Технология аутентификации

Применение открытых каналов передачи данных создает потенциальные возможности для действий злоумышленников (нарушителей). Поэтому одной из важных задач обеспечения информационной безопасности при взаимодействии пользователей является использование методов и средств, позволяющих одной (проверяющей) стороне убедиться в подлинности другой (проверяемой) стороны

С каждым зарегистрированным в компьютерной системе субъектом (пользователем или процессом, действующим от имени пользователя) связана некоторая информация, однозначно идентифицирующая его. Это может быть число или строка символов, именующие данный субъект. Эту информацию называют *идентификатором* субъекта. Если пользователь имеет идентификатор, зарегистрированный в сети, он считается легальным (законным) пользователем; остальные пользователи относятся к нелегальным пользователям. Прежде чем получить доступ к ресурсам компьютерной системы, пользователь должен пройти процесс первичного взаимодействия с компьютерной системой, который включает идентификацию и аутентификацию.

Идентификация (*Identification)* — процедура распознавания пользователя по его идентификатору (имени). Эта функция выполняется, когда пользователь делает попытку войти в сеть. Пользователь сообщает системе по ее запросу свой идентификатор, и система проверяет в своей базе данных его наличие.

Аутентификация (Authentication) — процедура проверки подлинности заявленного пользователя, процесса или устройства. Эта проверка позволяет достоверно убедиться, что пользователь (процесс или устройство) является именно тем, кем себя объявляет. При проведении аутентификации проверяющая сторона убеждается в подлинности проверяемой стороны, при этом проверяемая сторона тоже активно участвует в процессе обмена информацией. Обычно пользователь подтверждает свою идентификацию, вводя в систему уникальную, не известную другим пользователям информацию о себе (например, пароль или сертификат).

Идентификация и аутентификация являются взаимосвязанными процессами распознавания и проверки подлинности субъектов (пользователей). Именно от них зависит последующее решение системы: можно ли разрешить доступ к ресурсам системы конкретному пользователю или процессу. После идентификации и аутентификации субъекта выполняется его авторизация.

Авторизация (.Authorization) — процедура предоставления субъекту определенных полномочий и ресурсов в данной системе. Иными словами, авторизация устанавливает сферу его действия и доступные ему ресурсы. Если система не может надежно отличить авторизованное лицо от неавторизованного, то конфиденциальность и целостность информации в этой системе могут быть нарушены. Организации необходимо четко определить свои требования к безопасности, чтобы принимать решения о соответствующих границах авторизации.

С процедурами аутентификации и авторизации тесно связана процедура администрирования действий пользователя.

Администрирование (.Accounting) — регистрация действий пользователя в сети, включая его попытки доступа к ресурсам. Хотя эта учетная информация может быть использована для выписывания счета, с позиций безопасности она особенно важна для обнаружения, анализа инцидентов безопасности в сети и соответствующего реагирования на них. Записи в системном журнале, аудиторские проверки и ПО accounting — все это может быть использовано для обеспечения подотчетности пользователей, если что-либо случится при входе в сеть с их идентификатором.

Необходимый уровень аутентификации определяется требованиями безопасности, которые установлены в организации. Общедоступные Web-серверы могут разрешить анонимный или гостевой доступ к информации. Финансовые транзакции могут потребовать строгой аутентификации. Примером слабой формы аутентификации может служить использование IP-адреса для определения пользователя. Подмена (spoofing) IP-адреса может легко разрушить механизм аутентификации. Надежная аутентификация является тем ключевым фактором, который гарантирует, что только авторизованные пользователи получат доступ к контролируемой информации.

При защите каналов передачи данных должна выполняться *взаимная аутентификация субъектов*, т. е. взаимное подтверждение подлинности субъектов, связывающихся между собой по линиям связи. Процедура подтверждения подлинности выполняется обычно в начале сеанса установления соединения абонентов. Термин «соединение» указывает на логическую связь (потенциально двустороннюю) между двумя субъектами сети. Цель данной процедуры — обеспечить уверенность, что соединение установлено с законным субъектом и вся информация дойдет до места назначения.

