ЛЕКЦИЯ №3

«Идентификация и аутентификация»

по дисциплине

«Безопасность операционных систем»

Текст лекции рассмотрен и одобрен на заседании кафедры протокол № _____ от " " 201__ г.

(Слайд 1. Титульный слайд)

Уважаемые студенты! Сегодня вы продолжаете изучение дисциплины «Безопасность операционных систем». Лекция №3 «Идентификация и аутентификация». Продолжительность лекции - 4 академических часа.

Слайд 2 (план проведения занятия)

Наш курс состоит из лекций и практических занятий по отдельным подсистемам и механизмам безопасности, реализованным в ОС.

В данной лекции мы начнем изучать "Идентификацию и аутентификацию". В частности: Понятие идентификации и аутентификации. Методы аутентификации и их реализация в современных ОС. Протоколы аутентификации. Аутентификация на основе паролей, методы подбора паролей, средства и методы повышения защищенности ОС от подбора паролей. Аутентификация на основе внешних носителей ключа, биометрических характеристик пользователя.

Понятие идентификации и аутентификации.

Идентификацию и аутентификацию можно считать основой программно-технических средств безопасности, поскольку остальные сервисы рассчитаны на обслуживание именованных субъектов.

Идентификация и аутентификация – это первая линия обороны, "проходная" информационного пространства организации.

Идентификация позволяет субъекту (пользователю, процессу, действующему от имени определенного пользователя, или иному аппаратно-программному компоненту) назвать себя (сообщить свое имя). Посредством аутентификации вторая сторона убеждается, что субъект действительно тот, за кого он себя выдает. В качестве синонима слова "аутентификация" иногда используют словосочетание "проверка подлинности".

Аутентификация бывает ***односторонней*** (обычно клиент доказывает свою подлинность серверу) и ***двусторонней*** (взаимной). Пример односторонней аутентификации – процедура входа пользователя в систему.

Современные средства идентификации/аутентификации должны поддерживать концепцию единого входа в сеть.

Единый вход в сеть – это, в первую очередь, требование удобства для пользователей. Если в корпоративной сети много информационных сервисов, допускающих независимое обращение, то многократная идентификация/аутентификация становится слишком обременительной.

Слайд

Идентификация – назначение метки объекту или субъекту.

Например, имя пользователя в ОС и имя почтового ящика являются идентификаторами.

Аутентификация - процедура проверки подлинности.

- Односторонняя и Взаимная (двусторонняя)
- Простая и Строгая

Методы аутентификации:

- Парольная
- По сертификатам
- Децентрализованная
- Биометрическая
- Многофакторная

ГОСТ Р ИСО/МЭК 9594-8-98 - Основы аутентификации

AAA

Authentication (аутентификация) — сопоставление персоны (запроса) существующей учётной записи в системе безопасности. Осуществляется по логину, паролю, сертификату, смарт-карте и т. д.

Authorization (авторизация, проверка полномочий, проверка уровня доступа) — сопоставление учётной записи в системе (и персоны, прошедшей аутентификацию) и определённых полномочий (или запрета на доступ). В общем случае авторизация может быть «негативной» (пользователю А запрещён доступ к серверам компании).

Accounting (учёт) — слежение за потреблением ресурсов (преимущественно сетевых) пользователем.

Идентификация пользователей в Windows.

Для защиты данных Windows использует следующие основные механизмы: аутентификация и авторизация пользователей, аудит событий в системе, шифрование данных, поддержка инфраструктуры открытых ключей, встроенные средства сетевой защиты.

Эти механизмы поддерживаются такими подсистемами Windows как LSASS (Local Security Authority Subsystem Service, подсистема локальной аутентификации), SAM (Security Account Manager, диспетчер локальных записей безопасности), SRM (Security reference Monitor, монитор состояния защиты), Active Directory (служба каталогов), EFS (Encrypting File System, шифрующая файловая система) и др.

Все действующие в системе объекты (пользователи, группы, локальные компьютеры, домены) идентифицируются в Windows не по именам, уникальность которых не всегда удается достичь, а по **идентификаторам защиты (Security Identifiers, SID)**. SID представляет собой числовое значение переменной длины:

S – R – I – S0 - S1 - … - Sn – RID

- **S** - неизменный идентификатор строкового представления SID;
- **R** – уровень ревизии (версия). На сегодня 1.
- **I** - (identifier-authority) идентификатор полномочий. Представляет собой 48-битную строку, идентифицирующую компьютер или сеть, который(ая) выдал SID объекту.

