты Windows включают файлы, устройства, почтовые ящики, каналы (именованные и анонимные), задания, процессы, потоки, события, пары событий, мьютексы, семафоры, порты завершения ввода-вывода, разделы общей памяти, LPC-порты, ожидаемые таймеры, маркеры доступа, тома, объекты WindowStation, рабочие столы, сетевые ресурсы, сервисы, разделы реестра, принтеры и объекты Active Directory.

Windows требует от пользователя входа с аутентификацией, прежде чем ему будет разрешено обращаться к системным ресурсам. Когда какой-либо процесс запрашивает описатель объекта, диспетчер объектов и система защиты на основе идентификационных данных вызывающего процесса определяют, можно ли предоставить ему описатель, разрешающий доступ к нужному объекту.

Контекст защиты потока может отличаться от контекста защиты его процесса. Этот механизм называется олицетворением (impersonation), или подменой. При олицетворении механизмы проверки защиты используют вместо контекста защиты процесса контекст защиты потока, а без олицетворения — контекст защиты процесса, которому принадлежит поток. Важно не забывать, что все потоки процесса используют одну и ту же таблицу описателей, поэтому, когда поток открывает какой-нибудь объект (даже при олицетворении), все потоки процесса получают доступ к этому объекту.

Модель защиты Windows требует, чтобы поток заранее — еще до открытия объекта — указывал, какие операции он собирается выполнять над этим объектом. Система проверяет тип доступа, запрошенный потоком, и, если такой доступ ему разрешен, он получает описатель, позволяющий ему (и другим

потокам того же процесса) выполнять операции над объектом. Диспетчер объектов регистрирует права доступа, предоставленные для данного описателя, в таблице описателей, принадлежащей процессу.

Диспетчер объектов (Object Manager) – это исполнительная подсистема, к которой обращаются все остальные модули исполнительной подсистемы, в частности, системные вызовы, когда им необходимо получить доступ к ресурсам Windows. Диспетчер объектов служит для уменьшения дублирования объектов, которое могло бы привести к ошибкам в работе системы.

Одно из событий, заставляющее диспетчер объектов проверять права доступа, — открытие процессом существующего объекта по имени. При открытии объекта по имени диспетчер объектов передает своей внутренней функции ObCheckObjectAccess маркер защиты потока, открывающего объект, типы запрошенного им доступа (чтение, запись, удаление и т. д.), а также указатель на объект.

У диспетчера ввода-вывода, определяющего объекты типа «файл», имеется драйвер файловой системы, который управляет защитой своих файлов (или решает не реализовать ее). Таким образом, когда система запрашивает информацию о защите объекта «файл», представляющего файл на томе NTFS, она получает эту информацию от драйвера файловой системы NTFS, который в свою очередь получает ее от метода защиты объекта «файл», принадлежащего диспетчеру ввода-вывода. Заметьте, что при открытии файла ObCheckObjectAccess не выполняется, так как объекты «файл» находятся во вторичных пространствах имен; система вызывает метод защиты объекта «файл», только если поток явно запрашивает или устанавливает параметры защиты файла (например, через Windows-функции SetFileSecurity или GetFileSecurity).

Функция ObCheckObjectAccess получает информацию о защите объекта (в случае файла она обращается к драйверу файловой системы) и вызывает SRM-функцию SeAccessCheck, на которую опирается вся модель защиты Windows. Она принимает параметры защиты объекта, идентификационные данные защиты потока (в том виде, в каком они получены ObCheckObjectAccess) и тип доступа, запрашиваемый потоком. SeAccessCheck возвращает True или False в зависимости от того, предоставляет ли она потоку запрошенный тип доступа к объекту.

Функции защиты Windows также позволяют Windows-приложениям определять собственные закрытые объекты и вызывать SRM-сервисы для применения к этим объектам средств защиты Windows. Многие функции режима ядра, используемые диспетчером объектов и другими компонентами исполнительной системы для защиты своих объектов, экспортируются в виде Windows-функций пользовательского режима. Например, эквивалентом SeAccessCheck для пользовательского режима является AccessCheck. Таким образом, Windows-приложения могут применять модель защиты Windows и интегрироваться с интерфейсами аутентификации и администрирования этой операционной системы.

Итак, сущность модели защиты SRM отражает математическое выражение с тремя входными параметрами: идентификационными данными защиты потока, запрошенным типом доступа и информацией о защите объекта. Его результат — значения «да» или «нет», которые определяют, предоставит ли модель защиты запрошенный тип доступа.

5.2.2. Идентификаторы безопасности

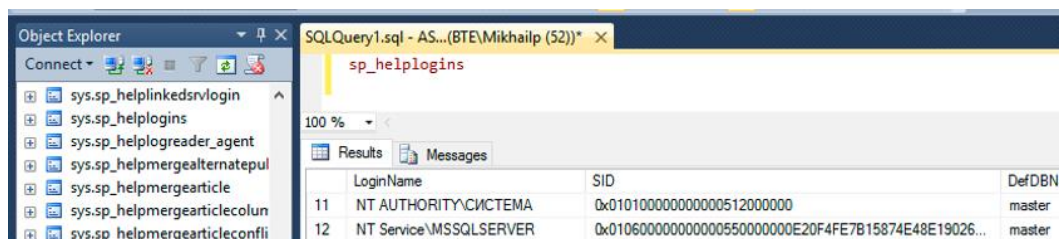
Для идентификации объектов, выполняющих в системе различные действия, Windows использует не имена (которые могут быть не уникальными), а идентификаторы безопасности (security identifiers, SID). SID имеются у пользователей, локальных и доменных групп, локальных компьютеров, доменов и членов доменов. SID представляет собой числовое значение переменной длины, формируемое из номера версии структуры SID, 48-битного кода агента идентификатора и переменного количества 32-битных кодов субагентов и/или относительных идентификаторов (relative identifiers, RID). Код агента идентификатора (identifier authority value) определяет агент, выдавший SID. Таким агентом обычно является локальная система или домен под управлением Windows. Коды субагентов идентифицируют попечителей, уполномоченных агентом, который выдал SID, а RID — не более чем средство создания уникальных SID на основе общего базового SID (common-based SID). Поскольку длина SID довольно велика и Windows старается генерировать истинно случайные значения для каждого SID, вероятность появления двух одинаковых SID практически равна нулю.

В текстовой форме каждый SID начинается с префикса S за которым следуют группы чисел, разделяемые дефисами, например:

S-1-5-21-1342198467-4818632800-834512242-1171

В этом SID номер версии равен 1, код агента идентификатора — 5 (центр безопасности Windows), далее идут коды четырех субагентов и один RID в конце (1171). Этот SID относится к домену, так что локальный компьютер этого домена получит SID с тем же номером версии и кодом агента идентификатора; кроме того, в нем будет столько же кодов субагентов.

Некоторые программы, например, Microsoft SQL Server Management Studio, отображают SID в двоичном виде. Зная внутреннюю структуру SID, легко записать его в текстовом виде (см. рис. 5.3).



| LoginName | SID | DefDBN |
|------------------------|--|--------|
| NT AUTHORITY\SYSTEM | 0x010100000000000512000000 | master |
| NT Service\MSSQLSERVER | 0x01060000000000000550000000E20F4FE7B15874E48E19026... | master |

0x010100000000000512000000
 01 = Revision
 01 = Sub-Authority Count
 000000000005 (5) = Authority
 12000000 (18) = Sub-Authority 1
 S-1-5-18 Локальная система

Рис. 5.3. Пример анализа двоичного представления SID

SID назначается компьютеру при установке Windows (программой Windows Setup). Далее Windows назначает SID локальным учетным записям на этом компьютере. SID каждой локальной учетной записи формируется на основе SID компьютера с добавлением RID. RID пользовательской учетной записи начинается с 1000 и увеличивается на 1 для каждого нового пользователя или группы. Аналогичным образом мастер конфигурирования Active Directory выдает SID только что созданному домену. Новые учетные записи домена получают SID, формируемые на основе SID домена с добавлением RID (который также начинается с 1000 и увеличивается на 1 для каждого нового пользователя или группы).

