Аутентификация и авторизация.

Аутентификация - процесс проверки подлинности данных о пользователе.

Аутентификация бывает однофакторная и многофакторная.

Единый вход в систему

Развитие цифровых технологий требует от современного пользователя множество пользовательских идентификаторов, что усложняет работу.

«Единый вход в систему» - термин определяющий возможность применения единых аутентификационных параметров в различных системах.

Для поддержки такой возможности могут применяться смарт-карты, маркеры доступа и центральные базы аутентификации.

Достоинство данной системы.

- Облегчается работа пользователей.
- Уменьшается риск доступа сторонних лиц к паролям.

Недостатки.

- Необходимость администрирования аутентификационной базы данных.
- При едином пароле на все системы повышается ущерб в случае кражи пароля.

Существующие системы аутентификации

- Системы, использующие сочетание имен и паролей, в том числе Kerberos.
- Системы, использующие сертификаты и маркеры доступа.
- Биометрические системы.

Имена пользователей и пароли

- Пароль хорош тем, что без него сложно проникнуть в систему, но как только его узнает злоумышленник, то возникает угроза.

- При выборе парольной системы необходимо ориентироваться не только на алгоритм, но и на механизм управления защитой паролей.

Системы парольной аутентификации

- Локальное хранение и сравнение.
- Централизованное хранение и сравнение.
- Системы типа «вопрос-ответ».
- Kerberos.
- Системы с одноразовыми паролями.

Локальное хранение и сравнение

Изначально пароли хранились в базах данных в открытом виде.

В дальнейшем начались попытки зашифровать пароли и скрыть файлы, содержащие их.

Существует множество программ (в свободном доступе), позволяющих снимать парольную защиту с операционных систем Windows и Linux.

Например: LC4 (LophtCrack) и John the Ripper.

Они основываются на сочетании словарной атаки, эвристического анализа (наиболее часто создаваемые пароли) и грубой силы (полный перебор всех символов).

Поэтому необходимо контролировать процесс аутентификации.

Средства контроля аутентификации

- Длина пароля. Максимально количество символов в пароле ограничивается средствами ОС. На минимальное ограничений, как правило, не бывает. Рекомендуется не менее семи или восьми символов.
- Короткий пароль легко сломать, а длинный трудно запомнить.

- Сложность паролей и фильтры. При создании пароля выполняется проверка на длину и устойчивость (требование на наличие символов верхнего и нижнего регистров, цифр, специальных символов).

- История паролей. При смене пароля у пользователей возникает желание работать со старыми паролями. Данный механизм запрещает использование старых паролей (для каждого пользователя в истории может храниться 9 -15 старых паролей).

- Максимальный возраст пароля. Необходимо менять пароли по расписанию. Для этого указывается время, через которое пароль д. б. заменен. Рекомендуемое значение - 30 дней.

- Минимальный возраст пароля. Невозможность изменения пароля до истечения некоторого минимального числа дней. Направлено против хитрых пользователей переполняющих буфер истории, что позволяет использовать старый пароль. Рекомендуется устанавливать срок не менее 5 дней.

- Истечение срока действия учетных записей. Если сотрудник уволен, а его учетная запись активна, то возникает угроза безопасности. При этом, если персонала много, то регулярное обновление учетных записей приведет к проблемам управления. Можно создавать временные учетные записи только для сотрудников, имеющих срочные договора.

- Ограничения учетных записей. Ограничение доступа пользователей к системам на некоторое время суток.

- Блокировка учетных записей. При выполнении атаки на определенные учетные записи, злоумышленник сможет через некоторое время установить аутентификацитонные параметры. Для исключения такой угрозы можно установить параметры блокировки учетной записи после ряда неудачных попыток входа. Эффективно, если атака направлена на одну учетную запись. В противном случае возникает Dos.

Централизованное хранение

Общий алгоритм.

- Система принимает пароль пользователя в открытом виде.
- Пароль шифруется.
- Зашифрованный пароль передается на сервер и сравнивается с зашифрованным паролем.

Некоторые системы передают пароль в открытом виде (telnet, FTP, PAP). Это дает возможность перехвата паролей.

Система «вопрос-ответ» Windows LAN Manager

Эта система используется в старых ОС – Windows 2000 или Windows server 2003.

- Аутентификация выполняется при начальном входе в систему или при доступе к ресурсу.
- Создано три версии системы, но они отличаются только способом защиты пароля и размером ключевого пространства.

Шаги аутентификации.