Для подтверждения своей подлинности субъект может предъявлять системе разные сущности. В зависимости от предъявляемых субъектом сущностей процессы аутентификации могут быть разделены на основе:

- *знания* чего-либо. Примерами могут служить пароль, персональный идентификационный код PIN (Personal Identification Number), а также секретные и открытые ключи, знание которых демонстрируется в протоколах типа запрос-ответ;
- *обладания* чем-либо. Обычно это магнитные карты, смарт-карты, сертификаты и устройства *touch memory*;
- каких-либо неотъемлемых *характеристик.* Эта категория включает методы, базирующиеся на проверке биометрических характеристик пользователя (голоса, радужной оболочки и сетчатки глаза, отпечатков пальцев, геометрии ладони и др.). В данной категории не используются криптографические методы и средства. Аутентификация на основе биометрических характеристик применяется для контроля доступа в помещения или к какой-либо технике

Пароль — это то, что знает пользователь и другой участник взаимодействия. Для взаимной аутентификации участников взаимодействия может быть организован обмен паролями между ними.

Персональный идентификационный номер PIN (Personal Identification Number) является испытанным способом аутентификации держателя пластиковой карты и смарт-карты. Секретное значение PIN-кода должно быть известно только держателю карты.

Динамический (*одноразовый) пароль* — это пароль, который после однократного применения никогда больше не используется. На практике обычно используется регулярно меняющееся значение, которое базируется на постоянном пароле или ключевой фразе.

Система запрос—ответ. Одна из сторон инициирует аутентификацию с помощью посылки другой стороне уникального и непредсказуемого значения «запрос», а другая сторона посылает ответ, вычисленный с помощью «запроса» и секрета. Так как обе стороны владеют одним секретом, то первая сторона может проверить правильность ответа второй стороны.

Сертификаты и цифровые подписи. Если для аутентификации используются сертификаты, то требуется применение цифровых подписей на этих сертификатах. Сертификаты выдаются ответственным лицом в организации пользователя, сервером сертификатов или внешней доверенной организацией. В рамках Интернета появились коммерческие инфраструктуры управления открытыми ключами PKI (Public Key Infrastructure) для распространения сертификатов открытых ключей. Пользователи могут получить сертификаты различных уровней.

Процессы аутентификации можно также классифицировать по уровню обеспечиваемой безопасности [9, 54]. В соответствии с этим процессы аутентификации разделяются на следующие типы:

• аутентификация, использующая пароли и PIN-коды;

• строгая аутентификация на основе использования криптографических методов и средств;

• биометрическая аутентификация пользователей.

С точки зрения безопасности каждый из перечисленных типов способствует решению своих специфических задач, поэтому процессы и протоколы аутентификации активно используются на практике.

Основные атаки на протоколы аутентификации:

• *маскарад* (*impersonation*). Пользователь выдает себя за другого с целью получения полномочий и возможности действий от лица другого пользователя;

• *подмена стороны* аутентификационного обмена (*interleaving attack).* Злоумышленник в ходе данной атаки участвует в процессе аутентификационного обмена между двумя сторонами с целью модификации проходящего через него трафика;

• *повторная передача {replay attack)* заключается в повторной передаче аутентификационных данных каким-либо пользователем;

• *принудительная задержка {forced delay).* Злоумышленник перехватывает некоторую информацию и передает ее спустя некоторое время;

• *атака с выборкой текста {chosen-text attack).* Злоумышленник перехватывает аутентификационный трафик и пытается получить информацию о долговременных криптографических ключах.

Для предотвращения таких атак при построении протоколов аутентификации применяются:

• использование механизмов типа «запрос—ответ», «отметка времени», случайных чисел, идентификаторов, цифровых подписей;

• привязка результата аутентификации к последующим действиям пользователей в рамках системы. Примером подобного подхода может служить осуществление в процессе аутентификации обмена секретными сеансовыми ключами, которые используются при дальнейшем взаимодействии пользователей;

• периодическое выполнение процедур аутентификации в рамках уже установленного сеанса связи и т. п.