Многим предопределенным учетным записям и группам Windows выдает SID, состоящие из SID компьютера или домена и предопределенного RID. Так, RID учетной записи администратора равен 500, а RID гостевой учетной записи — 501. Например, в основе SID учетной записи локального администратора лежит SID компьютера, к которому добавлен RID, равный 500:

S-1-5-21-1342198467-4818632800-834512242-500

Для групп Windows также определяет ряд встроенных локальных и доменных SID. Например, SID, представляющий любую учетную запись, называется Everyone или World и имеет вид S-1-1-0. Еще один пример — группа NETWORK, то есть группа, пользователи которой зарегистрировались на данном компьютере из сети. SID сетевой группы имеет вид S-1-5-2. Список некоторых общеизвестных SID приведен в таблице 5.1 (полный список можно найти по ссылке, приведенной ниже).

Таблица 5.1. Некоторые SID, используемые в Windows

| SID | Название | Описание |
|----------|---|--|
| S-1-1-0 | Everyone Все | Группа, в которую входят все пользователи, даже анонимные пользователи и гости. Принадлежность контролируется операционной системой. |
| S-1-5-7 | Anonymous Анонимный | Группа, в которую входят все пользователи, вошедшие в систему анонимно. Принадлежность контролируется операционной системой. |
| S-1-5-11 | Authenticated Users Прошедшие проверку | Группа, в которую входят все пользователи, идентификаторы которых были проверены при входе в систему. Принадлежность контролируется операционной системой. |
| S-1-5-18 | Local System Локальная система | Учетная запись службы, используемая операционной системой |
| S-1-5-19 | Local Service Локальная служба | Учетная запись для служб, на локальной системе получает права, эквивалентные группе Users, на удаленной системе — Anonymous |

| | | |
|--------------|-----------------------------------|--|
| S-1-5-20 | Network Service Сетевая служба | Учетная запись для служб, на локальной системе получает права, эквивалентные группе Users, на удаленной системе использует учетную запись компьютера |
| S-1-5-32-544 | Administrators Администраторы | Встроенная группа. После первоначальной установки операционной системы единственным членом этой группы является учетная запись «Администратор». Когда компьютер присоединяется к домену, группа «Администраторы домена» добавляется к группе «Администраторы». Когда сервер становится контроллером домена, группа «Администраторы предприятия» также добавляется к группе «Администраторы». |
| S-1-5-32-545 | Users Пользователи | Встроенная группа. После первоначальной установки операционной системы единственным членом этой группы является группа «Прошедшие проверку». Когда компьютер присоединяется к домену, группа «Пользователи домена» добавляется к группе «Пользователи» на этом компьютере. |
| S-1-5-32-546 | Guests Гости | Встроенная группа. По умолчанию единственным членом группы является учетная запись «Гость». Группа «Гости» предоставляет возможность периодическим или однократным пользователям входить в систему с ограниченными правами встроенной в компьютер учетной записи «Гость». |

Хорошо известные идентификаторы безопасности в операционных системах Windows

<https://support.microsoft.com/ru-ru/help/243330/well-known-security-identifiers-in-windows-operating-systems>

Наконец, Winlogon создает уникальный SID для каждого интерактивного сеанса входа. SID входа, как правило, используется в элементе списка управления доступом (access-control entry, ACE), который разрешает доступ на время сеанса входа клиента. Например, Windows-сервис может вызвать функцию LogonUser для запуска нового сеанса входа. Эта функция возвращает маркер доступа, из которого сервис может извлечь SID входа. Потом этот SID сервис может использовать в ACE, разрешающем обращение к интерактивным объектам WindowStation и Desktop из сеанса входа клиента. SID для сеанса входа выглядит как S-1-5-5-0, а RID генерируется случайным образом.

5.2.3. Маркер доступа пользователя

Для идентификации контекста защиты процесса или потока SRM использует объект, называемый маркером (token), или маркером доступа (access token).

В контекст защиты входит информация, описывающая привилегии, учетные записи и группы, сопоставленные с процессом или потоком. В процессе входа в систему (этот процесс рассматривается в конце главы) Winlogon создает начальный маркер, представляющий пользователя, который входит в систему, и сопоставляет его с начальным процессом (или процессами) — по умолчанию запускается Userinit.exe. Так как дочерние процессы по умолчанию наследуют копию маркера своего создателя, все процессы в сеансе данного пользователя выполняются с одним и тем же маркером. Вы можете сгенерировать маркер вызовом Windows-функции LogonUser и применить его для создания процесса, который будет выполняться в контексте защиты пользователя, зарегистрированного с помощью функции LogonUser, с этой целью вы должны передать полученный маркер Windows-функции CreateProcessAsUser. Функция CreateProcessWithLogon тоже создает маркер, создавая новый сеанс входа с начальным процессом. Именно так команда runas запускает процессы с альтернативными маркерами.

Длина маркеров варьируется из-за того, что учетные записи разных пользователей имеют неодинаковые наборы привилегий и сопоставлены с разными учетными записями групп. Но все маркеры содержат одну и ту же информацию, показанную на рис. 5.4.

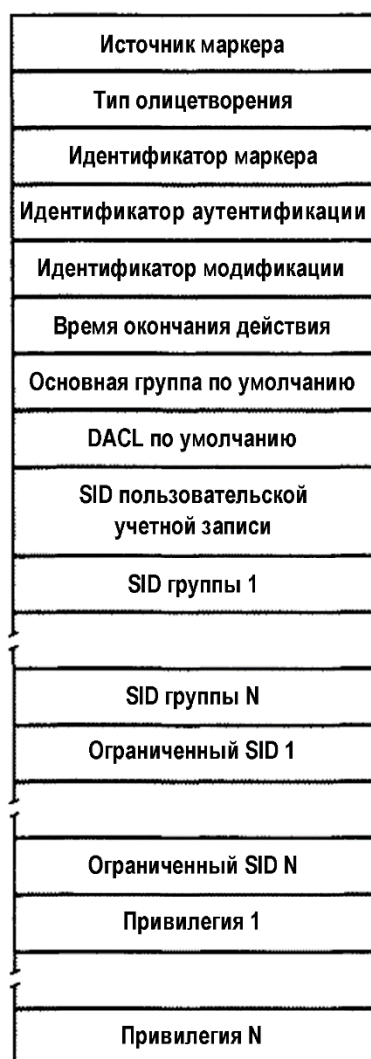


Рис. 5.4. Структура маркера доступа

Маркер доступа состоит из различных полей, включая следующие:

- **Идентификатор.**

- **Идентификатор ассоциированной сессии входа в систему.** Сессия обслуживается сервисом идентификации и заполняется идентификационными пакетами с коллекцией всей информации (мандат), сообщенной пользователем во время входа в систему. Мандат используется для доступа к удаленным системам без необходимости переидентифицировать клиента, предусматривающий, что все вовлеченные системы делятся информацией по идентификации.

- **Идентификатор пользователя.** Это поле наиболее важное и защищено от записи.

- **Идентификаторы групп,** частью которых является пользователь (или, точнее, субъект). Идентификаторы групп не могут быть удалены, но могут быть отключены. Как максимум, одна из групп назначается идентификатором сессии, произвольная группа, представляющая собой сессию входа в систему, позволяющая получить доступ к различным объектам, ассоциированным с сессией.

- **Ограничивающие идентификаторы группы** (поле не обязательно). Это дополнительное множество групп не дающее дополнительного доступа, но ограничивающее его: доступ к объекту открыт только если он также открыт для одной из этих групп. Данный вид групп не может быть ни удалён, ни отключён.

- **Привилегии, то есть специальные возможности пользователя.** Большинство привилегий по умолчанию отключены, чтобы исключить возможные повреждения от плохо защищённых программ.

- **Владелец по умолчанию, первичная группа и ACL для объектов,** созданных субъектом, ассоциированным с маркером пользователя.

Механизмы защиты в Windows используют два элемента маркера, определяя, какие объекты доступны и какие операции можно выполнять. Первый элемент состоит из SID учетной записи пользователя и полей SID групп. Используя SID-идентификаторы, SRM определяет, можно ли предоставить запрошенный тип доступа к защищаемому объекту, например, к файлу в NTFS.

SID групп в маркере указывают, в какие группы входит учетная запись пользователя. При обработке клиентских запросов серверные приложения могут блокировать определенные группы для ограничения удостоверений защиты, сопоставленных с маркером. Блокирование группы дает почти тот же эффект, что и ее исключение из маркера.