- Ввод пароля.
- Запрос аутентификации на сервер.
- Генерация сервером «вопроса» в виде случайного числа.
- Клиент шифрует хэш-кодом пароля полученное число, и данный код передается серверу (ответ).
- Сервер с помощью хэш-кода пароля расшифровывает «ответ». Если «вопрос» и «ответ» совпали, то аутентификация прошла успешно.

На клиенте и сервере хэширование выполняется по одинаковому алгоритму.

- Первая версия LM считается ненадежной, т.к.
- Ключ может иметь длину не более 14 символов.
- Недостающая часть пароля заполняется пробелами.
- Ключ может содержать только символы верхнего регистра, спец символы и числа.
- Пароль делится на части по 7 символов после чего эти части используются для шифрования. Каждая часть пароля может быть атакована отдельно.

Система NTLM для Windows NT

Система NTLM для Windows NT более надежна.

- Пароль может иметь длину до 128 символов (стандартный графический интерфейс поддерживает 14).

- Пароль хешируется с использованием MD4 с формированием 16 байтового кода.

- Полученный криптографический хэш является односторонним и его нельзя расшифровать.

Недостатки NTLM.

- Перехваченный ответ может быть использован злоумышленником (атака воспроизведения).

CHAP и MS-CHAP

- Протокол Challenge Handshake Authentication Protocol и MS-CHAP предназначены для удаленной аутентификации.

- Используется хэш код пароля для шифрования строки вопроса.

CHAP сохраняет пароль пользователя в обратимо зашифрованном виде, а MS-CHAP нет.

- MS-CHAPv2 требует двусторонней аутентификации.

Пользователь аутентифицируется на сервере, а затем сервер предоставляет доказательство своей аутентичности.

Сервер шифрует вопрос отправленный клиентом. Это делается с помощью пароля, хранящегося в б. д. учетных записей сервера, что еще более повышает надежность алгоритма.

Kerberos. Управление ключами

В протоколе Kerberos присутствует три участника безопасной связи: клиент, сервер и доверенный посредник между ними - центр распределения ключей Key Distribution Center (KDC).

- KDC служба, работающая на физически защищенном сервере, которая ведет базу данных с информацией об учетных записях всех абонентов безопасности своей области (домена).
- В базе данных KDC сохраняется криптографический ключ, известный только абоненту и службе KDC.
- Данный ключ называется долговременным и используется для связи пользователя системы безопасности с центром распределения ключей (долговременные ключи создаются на основе пароля пользователя).

- Когда клиенту нужно обратиться к серверу, он, направляет запрос в центр KDC, который в ответ направляет каждому участнику предстоящего сеанса копии уникального сеансового ключа (session key), действующие в течение короткого времени.

- Назначение этих ключей – проведение аутентификации клиента и сервера. Копия сеансового ключа, пересылаемая на сервер, шифруется с помощью долговременного ключа этого сервера, а направляемая клиенту – долговременного ключа данного клиента.

Но ключи могут прийти неодновременно и тогда возникнут проблему аутентификации.

Kerberos. Сеансовые билеты

- В ответ на запрос клиента, желающего подключиться к серверу, служба KDC направляет обе копии сеансового ключа клиенту.

- Сообщение, предназначенное клиенту, шифруется долговременным ключом клиента, а сеансовый ключ для сервера вместе с информацией о клиенте вкладывается в блок данных, получивший название сеансового билета (session ticket).
- Сеансовый билет целиком шифруется с помощью долговременного ключа сервера, который знают только служба KDC и данный сервер.

- Получив сообщение, клиент должен доставить зашифрованный сеансовый билет на сервер.

- Расшифровать клиентскую копию сеансового ключа может только тот, кто знает секретный долговременный ключ данного клиента.
- Расшифровать содержимое сеансового билета можно только долговременным секретным ключом сервера.

- Получив ответ KDC, клиент извлекает из него сеансовый билет и свою копию сеансового ключа.
- Когда возникает необходимость связаться с сервером, клиент посылает ему сообщение, состоящее из билета, и собственного аутентификатора, зашифрованного посредством сеансового ключа. Этот билет в комбинации с аутентификатором как раз и составляет удостоверение, по которому сервер определяет клиента.