Механизм «запрос—ответ» состоит в следующем. Если пользователь *Л* хочет быть уверенным, что сообщения, получаемые им от пользователя *В*, не являются ложными, он включает в посылаемое для *В* сообщение непредсказуемый элемент — запрос *X* (например, некоторое случайное число). При ответе пользователь *В* должен выполнить над этим элементом некоторую операцию (например, вычислить некоторую функцию/(ЛО). Это невозможно осуществить заранее, так как пользователю *В* неизвестно, какое случайное число *X* придет в запросе. Получив ответ с результатом действий *В*, пользователь *Л* может быть уверен, что *В* — подлинный. Недостаток этого метода — возможность установления закономерности между запросом и ответом.

Механизм «отметка времени» подразумевает регистрацию времени для каждого сообщения. В этом случае каждый пользователь сети определяет, насколько «устарело» пришедшее сообщение, и решает не принимать его, поскольку оно может быть ложным.

В обоих случаях для защиты механизма контроля следует применять шифрование, чтобы быть уверенным, что ответ послан не злоумышленником.

При использовании отметок времени возникает проблема *допустимого временного интервала задержки* для подтверждения подлинности сеанса: сообщение с «временным штемпелем» в принципе не может быть передано мгновенно. Кроме того, компьютерные часы получателя и отправителя не могут быть абсолютно синхронизированы.

При сравнении и выборе протоколов аутентификации необходимо учитывать следующие характеристики:

• наличие *взаимной аутентификации.* Это свойство отражает необходимость обоюдной аутентификации между сторонами аутентификационного обмена;

• *вычислительную эффективность.* Это количество операций, необходимых для выполнения протокола;

• *коммуникационную эффективность.* Данное свойство отражает количество сообщений и их длину, необходимую для осуществления аутентификации;

• *наличие третьей стороны.* Примером третьей стороны может служить доверенный сервер распределения симметричных ключей или сервер, реализующий дерево сертификатов для распределения открытых ключей;

• *гарантии безопасности.* Примером может служить применение шифрования и цифровой подписи

Одной из распространенных схем аутентификации является *простая аутентификация*, которая основана на применении традиционных многоразовых паролей с одновременным согласованием средств его использования и обработки. Аутентификация на основе многоразовых паролей — простой и наглядный пример использования разделяемой информации. Пока в большинстве защищенных виртуальных сетей VPN (Virtual Private Network) доступ клиента к серверу разрешается по паролю. Однако все чаще применяются более эффективные средства аутентификации, например программные и аппаратные системы аутентификации на основе одноразовых паролей, смарт-карт, PIN-кодов и цифровых сертификатов

Аутентификация на основе многоразовых паролей

Базовый принцип «единого входа» предполагает достаточность одноразового прохождения пользователем процедуры аутентификации для доступа ко всем сетевым ресурсам. Поэтому в современных операционных системах предусматривается централизованная служба аутентификации, которая выполняется одним из серверов сети и использует для своей работы базу данных (БД). В этой БД хранятся учетные данные о пользователях сети, включающие идентификаторы и пароли пользователей, а также другую информацию [45].

Процедуру простой аутентификации пользователя в сети можно представить следующим образом. Пользователь при попытке логического входа в сеть набирает свои идентификатор и пароль. Эти данные поступают для обработки на сервер аутентификации. В БД, хранящейся на сервере аутентификации, по идентификатору пользователя находится соответствующая запись. Из нее извлекается пароль и сравнивается с тем паролем, который ввел пользователь. Если они совпали, то аутентификация прошла успешно — пользователь получает легальный статус и получает те права и ресурсы сети, которые определены для его статуса системой авторизации

Схемы организации простой аутентификации отличаются не только методами передачи паролей, но и видами их хранения и проверки. Наиболее распространенным способом является хранение паролей пользователей в открытом виде в системных файлах, причем на эти файлы устанавливаются атрибуты защиты от чтения и записи (например, при помощи описания соответствующих привилегий в списках контроля доступа ОС). Система сопоставляет введенный пользователем пароль с хранящейся в файле паролей записью. При этом способе не используются криптографические механизмы, такие как шифрование или однонаправленные функции. Очевидным недостатком этого способа является возможность получения злоумышленником в системе привилегий администратора, включая права доступа к системным файлам, и в частности, к файлу паролей.