Вторым элементом маркера, определяющим, что может делать поток или процесс, которому назначен данный маркер, является список привилегий — прав, сопоставленных с маркером. Примером привилегии может служить право процесса или потока, сопоставленного с маркером, на выключение компьютера. Поля основной группы маркера по умолчанию и списка управления избирательным доступом (discretionary access-control list, DACL) представляют собой атрибуты защиты, применяемые Windows к объектам, которые создаются процессом или потоком с использованием маркера. Включая в маркеры информацию о защите, Windows упрощает процессам и потокам создание объектов со стандартными атрибутами защиты, так как в этом случае им не требуется запрашивать информацию о защите при создании каждого объекта.

Маркер может быть основным (primary token) (идентифицирует контекст защиты процесса) и олицетворяющим (impersonation token) (применяется для временного заимствования потоком другого контекста защиты — обычно другого пользователя). Маркеры олицетворения сообщают уровень олицетворения, определяющий, какой тип олицетворения активен в маркере.

Содержимое маркера можно косвенно увидеть с помощью Process Explorer (www.sysinternals.com) на вкладке Security в диалоговом окне свойств процесса. В этом окне показываются группы и привилегии, включенные в маркер исследуемого вами процесса (рис. 5.5).

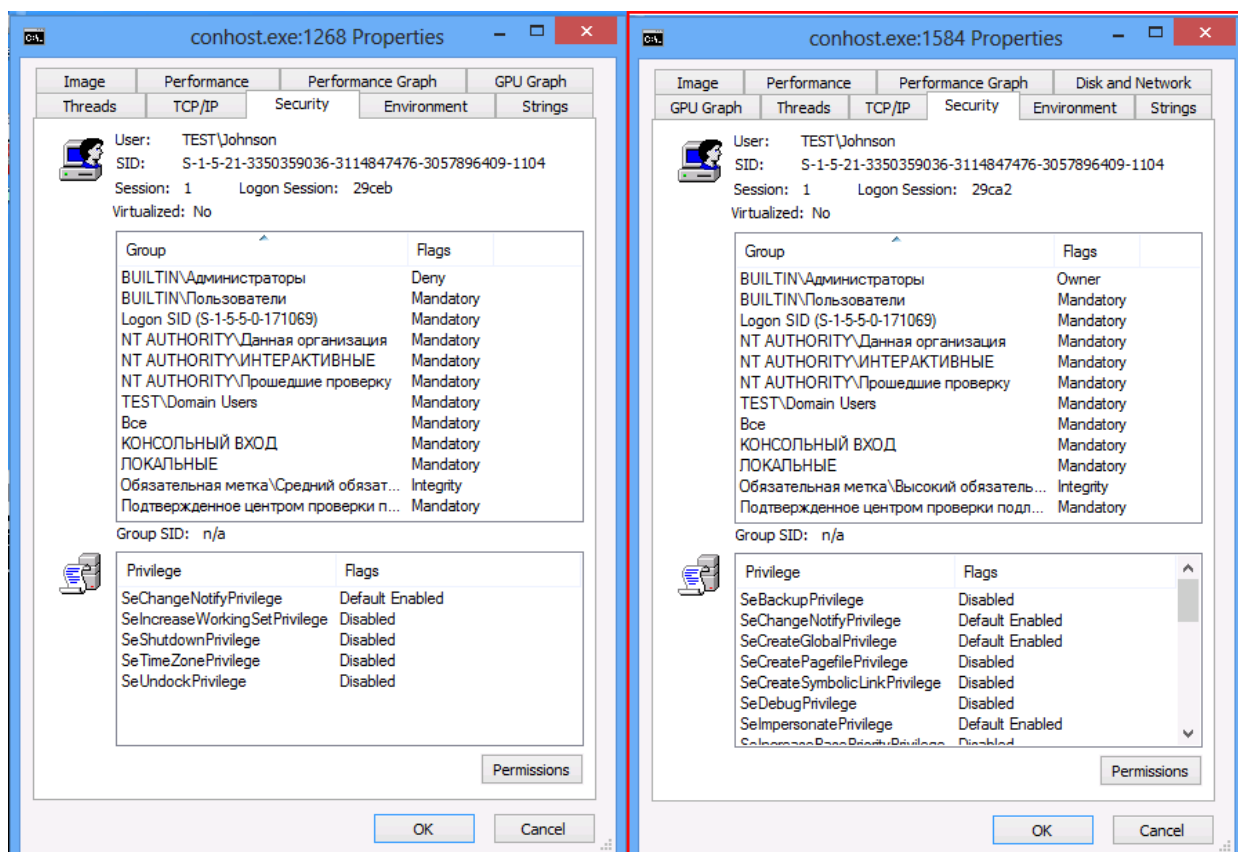


Рис. 5.5. Ограниченный и основной маркеры доступа пользователя с административными полномочиями

Ограниченный маркер (restricted token) создается на базе основного или олицетворяющего с помощью функции CreateRestrictedToken и является его копией, в которую можно внести следующие изменения:

- удалить некоторые элементы из таблицы привилегий маркера;
- пометить SID-идентификаторы маркера атрибутом проверки только на запрет (deny-only);
- пометить SID-идентификаторы маркера как ограниченные.

Ограниченные маркеры удобны, когда приложение подменяет клиент при выполнении небезопасного кода. В ограниченном маркере может, например, отсутствовать привилегия на перезагрузку системы, что не позволит коду, выполняемому в контексте защиты ограниченного маркера, перезагрузить систему.

5.2.4. Списки контроля доступа (ACL);

Маркеры, которые идентифицируют удостоверения пользователя, являются лишь частью выражения, описывающего защиту объектов. Другая его часть — информация о защите, сопоставленная с объектом и указывающая, кому и какие действия разрешено выполнять над объектом. Структура данных, хранящая эту информацию, называется дескриптором защиты (security descriptor). Дескриптор защиты включает следующие атрибуты.

- **Номер версии модели защиты SRM**, использованной для создания дескриптора.
- **Флаги**. Необязательные модификаторы, определяющие поведение или характеристики дескриптора. Пример — флаг SE_DACL_PROTECTED, который запрещает наследование дескриптором параметров защиты от другого объекта.
- **SID владельца**. Идентификатор защиты владельца.
- **SID группы**. Идентификатор защиты основной группы для данного объекта (используется только POSIX).
- **Список управления избирательным доступом (discretionary access-control list, DACL)**. Указывает, кто может получать доступ к объекту и какие виды доступа (рис. 5.6).
- **Системный список управления доступом (system access-control list, SACL)**. Указывает, какие операции и каких пользователей должны регистрироваться в журнале аудита безопасности.

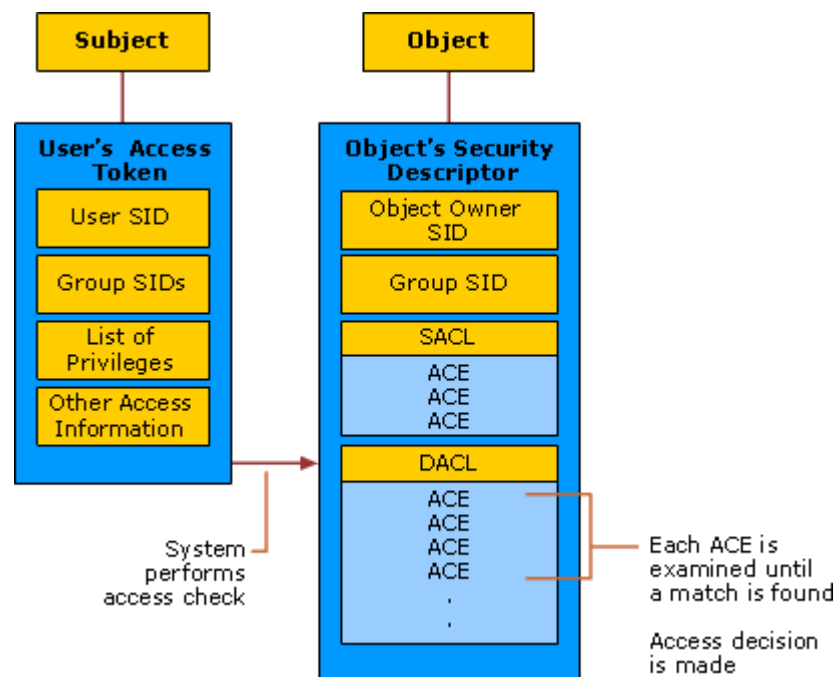


Рис. 5.6. Участие дескриптора безопасности в проверке доступа к объекту

Список управления доступом (access-control list, ACL) состоит из заголовка и может содержать элементы (access-control entries, ACE). Существует два типа ACL: DACL и SACL. В DACL каждый ACE содержит SID и маску доступа (а также набор флагов), причем ACE могут быть четырех типов: «доступ разрешен» (access allowed), «доступ отклонен» (access denied), «разрешенный объект» (allowed-object) и «запрещенный объект» (denied-object). Первый тип ACE

разрешает пользователю доступ к объекту, а второй — отказывает в предоставлении прав, указанных в маске доступа.