Kerberos. Сеансовые билеты

- Сервер, получив удостоверение клиента, с помощью своего секретного ключа расшифровывает сеансовый билет и извлекает из него сеансовый ключ, который затем использует для расшифрования аутентификатора клиента.
- Клиент может потребовать у сервера проведения взаимной аутентификации. В этом случае сервер с помощью своей копии сеансового ключа шифрует метку времени из аутентификатора клиента и в таком виде пересылает ее клиенту в качестве собственного аутентификатора.

Преимущества сеансовых билетов.

- Серверу не нужно хранить сеансовые ключи для связи с клиентами.
- Клиенту не надо обращаться к KDC перед каждым сеансом связи с конкретным сервером. Сеансовые билеты можно использовать многократно.
- На случай хищения устанавливается срок годности билета, который KDC указывает в самой структуре данных. Это время определяется политикой Kerberos для конкретного домена (обычно не более 8 часов).

Этапы аутентификации Kerberos

1. Пользователь вводит пароль.
2. Данные (аутентификатор) передаются на сервер. Аутентификатор и копия временного штампа вместе с запросом в открытом виде передаются на сервер Kerberos (KDC) (сообщение KRB_AS_REQ).
3. KDC сравнивает временной штамп, полученный от клиента со своим временем. Разница не должна выходить за рамки 5 минут. В противном случае запрос отклоняется.
4. KDC шифрует временной штамп паролем пользователя и сравнивает с аутентификатором. При совпадении пользователь проходит аутентификацию и получает билет на выдачу билетов (ticket-granting ticket - TGT) в виде сообщения KRB_AS_REP.
5. Клиент отправляет TGT в KDC с запросом конкретного ресурса и прилагает «свежий» аутентификатор (KRB_TGS_REQ).
6. KDC подтверждает аутентификатор и проверяет TGT (штамп времени защищает от перехвата).
7. KDC создает билет службы для запрошенного ресурса (KRB_TGS_REP). Часть билета шифруется аутентификационными данными клиента, а часть данными сервера.
8. Клиент расшифровывает свою часть, определяет доступный сервер и передает туда билет вместе со «свежим» аутентификатором.
9. Сервер проверяет временной штамп на попадание в диапазон, а затем расшифровывает свою часть билета.

Системы с одноразовыми паролями

- Требуется ввод нового пароля при каждой последующей аутентификации.
- При этом возможно использовать текущий пароль для генерирования следующего.
- Такой принцип реализован в RSA SequrID и S/Key.

RSA SequrID

- Каждые 60 сек генерируется одноразовый код аутентификации. Пользователь комбинирует свой PIN и полученный код для создания паролей.
- Подтверждение пароля основывается на синхронизированных часах и PIN пользователя.

- Это пример двухфакторной аутентификации.

S/Key

Для генерации паролей используется идентификационная фраза и число, определяющее количество паролей, которые будут получены из этой фразы.

При каждом запросе генерируется новый пароль.

Шаги аутентификации.

1. Пользователь вводит идентификационную фразу.
2. Клиентская машина выполняет запрос аутентификации.
3. Сервер генерирует вопрос.
4. Генератор на сервере и клиенте генерируют одинаковый одноразовый пароль.
5. Сгенерированный пароль используется для шифрования ответа.
6. Ответ передается на сервер.
7. Сервер сравнивает ответ с шифром вопроса и пароля.

Аутентификация по сертификатам

Сертификат – набор данных, сопоставляющий пользователя с парой ключей – открытый/закрытый.

- Секретный ключ используется для шифрования и цифровой подписи, а для расшифровки используется открытый ключ.

Шаги аутентификации.

1. Клиент подает запрос на аутентификацию.
2. Сервер создает вопрос.
3. Клиент шифрует вопрос секретным ключом.
4. Возвращает ответ (зашифрованный) вопрос серверу.
5. Сервер имеет копию сертификата и расшифровывает ответ открытым ключом.
6. Выполняется сравнение с вопросом.
7. При совпадении аутентификация прошла успешно.

Смарт-карты

- Смарт-карта позволяет отделить секретный ключ от компьютера.
- Смарт-карты являются стойкими к словарным атакам, так как имеется ограниченное число неправильных попыток ввода PIN.

Проблемы

- процесс создания
- обучение пользователей
- оборудование.

Биометрическая аутентификация

- Биометрическая аутентификация является самой надежной двухфакторной аутентификацией. То, что вы имеете, является неотъемлемой частью вашего организма.

Для выполнения такой аутентификации используются специализированные модули распознавания

- по лицу
- радужной оболочка глаза
- сетчатке глаза
- отпечаткам пальцев
- голосу
- нажатию клавиш