Для обеспечения надежной защиты ОС пароль каждого пользователя должен быть известен только этому пользователю и никому другому, в том числе и администраторам системы. На первый взгляд то, что администратор знает пароль некоторого пользователя, не отражается негативно на безопасности системы, поскольку администратор, войдя в систему от имени обычного пользователя, получает права меньшие чем те, которые он получит, зайдя в систему от своего имени. Однако, входя в систему от имени другого пользователя, администратор получает возможность обходить систему аудита, а также совершать действия, компрометирующие этого пользователя, что недопустимо в защищенной системе. Таким образом, пароли пользователей не должны храниться в ОС в открытом виде.

С точки зрения безопасности предпочтительным является метод передачи и хранения паролей с использованием односторонних функций. Обычно для шифрования паролей в списке пользователей используют одну из известных криптографически стойких *хэш-функций.* В списке пользователей хранится не сам пароль, а *образ пароля,* являющийся результатом применения к паролю хэш-функции.

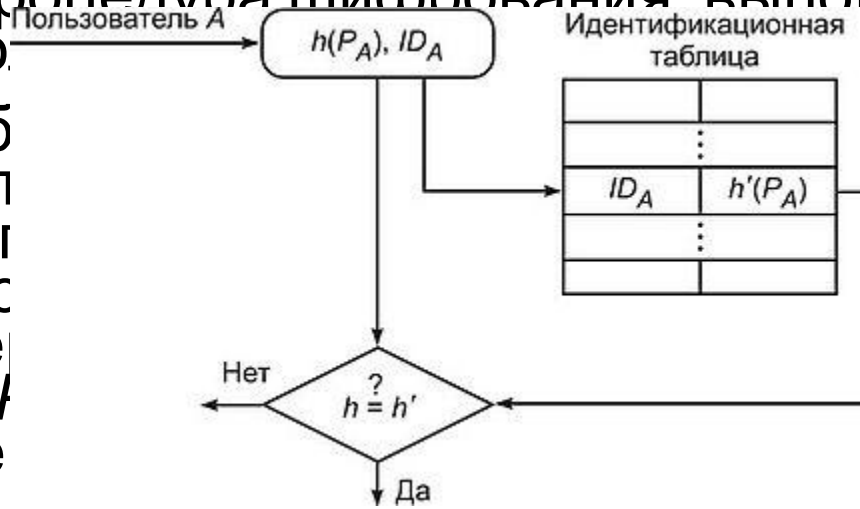
Однонаправленность хэш-функции не позволяет восстановить пароль по образу пароля, но позволяет, вычислив хэш-функцию, получить образ введенного пользователем пароля и таким образом проверить правильность введенного пароля. В простейшем случае в качестве хэш-функции используется результат шифрования некоторой константы на пароле.

Например, односторонняя функция /?(•) может быть определена следующим образом:

$КР) = E_P(Ю),$

где *Р* — пароль пользователя; *Ю* — идентификатор пользователя; E_P — процедура шифрования, выполняемая с использованием паро

Такие функции удоб на одинаковы. В этом случае проверка подл мощью пароля P_A состоит из п икации отображения $И(P_A)$ и с но вычисленным и хранимым в БД серве нтом $И'(P_A)$ (рис. 7.2). Если отображения $И(P$ ся, что пользователь успешно прошел ауте



На практике пароли состоят лишь из нескольких символов, чтобы дать возможность пользователям запомнить их. Короткие пароли уязвимы к атаке полного перебора всех вариантов. Для того чтобы предотвратить такую атаку, функцию *И(Р)* можно определить иначе, например в виде:

$h(P) - E_{РФК}(Ю)$,

где *К и ID* — соответственно ключ и идентификатор отправителя.