Разница между ACE типа «разрешенный объект» и «доступ разрешен», а также между ACE типа «запрещенный объект» и «доступ отклонен» заключается в том, что эти типы используются только в Active Directory. ACE этих типов имеют поле глобально уникального идентификатора (globally unique identifier, GUID), которое сообщает, что данный ACE применим только к определенным объектам или дочерним объектам (с GUID-идентификаторами). Кроме того, необязательный GUID указывает, что тип дочернего объекта наследует ACE при его (объекта) создании в контейнере Active Directory, к которому применен ACE. (GUID — это гарантированно уникальный 128-битный идентификатор.)

За счет аккумуляции прав доступа, сопоставленных с индивидуальными ACE, формируется набор прав, предоставляемых ACL-списком. Если в дескрипторе защиты нет DACL (DACL = null), любой пользователь получает полный доступ к объекту. Если DACL пуст (т. е. в нем нет ACE), доступа к объекту не получает никто.

ACE, используемые в DACL, также имеют набор флагов, контролирующих и определяющих характеристики ACE, связанные с наследованием. Некоторые пространства имен объектов содержат объекты-контейнеры и объекты-листья (leaf objects). Контейнер может включать другие контейнеры и листья, которые являются его дочерними объектами. Примеры контейнеров — каталоги в пространстве имен файловой системы и разделы в пространстве имен реестра. Отдельные флаги контролируют, как ACE применяется к дочерним объектам контейнера, сопоставленного с этим ACE.

SACL состоит из ACE двух типов: системного аудита (system audit ACE) и объекта системного аудита (system audit-object ACE). Эти ACE определяют, какие операции, выполняемые над объектами конкретными пользователями или группами, подлежат аудиту. Информация аудита хранится в системном журнале аудита. Аудиту могут подлежать как успешные, так и неудачные операции. Как и специфические для объектов ACE из DACL, ACE объектов системного аудита содержат GUID, указывающий типы объектов или дочерних объектов, к которым применим данный ACE, и необязательный GUID, контролирующий передачу ACE дочерним объектам конкретных типов. При SACL, равном null, аудит объекта не ведется. Флаги наследования, применимые к DACL ACE, применимы к ACE системного аудита и объектов системного аудита.

Дескриптор защиты содержит два ACE типа «доступ разрешен», причем один из них указывает учетную запись администратора (ее можно распознать по RID, равному 500), а другой — учетную запись System (которая всегда выглядит как S-1-5-18). Без декодирования битов, установленных в масках доступа в ACE и определения того, каким типам доступа к процессам они соответствуют, очень трудно сказать, какими правами доступа к объекту «процесс» для Winlogon обладает каждая из этих учетных записей. Однако, если вы сделаете это, используя заголовочные файлы из SDK, то обнаружите, что обе учетные записи имеют полные права доступа.

Чтобы определить, какой DACL следует назначить новому объекту, система защиты использует первое применимое правило из следующего списка.

1. Если вызывающий поток явно предоставляет дескриптор защиты при создании объекта, то система защиты применяет его к объекту. Если у объекта есть имя и он находится в объекте-контейнере (например, именованное событие в каталоге \BaseNamedObjects пространства имен диспетчера объектов), система объединяет в DACL все наследуемые ACE (ACE, которые могут быть переданы от контейнера объекта), но только в том случае, если в дескрипторе защиты не установлен флаг SE_DACL_PROTECTED, запрещающий наследование.

2. Если вызывающий поток не предоставляет дескриптор защиты и объекту присваивается имя, система защиты ищет этот дескриптор в контейнере, в котором хранится имя нового объекта. Некоторые ACE каталога объектов могут быть помечены как наследуемые. Это означает, что они должны применяться к новым объектам, создаваемым в данном каталоге. При наличии наследуемых ACE система защиты формирует из них ACL, назначаемый новому объекту. (В ACE, наследуемых только объектами-контейнерами, устанавливаются отдельные флаги.)

3. Если дескриптор защиты не определен и объект не наследует какие-либо ACE, система защиты извлекает DACL по умолчанию из маркера доступа вызывающего потока и применяет его к новому объекту. В некоторые подсистемы Windows (например, службы, LSA и SAM-объекты) «защиты» свои DACL, назначаемые ими объектам при создании.

4. Если дескриптор защиты не определен и нет ни наследуемых ACE, ни DACL по умолчанию, система создает объект без DACL, что открывает полный доступ к нему любым пользователям и группам. Это правило идентично третьему, если маркер содержит нулевой DACL по умолчанию. Правила, используемые системой при назначении SACL новому объекту, аналогичны правилам присвоения DACL за двумя исключениями. Первое заключается в том, что наследуемые ACE системного аудита не передаются объектам с дескрипторами защиты, помеченными флагом SE_SACL_PROTECTED (DACL точно так же защищается флагом SE_DACL_PROTECTED). Второе исключение: если ACE системного аудита не определены и наследуемого SACL нет, то SACL вообще не присваивается объекту (в маркерах нет SACL по умолчанию).

Когда к контейнеру применяется новый дескриптор защиты, содержащий наследуемые ACE, система автоматически передает их в дескрипторы защиты дочерних объектов. (Заметьте, что DACL дескриптора защиты не принимает наследуемые DACL ACE, если установлен флаг SE_DACL_PROTECTED, а его SACL не наследует SACL ACE, если установлен флаг SE_SACL_PROTECTED.) В соответствии с порядком слияния наследуемых ACE с дескриптором защиты дочернего объекта любые ACE, явно примененные к ACL, размещаются до ACE, унаследованных объектом. Система использует следующие правила передачи наследуемых ACE.

- Если дочерний объект без DACL наследует ACE, он получает DACL, содержащий лишь унаследованные ACE.

- Если дочерний объект с пустым DACL наследует ACE, он также получает DACL, содержащий лишь унаследованные ACE.

- Только для объектов в Active Directory: если наследуемый ACE удаляется из родительского объекта, все копии этого ACE автоматически удаляются из всех дочерних объектов.

- Только для объектов в Active Directory: если из DACL дочернего объекта автоматически удалены все ACE, у дочернего объекта остается пустой DACL.

Порядок ACE в ACL является важным аспектом модели защиты Windows.

5.2.5. Определение уровня доступа

Для определения прав доступа к объекту используются два алгоритма:

- сравнивающий запрошенные права с максимально возможными для данного объекта и экспортируемый в пользовательский режим в виде Windows-функции `GetEffectiveRightsFromAcl`;

- проверяющий наличие конкретных прав доступа и активизируемый через Windows-функцию `AccessCheck` или `AccessCheckByType`.

Первый алгоритм проверяет элементы DACL следующим образом.

1. В отсутствие DACL (DACL = null) объект является незащищенным, и система защиты предоставляет к нему полный доступ.

2. Если у вызывающего потока имеется привилегия на захват объекта во владение (take-ownership privilege), система защиты предоставляет владельцу право на доступ для записи (write-owner access) до анализа DACL.

3. Если вызывающий поток является владельцем объекта, ему предоставляются права управления чтением (read-control access) и доступа к DACL для записи (write-DACL access).

4. Из маски предоставленных прав доступа удаляется маска доступа каждого ACE типа «доступ отклонен», SID которого совпадает с SID маркера доступа вызывающего потока.

5. К маске предоставленных прав доступа добавляется маска доступа каждого ACE типа «доступ разрешен», SID которого совпадает с SID маркера доступа вызывающего потока (исключение составляют права доступа, в предоставлении которых уже отказано).

После анализа всех элементов DACL рассчитанная маска предоставленных прав доступа возвращается вызывающему потоку как максимальные права доступа. Эта маска отражает полный набор типов доступа, которые этот поток сможет успешно запрашивать при открытии данного объекта.

Второй алгоритм проверяет, можно ли удовлетворить конкретный запрос на доступ, исходя из маркера доступа вызывающего потока. У каждой Windows-функции открытия защищенных объектов есть параметр, указывающий желательную маску доступа — последний элемент выражения, описывающего защиту объектов. Чтобы определить, имеет ли вызывающий поток право на доступ к защищенному объекту, выполняются следующие операции.

1. В отсутствие DACL (DACL = null) объект является незащищенным, и система защиты предоставляет к нему запрошенный тип доступа.

2. Если у вызывающего потока имеется привилегия на захват объекта во владение, система защиты предоставляет владельцу право на доступ для записи,

а затем анализирует DACL. Однако, если такой поток запросил только доступ владельца для записи, система защиты предоставляет этот тип доступа и не просматривает DACL.

3. Если вызывающий поток является владельцем объекта, ему предоставляются права управления чтением и доступа к DACL для записи. Если вызывающий поток запросил только эти права, система защиты предоставляет их без просмотра DACL.

4. Просматриваются все ACE в DACL — от первого к последнему. Обработка ACE выполняется при одном из следующих условий:

а. SID в ACE типа «доступ отклонен» совпадает с незаблокированным SID (SID могут быть незаблокированными и заблокированными) или SID с атрибутом проверки только на запрет в маркере доступа вызывающего потока;

б. SID в ACE типа «доступ разрешен» совпадает с незаблокированным SID в маркере доступа вызывающего потока, и этот SID не имеет атрибута проверки только на запрет;

с. Идет уже второй проход поиска в дескрипторе ограниченных SID, и SID в ACE совпадает с ограниченным SID в маркере доступа вызывающего потока.

5. В случае ACE типа «доступ разрешен» предоставляются запрошенные права из маски доступа ACE; проверка считается успешной, если предоставляются все запрошенные права. Доступ к объекту не предоставляется в случае ACE типа «доступ отклонен» и отказа в предоставлении какого-либо из запрошенных прав.

6. Если достигнут конец DACL и некоторые из запрошенных прав доступа еще не предоставлены, доступ к объекту запрещается.

7. Если все права доступа предоставлены, но в маркере доступа вызывающего потока имеется хотя бы один ограниченный SID, то система повторно сканирует DACL в поисках ACE, маски доступа которых соответствуют набору запрошенных прав доступа. При этом также идет поиск ACE, SID которых совпадает с любым из ограниченных SID вызывающего потока. Поток получает доступ к объекту, если запрошенные права доступа предоставлялись после каждого прохода по DACL.

Поведение обоих алгоритмов проверки прав доступа зависит от относительного расположения разрешающих и запрещающих ACE. Возьмем для примера объект с двумя ACE, первый из которых указывает, что определенному пользователю разрешен полный доступ к объекту, а второй отказывает в доступе. Если разрешающий ACE предшествует запрещающему, пользователь получит полный доступ к объекту. При другом порядке этих ACE пользователь вообще не получит доступа к объекту.

Более старые Windows-функции вроде `AddAccessAllowedAce` добавляли ACE в конец DACL, что нежелательно. Таким образом, до появления Windows 2000 большинство Windows-приложений были вынуждены создавать DACL вручную, помещая запрещающие ACE в начало списка. Несколько функций Windows, например `SetSecurityInfo` и `SetNamedSecurityInfo`, используют предпочтительный порядок ACE: запрещающие ACE предшествуют разрешающим. Заметьте, что эти функции вызываются при редактировании, например, прав доступа к NTFS-

файлам и разделам реестра. SetSecurityInfo и SetNamedSecurityInfo также применяют правила наследования ACE к дескриптору защиты, для операций над которым они вызываются.

Как уже говорилось, обработка DACL системой защиты при каждом использовании описателя процессом была бы неэффективной, поэтому SRM проверяет права доступа только при открытии описателя, а не при каждом его использовании. Так что, если процесс один раз успешно открыл описатель, система защиты не может аннулировать предоставленные при этом права доступа — даже когда DACL объекта изменяется. Учтите и вот еще что: поскольку код режима ядра обращается к объектам по указателям, а не по описателям, при использовании объектов операционной системой права доступа не проверяются. Иначе говоря, исполнительная система полностью доверяет себе в смысле защиты.

Тот факт, что владелец объекта всегда получает право на запись DACL при доступе к объекту, означает, что пользователям нельзя запретить доступ к принадлежащим им объектам. Если в силу каких-то причин DACL объекта пуст (доступ запрещен), владелец все равно может открыть объект с правом записи DACL и применить новый DACL, определяющий нужные права доступа.

Но, начиная с Windows Vista и Windows 2008, для ограничения доступа владельцев к созданным ими объектам можно использовать учетную запись OWNER RIGHTS.

5.2.6. Утверждения и динамический контроль доступа

В Windows Server 2012 появилась новая возможность – динамический контроль доступа (Dynamic Access Control — DAC). Это совершенно новый способ управления доступом к данным общим файловых ресурсов. В то время как права доступа к общим ресурсам и права NTFS управляют доступом к общим ресурсам, папкам и файлам на основе информации из учетной записи пользователя или членства в группе доступа, DAC добавляет к этим возможностям дополнительный уровень защиты. Этот уровень, в частности, содержит сравнение значений дополнительных атрибутов со значениями классификационных свойств папки, а затем разрешает или запрещает доступ на основе централизованных политик доступа и связанных с ними правил.

В DAC введен новый формат ACL для файлов и папок, в котором можно задавать выражения. Эти выражения основаны на трех компонентах.

- **Утверждения о пользователе/устройстве.** Эти свойства пользователей и устройств хранятся в маркере, что ускоряет проверку в виде членства в группах. Свойства могут представлять любые утверждения, которые пользователь или устройство могут заявить о себе. В частности, это могут быть любые атрибуты пользователя или компьютера в Active Directory.

Например:

Отдел: User.Department = "Маркетинг"

Роль: User.Role = "Руководитель "

Местоположение: Device.Location "Минск"

Тип: Device.Type = "Ноутбук"

• **Свойства ресурсов.** Особые свойства, связанные с защищенными ресурсами (файлы и папки), которые обычно применяются для классификации данных.

Например:

Секретность: Resource.Sensitivity = "Высокая"

Местоположение: Resource.Location = "Минск"

• **Права доступа.** Знакомые нам права доступа, которые применяются в стандартных ACL — в том числе и является ли элемент разрешением или запрещением.

Результирующая политика доступа может иметь вид:

ApplyTo : \$Resource.Sensitivity= "Высокая" | Allow Read/Write |

If { \$User.Role = "Руководитель" } and { \$Device.Type = "Ноутбук" }

Для использования динамического управления доступом требуются подготовительные действия — классификация данных и проверка доступа.

Все контроллеры доменов, файловые серверы и клиенты должны работать под управлением Windows Server 2012 и Windows 8 или новее. Ниже описан один из возможных сценариев построения инфраструктуры DAC.

1. **Создание типов утверждений DAC.** Это может быть, например, Department, если требуется предоставлять доступ в зависимости от того, в каком отделе работает пользователь.

2. **Настройка свойств ресурсов DAC.** Свойства ресурсов (resource property), которые можно настраивать и делать активными, можно использовать для определения классификационных свойств папок. Эти значения свойств ресурсов можно использовать при сравнении с утверждениями пользователей для управления доступом к данным в папках.

3. **Добавление и настройка свойств ресурсов в список свойств ресурсов (Resource Property).** Файловые серверы просто загружают списки свойств ресурсов, но не отдельные свойства ресурсов. Такой список позволяет группировать и сегментировать только свойства ресурсов, необходимые для построения классификации.

4. **Создание центрального правила доступа.** Центральное правило доступа определяет критерий для разрешения или запрета доступа к данным, защищаемым технологией DAC.

5. **Создание центральной политики доступа.** Центральная политика доступа может содержать одно или много центральных правил доступа и применяется или назначается файловым серверам как единое целое с помощью групповых политик.