Различают две формы представления объектов, аутентифицирующих пользователя:

- внешний аутентифицирующий объект, не принадлежащий системе;
- внутренний объект, принадлежащий системе, в который переносится информация из внешнего объекта

Внешние объекты могут быть представлены на различных носителях информации: пластиковых картах, смарт-картах, гибких магнитных дисках и т. п. Естественно, что внешняя и внутренняя формы представления аутентифицирующего объекта должны быть семантически тождественны.

Системы простой аутентификации на основе многоразовых паролей имеют пониженную стойкость, поскольку выбор аутентифицирующей информации происходит из относительно небольшого числа слов. Срок действия многоразового пароля должен быть определен в политике безопасности организации. Пароли должны регулярно изменяться, быть трудными для угадывания и не присутствовать в словаре.

Аутентификация на основе одноразовых паролей

Схемы аутентификации, основанные на традиционных многоразовых паролях, не обладают достаточной безопасностью. Такие пароли можно перехватить, разгадать, подсмотреть или просто украсть. Более надежными являются процедуры аутентификации на основе одноразовых паролей.

Суть схемы одноразовых паролей — использование различных паролей при каждом новом запросе на предоставление доступа. Одноразовый динамический пароль действителен только для одного входа в систему, и затем его действие истекает. Даже если его перехватили, он будет бесполезен. Динамический механизм задания пароля — один из лучших способов защиты процесса аутентификации от угроз извне. Обычно системы аутентификации с одноразовыми паролями используются для проверки удаленных пользователей

Генерация одноразовых паролей может осуществляться аппаратным или программным способом. Некоторые аппаратные средства доступа на основе одноразовых паролей реализуются в виде миниатюрных устройств со встроенным микропроцессором, внешне похожих на платежные пластиковые карточки. Такие карты, обычно называемые ключами, могут иметь клавиатуру и небольшое дисплейное окно.

В качестве примера рассмотрим технологию аутентификации SecurID на основе одноразовых паролей с использованием аппаратных ключей и механизма временной синхронизации. Эта технология разработана компанией Security Dynamics и реализована в коммуникационных серверах ряда компаний, в частности в серверах компании Cisco Systems и др.

Схема аутентификации с использованием временной синхронизации базируется на алгоритме генерации случайных чисел через определенный интервал времени. Этот интервал устанавливается и может быть изменен администратором сети. Схема аутентификации использует два параметра:

- секретный ключ, представляющий собой уникальное 64-битное число, назначаемое каждому пользователю и хранящееся в БД аутентификационного сервера и в аппаратном ключе пользователя;
- значение текущего времени.

Когда удаленный пользователь делает попытку логического входа в сеть, ему предлагается ввести его персональный идентификационный номер PIN, состоящий из четырех десятичных цифр, и шесть цифр случайного числа, отображаемого в этот момент на дисплее аппаратного ключа. Используя введенный пользователем PIN-код, сервер извлекает из БД секретный ключ пользователя и выполняет алгоритм генерации случайного числа, используя в качестве параметров извлеченный секретный ключ и значение текущего времени. Затем сервер проверяет, совпадают ли сгенерированное число и число, введенное пользователем. Если эти числа совпадают, то сервер разрешает пользователю осуществить логический вход в систему.

При использовании этой схемы аутентификации требуется жесткая временная синхронизация аппаратного ключа и сервера. Со схемой аутентификации, основанной на временной синхронизации, связана еще одна проблема. Генерируемое аппаратным ключом случайное число является достоверным паролем в течение небольшого конечного промежутка времени. Поэтому возможна кратковременная ситуация, когда можно перехватить PIN-код и случайное число, чтобы использовать их для доступа в сеть. Это — уязвимое место схемы