6. **Создание и назначение объекта GPO центральной политики доступа к файловым серверам.** На этом шаге выполняется создание нового объекта групповой политики, настройка этого объекта и применение политики к назначенным файловым серверам Windows Server 2012.

7. **Разрешение бронирования (armoring) Kerberos для контроллеров доменов.** На этом шаге производится обновление конфигурации контроллеров доменов Kerberos, чтобы включить информацию DAC при генерации билетов

Kerberos для пользователей и компьютеров. Необходимый параметр групповой политики находится в узле Computer Configuration\Policies\Administrative Templates\System\KDC (Конфигурация компьютера\Политики\Административные шаблоны\Система\KDC) и называется **Support Dynamic Access Control and Kerberos Armoring** (Поддержка динамического управления доступом и бронирование Kerberos).

8. Создание и обновление общих файловых ресурсов для применения динамического управления доступом. На этом шаге выполняется обновление конфигурации безопасности для открытой папки файлового сервера, чтобы задействовать классификационные данные и критерии, необходимые в организации.

9. Настройка пользовательских учетных записей и тестирование доступа к данным. На этом шаге производится настройка пользовательских учетных записей и проверка доступа к папке, защищенной технологией DAC.

Дополнительную информацию можно получить по следующим ссылкам.

Обзор динамического контроля доступа

<https://docs.microsoft.com/ru-ru/windows-server/identity/solution-guides/dynamic-access-control-overview>

Dynamic Access Control в Windows Server 2012

<http://winitpro.ru/index.php/2013/01/24/dynamic-access-control-v-windows-server-2012/>

5.3. Инфраструктура открытого ключа

Если говорить кратко, инфраструктура открытых ключей (Public Key Infrastructure) — это совокупность цифровых сертификатов, бюро регистрации и центров сертификации, которые проверяют подлинность каждого участника обмена зашифрованными сообщениями. По сути, сама по себе инфраструктура открытых ключей — просто концепция, которая определяет механизмы защиты данных от чтения при их передаче и проверки подлинности пользователя, передавшего эти данные.

Реализации PKI широко распространены и становятся исключительно важным компонентом современных реализаций сетей.

Реализации PKI могут быть как простыми, так и сложными, а некоторые применяют массивы смарт-карт (или другие способы двухфакторной аутентификации) и сертификаты для проверки подлинности всех пользователей с высокой степенью достоверности. Поэтому каждая организация должна разобраться в возможностях PKI и выбрать нужную реализацию.

Лежащее в основе PKI шифрование с открытым ключом (public key), называемое также асимметричным шифрованием, использует комбинацию двух ключей, которые математически связаны друг с другом. Первый ключ, являющийся секретным ключом, хранится в строгой тайне и используется для цифровой подписи или расшифровки информации. Вторым — открытый — ключ может использоваться для проверки цифровой подписи или шифрования информации. Целостность открытого ключа обеспечивается сертификатами, которые подробно описаны в последующих разделах этой главы. Асимметричный подход к шифрованию значительно облегчает управление ключами, т.к. открытый ключ не нужно защищать, и у каждого пользователя имеется лишь один секретный ключ. Правда, это упрощение управления достигается за счет снижения производительности из-за усложнения математических операций.

5.3.1. Сертификаты X509.3.

Сертификат (certificate) представляет собой цифровой документ, который выдается доверяемым центром (централизованным, внутренним или локальным) и используется им для подтверждения подлинности пользователя. Доверяемые центры сертификации, такие как VeriSign, широко используются в Интернете, чтобы, например, подтверждать, что программное обеспечение Microsoft действительно разработано компанией Microsoft, а не служит маскировкой какого-либо вируса.

Сертификаты применяются для выполнения различных функций, некоторые из них перечислены ниже:

- защита электронной почты;
- Аутентификация во всемирной Сети;
- Защита данных в Интернете (IPsec);
- Подписание кода;
- Создание иерархий сертификации.

Все эти функции сводятся, в конечном счете, либо к шифрованию данных (при защите почтовых сообщений или веб-паролей и контента), либо к цифровой подписи данных для гарантии целостности и подлинности (при подписании кода или почтовых сообщений).

Сертификаты подписываются с помощью информации из открытого ключа субъекта и идентификационной информации — имя, адрес электронной почты и тому подобные сведения, — а также цифровой подписи организации, выпустившей сертификат, которая называется центром сертификации (Certificate Authority — CA). Если оба пользователя или компьютера доверяют одному и тому же центру сертификации, который выпустил сертификаты, они могут доверять и друг другу. В русскоязычной литературе используется термин удостоверяющий центр.

Сертификат, подобно паспорту или другому документу, имеет имя субъекта, номер сертификата, срок действия, выдавший орган, и цифровую подпись (рис. 5.7). И самые главные поля сертификата, без которых он не имел бы никакой ценности, это Субъект и Открытый ключ.

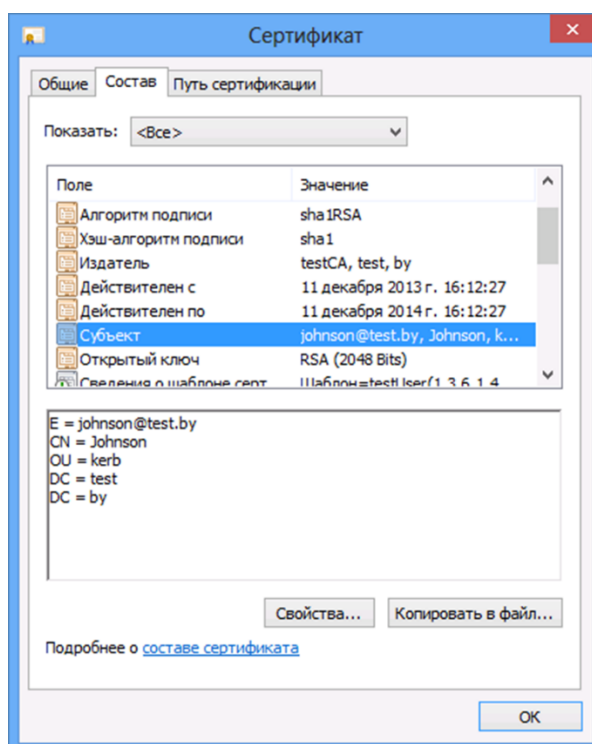


Рис. 5.7. Пример сертификата

5.3.2. Построение цепочки доверия сертификатов

Одной из основ PKI является модель доверия центрам сертификации (Certification Authority или просто CA). Прежде чем мы начнём доверять сертификату, мы должны явно доверять корневому сертификату CA и так же явно или косвенно (по правилу одностороннего транзитивного доверия) доверять всем промежуточным CA в цепочке (рис. 5.8). Корневые сертификаты устанавливаются в систему вручную путём добавления сертификата CA в секцию Trusted Root CAs. Если корневой сертификат CA есть в этом списке, то мы доверяем всем сертификатам, которые выдал этот CA и любые подчинённые CA (Intermediate CA).

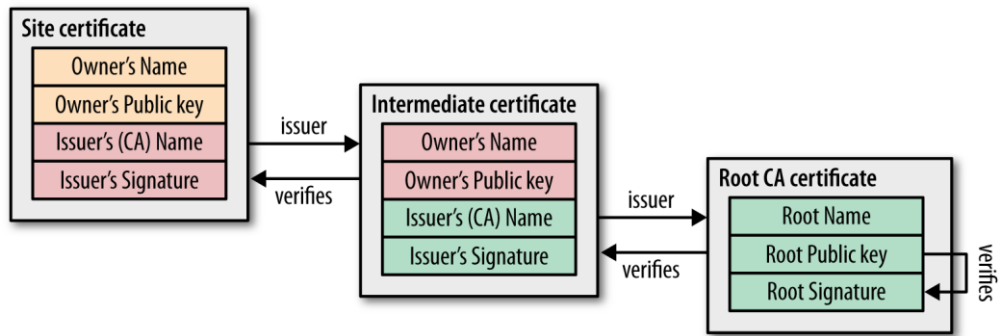


Рис. 5.8. Цепочка доверия сертификатов

Построение цепочки доверия производится в три этапа.

1. Поиск сертификатов в следующих местах:

- Кэш CryptoAPI
- Group Policy
- Enterprise Policy
- AIA расширение

2. Проверка годности сертификатов:

- Проверка подписи
- Проверка ограничений
- Проверка назначения
- Проверка специфических требований

3. Проверка отзыва сертификатов:

- Проверка CRL
- OCSP запрос (для Vista+)

При построении цепочки доверия используется один из трех методов:

1. Точное соответствие (рис. 5.9)

- AKI содержит Subject Name и Serial Number
- Выбирается сертификат с таким же именем и серийным номером
- Такой сертификат единственный

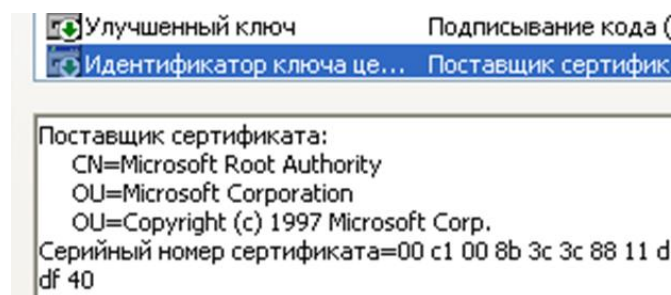


Рис. 5.9. Сопоставление сертификатов при точном соответствии

2. Соответствие ключей (рис. 5.10)

- AKI содержит хэш открытого ключа ЦС
- Выбирается сертификат, у которого SKI совпадает с требуемым AKI (если алгоритм SHA1, наличие поля SKI необязательно)
- Таких сертификатов может быть несколько, если обновление сертификата ЦС происходит с использованием существующей ключевой пары

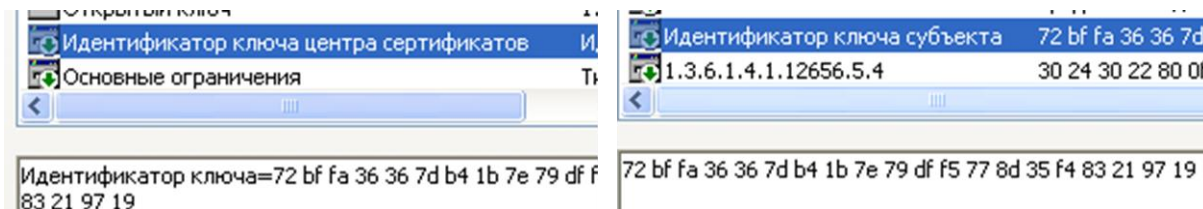


Рис. 5.10. Сопоставление сертификатов при соответствии ключей

3. Соответствие имен

- Если поле AKI отсутствует, выбирается сертификат ЦС, Subject которого совпадает с полем Issuer сертификата

5.3.3. Обзор возможностей службы Active Directory Certification Services

Windows Server 2012 содержит встроенную технологию CA, называемую службой сертификации Active Directory (Active Directory Certificate Services — AD CS). Первый вариант AD CS появился в Windows Server 2008, а раньше эта технология называлась просто службой сертификации (Certificate Services). AD CS может использоваться для создания сертификатов и последующего управления ими и отвечает за обеспечение их подлинности, отзыв и сроки годности. Зачастую AD CS в Windows Server 2012 используется без особой необходимости проверки сертификатов организации какой-либо независимой стороной.

Поэтому если сертификаты требуются только для участников внутри организации, часто применяется развертывание внутреннего CA для нужд внутренних пользователей и систем.

Широко используются и сторонние центры сертификации наподобие VeriSign, но они требуют дополнительного вложения средств.

AD CS для Windows Server 2012 можно установить в виде центра сертификации одного из перечисленных ниже типов.

- **Головной центр сертификации предприятия.** Головной CA предприятия является наиболее доверяемым CA в организации и должен быть установлен раньше всех остальных CA. Все остальные CA являются подчиненными по отношению к головному CA предприятия. Защите этого CA следует уделить самое пристальное внимание, так как компрометация CA предприятия означает компрометацию всей цепочки центров сертификации.

- **Подчиненный центр сертификации предприятия.** Подчиненный CA предприятия должен получить сертификат от головного CA предприятия, но после этого может выдавать сертификаты всем пользователям и компьютерам предприятия. Часто CA этого типа используются для создания масштабируемого набора CA с высокой степенью готовности и защиты головного CA предприятия.

- **Самостоятельный головной центр сертификации.** Самостоятельный головной CA служит вершиной иерархии, не связанной с информацией домена предприятия. В специальных случаях можно создать несколько самостоятельных CA. Самостоятельный головной CA часто используется в качестве корневого для других подчиненных CA предприятия — для повышения безопасности среды, т.к. самостоятельный головной центр можно вывести в автономный режим. То есть головной центр конфигурируется как самостоятельный, а подчиненные CA,

интегрированные в домен предприятия, установлены в доменах леса, чтобы обеспечить автоматическое развертывание в масштабе предприятия.

- **Самостоятельный подчиненный центр сертификации.** Самостоятельные подчиненные СА получают свои сертификаты от самостоятельного головного СА, и затем могут использоваться для распространения сертификатов пользователям и компьютерам, связанным с этим самостоятельным СА.

Примечание. Используется и другая терминология, где головной центр сертификации называется корневым (root), а самостоятельный — автономным (standalone).

AD CS состоит из нескольких служб ролей, который выполняют для клиентов различные задачи. При необходимости одну или несколько этих ролей можно установить на сервере. Эти службы кратко описаны ниже.

- **Центр сертификации (Certification Authority).** Данная служба устанавливает базовый компонент СА, позволяющий серверу издавать и отзываться сертификатами для клиентов и управлять ими. Эту роль можно установить на нескольких серверах в цепочке одного и того же головного СА.

- **Веб-включение центра сертификации (Certification Authority Web Enrollment).** Данная служба управляет распространением сертификатов клиентам через Интернет. Для ее работы нужно, чтобы на сервере была установлена служба информации Интернета (Internet Information Services — IIS).

- **Онлайновый ответчик (Online Responder).** Данная служба отвечает на запросы индивидуальных клиентов по поводу проверки конкретных сертификатов. Она применяется для сложных или больших сетей, которые должны выдерживать интенсивные периоды активности по отзыву или загрузку больших списков отзывов сертификатов (Certificate Revocation List — CRL).

- **Веб-служба развертывания сертификатов (Certificate Enrollment Web Service).** Эта новая служба позволяет пользователям и компьютерам выполнять удаленное развертывание сертификатов или развертывание из систем, не включенных в домен, по протоколу HTTP.

- **Веб-служба политики развертывания сертификатов (Certificate Enrollment Policy Web Service).** Эта служба работает с соответствующей веб-службой развертывания сертификатов, но предоставляет информацию о политике, а не сертификаты.

- **Служба включения сетевых устройств (Network Device Enrollment Service).** Данная служба упрощает получение сертификатов сетевыми устройствами наподобие маршрутизаторов.

Для управления центром сертификации можно использовать консоль Certification Authority (рис. 5.11).

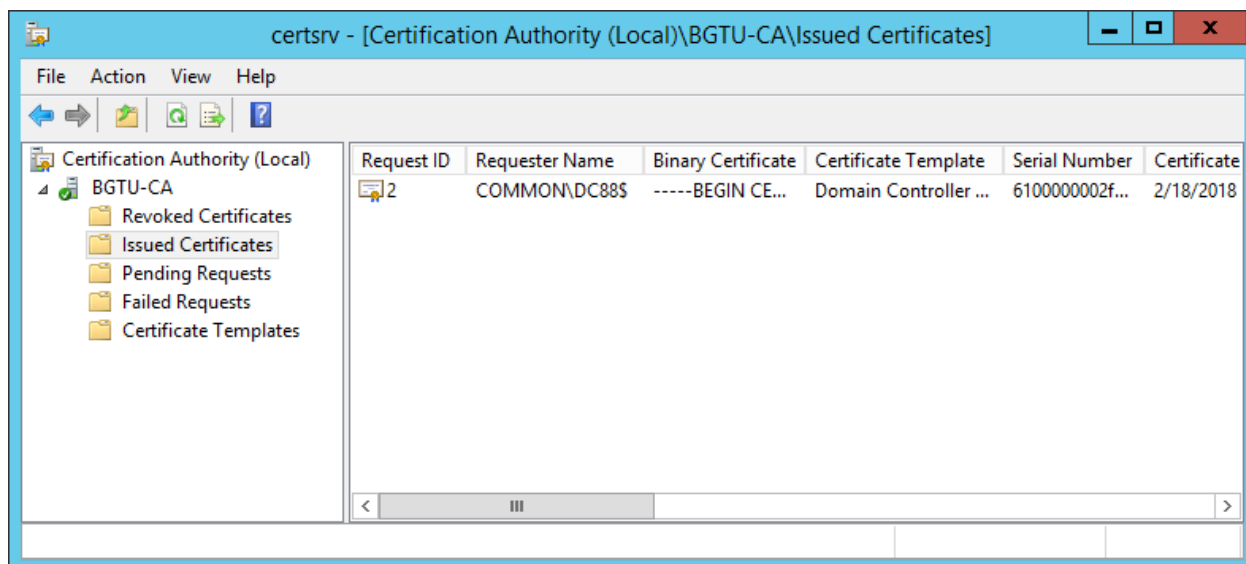


Рис. 5.11. Консоль **Certification Authority**

Также имеется утилита командной строки `certutil` и командлеты PowerShell.

При установке центра сертификации предприятия появляется возможность упростить управление сертификатами на основе шаблонов сертификатов.

5.3.4. Шаблоны сертификатов

Шаблоны сертификатов позволяют администраторам настраивать метод распространения сертификатов, определять назначение сертификатов и задавать тип использования, разрешенный сертификатом. Администраторы могут создавать шаблоны, а затем быстро разворачивать их на предприятии с помощью встроенного графического интерфейса пользователя (GUI) или инструментов командной строки.

Связанным с каждым шаблоном сертификата является его избирательный список контроля доступа (DACL). DACL определяет, какие участники безопасности имеют разрешения на чтение и настройку шаблона и какие участники безопасности могут получать вручную или автоматически новые сертификаты на основе шаблона. Шаблоны сертификатов и их разрешения определяются в AD DS и действительны в всем лесу. Если в лесу AD DS работает несколько CA, изменения прав будут влиять на все ЦС.

Когда вы определяете шаблон сертификата, определение шаблона сертификата должно быть доступно для всех центров сертификации в лесу. Для этого информация о шаблоне сертификата хранится в разделе конфигурации AD DS. Репликация этой информации зависит от расписания репликации AD DS, и шаблон сертификата может быть недоступен для всех центров сертификации до завершения репликации. Хранение и репликация происходят автоматически.

Версии шаблонов сертификатов

Служба сертификации Active Directory (AD CS) в Windows Server 2016 поддерживает четыре версии шаблонов сертификатов. Помимо соответствующих версий операционной системы Windows Server версии шаблонов сертификатов также имеют некоторые функциональные отличия, как описано ниже.

- **Версия 1.** Единственной модификацией, разрешенной для шаблонов версии 1, является возможность изменения разрешений на чтение, запись,

разрешение или запрет регистрации шаблона сертификата. Когда вы устанавливаете СА, по умолчанию создаются шаблоны сертификатов 1-й версии.

- **Версия 2.** Вы можете настроить несколько параметров в шаблонах версии 2. Установка по умолчанию AD CS предоставляет несколько предварительно сконфигурированных шаблонов версии 2. Вы также можете создавать шаблоны версии 2 для нужд своей организации. Кроме того, вы можете дублировать шаблон сертификата версии 1 для создания нового шаблона версии 2. Затем вы можете изменить вновь созданный шаблон сертификата версии 2. Шаблоны должны быть минимум версии 2 для поддержки автоматической подачи заявки (autoenrollment).

- **Версия 3.** Шаблоны сертификатов версии 3 поддерживают криптографию следующего поколения (CNG). CNG обеспечивает поддержку криптографических алгоритмов Suite B, таких как криптография с использованием эллиптических кривых. Вы можете дублировать шаблоны по умолчанию версии 1 и версии 2, чтобы обновить их до версии 3. При использовании шаблонов сертификатов версии 3 вы можете использовать алгоритмы шифрования и хеширования CNG для запросов сертификатов, выданных сертификатов и защиты закрытых ключей при обмене ключами и архивировании ключей.

- **Версия 4.** Шаблоны сертификатов версии 4 доступны только для операционных систем Windows Server 2012, Windows 8 и более поздних версий. Чтобы помочь администраторам определить, какие версии операционной системы поддерживают какие функции, на вкладке Свойства шаблона сертификата была добавлена вкладка Совместимость. Он указывает параметры как недоступные в свойствах шаблона сертификата, в зависимости от выбранных версий операционной системы клиента сертификата и ЦС. Шаблоны сертификатов версии 4 также поддерживают как поставщиков криптографических услуг (CSP), так и поставщиков хранилищ ключей. Вы также можете настроить их для обновления с помощью одного и того же ключа.

5.4. Методы аутентификации в службе Internet Information Services

Аутентификация — это процесс проверки, действительно ли пользователь является тем, за кого себя выдает. В IIS поддерживается множество методов аутентификации.

- **Анонимная Аутентификация (Anonymous Authentication).** Пользователи могут подключаться к веб-сайту без предъявления своих полномочий.
- **Аутентификация на основе клиентских сертификатов Active Directory (Active Directory Client Certificate Authentication).** Пользователи могут подключаться к веб-сайту с помощью аутентификации своих клиентских сертификатов Active Directory.
- **Заимствование прав ASP.NET (ASP.NET Impersonation).** Пользователи могут использовать для аутентификации учетную запись ASP.NET.
- **Аутентификация Windows (Windows Authentication).** Этот метод может интегрироваться с Active Directory. После входа пользователей в систему вместо пароля передается значение его хеша.
- **Дайджест-аутентификация (Digest Authentication).** Этот метод похож на предыдущий: здесь тоже передается хеш пароля, но для проверки его достоверности необходим контроллер домена Windows Server.
- **Базовая аутентификация (Basic Authentication).** Имя и пароль пользователей передаются по сети в виде открытого текста, из-за чего этот метод считается недостаточно защищенным от несанкционированного доступа и обычно применяется в сочетании с защитой сайта или страницы с помощью SSL.
- **Аутентификация с помощью форм (Forms Authentication).** Пользователи перенаправляются на специальную страницу для ввода учетных данных. После прохождения аутентификации они перенаправляются на страницу, которую запрашивали изначально.

Все эти методы аутентификации включаются на странице компонента Authentication (Аутентификация), которая показана на рис. 5.12. Для ее отображения необходимо выбрать данный компонент в разделе IIS на нужном сервере, сайте или в виртуальном каталоге.

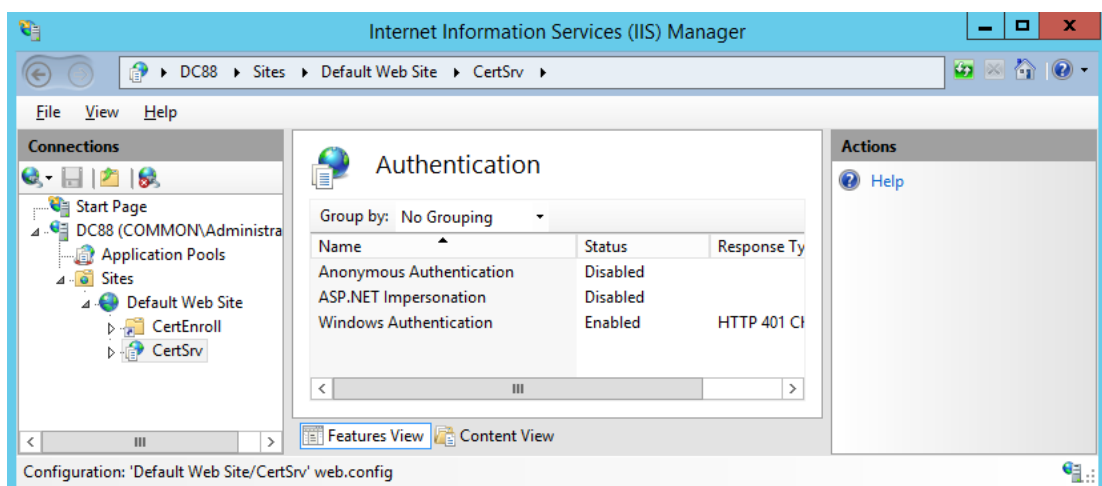


Рис. 5.12. Компонент Authentication в консоли IIS Manager