Возможные значения:

- 0 (SECURITY_NULL_SID_AUTHORITY) — используются для сравнений, когда неизвестны полномочия идентификатора;
- 1 (SECURITY_WORLD_SID_AUTHORITY) — применяются для конструирования идентификаторов SID, которые представляют всех пользователей. Например, идентификатор SID для группы *Everyone* (Все пользователи) — это *S-1-1-0*;
- 2 (SECURITY_LOCAL_SID_AUTHORITY) — используются для построения идентификаторов SID, представляющих пользователей, которые входят на локальный терминал;
- 5 (SECURITY_NT_AUTHORITY) — сама операционная система. То есть, данный идентификатор выпущен компьютером или доменом.

Sn – 32-битные коды (количеством 0 и более) субагентов, которым было передано право выдать SID. Значение первых подчиненных полномочий общеизвестно.

Они могут иметь значение:

- 5 — идентификаторы SID присваиваются сеансам регистрации для выдачи прав любому приложению, запускаемому во время определенного сеанса регистрации. У таких идентификаторов SID первые подчиненные полномочия установлены как 5 и принимают форму *S-1-5-5-x-y*
- 6 —когда процесс регистрируется как служба, он получает специальный идентификатор SID в свой маркер для обозначения данного действия. Этот идентификатор SID имеет подчиненные полномочия 6 и всегда будет *S-1-5-6*;
- 21 (SECURITY_NT_NON_UNIQUE) — обозначают идентификатор SID пользователя и идентификатор SID компьютера, которые не являются уникальными в глобальном масштабе;
- 32 (SECURITY_BUILTIN_DOMAIN_RID) — обозначают встроенные идентификаторы SID. Например, известный идентификатор SID для встроенной группы администраторов *S-1-5-32-544*;
- 80 (SECURITY_SERVICE_ID_BASE_RID) — обозначают идентификатор SID, который принадлежит службе.

Остальные подчиненные полномочия идентификатора совместно обозначают домен или компьютер, который издал идентификатор SID.

RID – 32-битный относительный идентификатор. Он является идентификатором уникального объекта безопасности в области, для которой был определен SID. Например, 500 — обозначает встроенную учетную запись Administrator, 501 — обозначает встроенную учетную запись Guest, а 502 — RID для билета на получение билетов протокола Kerberos.

При генерации SID Windows использует генератор случайных чисел, чтобы обеспечить уникальность SID для каждого пользователя. Для некоторого произвольного пользователя SID может выглядеть так:

S-1-5-21-789336058-484763869-725345543-1003

Предопределенным пользователям и группам Windows выдает характерные SID, состоящие из SID компьютера или домена и предопределенного RID. В таблице приведен перечень некоторых общеизвестных SID.

SID	Название	Описание
S-1-1-0	Все	Группа, в которую входят все пользователи
S-1-5-2	Сеть	Группа, в которую входят все пользователи, зарегистрировавшиеся в системе из сети
S-1-5-7	Анонимный вход	Группа, в которую входят все пользователи, вошедшие в систему анонимно
S-1-5-*домен*-500	Администратор	Учетная запись администратора системы. По умолчанию только эта запись обеспечивает полный контроль системы
S-1-5-*домен*-501	Гость	Учетная запись пользователя-гостя

Полный список общеизвестных SID можно посмотреть в документации Platform SDK. Узнать SID конкретного пользователя в системе, а также SID групп, в которые он включен, можно, используя консольную команду **whoami.**

```
C:\Users\user>whoami /user

USER INFORMATION
----------------

User Name SID
========= ============================================
host\user S-1-5-21-1512709811-4210725199-1529668133-1001
```

```
C:\Users\user>whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                          State
============================= ==================================== ========
SeShutdownPrivilege           Shut down the system                 Disabled
SeChangeNotifyPrivilege       Bypass traverse checking             Enabled
SeUndockPrivilege             Remove computer from docking station Disabled
SeIncreaseWorkingSetPrivilege Increase a process working set       Disabled
SeTimeZonePrivilege           Change the time zone                 Disabled
```

Соответствие имени пользователя и его SID можно отследить также в ключе реестра
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList

Процедура аутентификация в Windows.

Локальная аутентификация

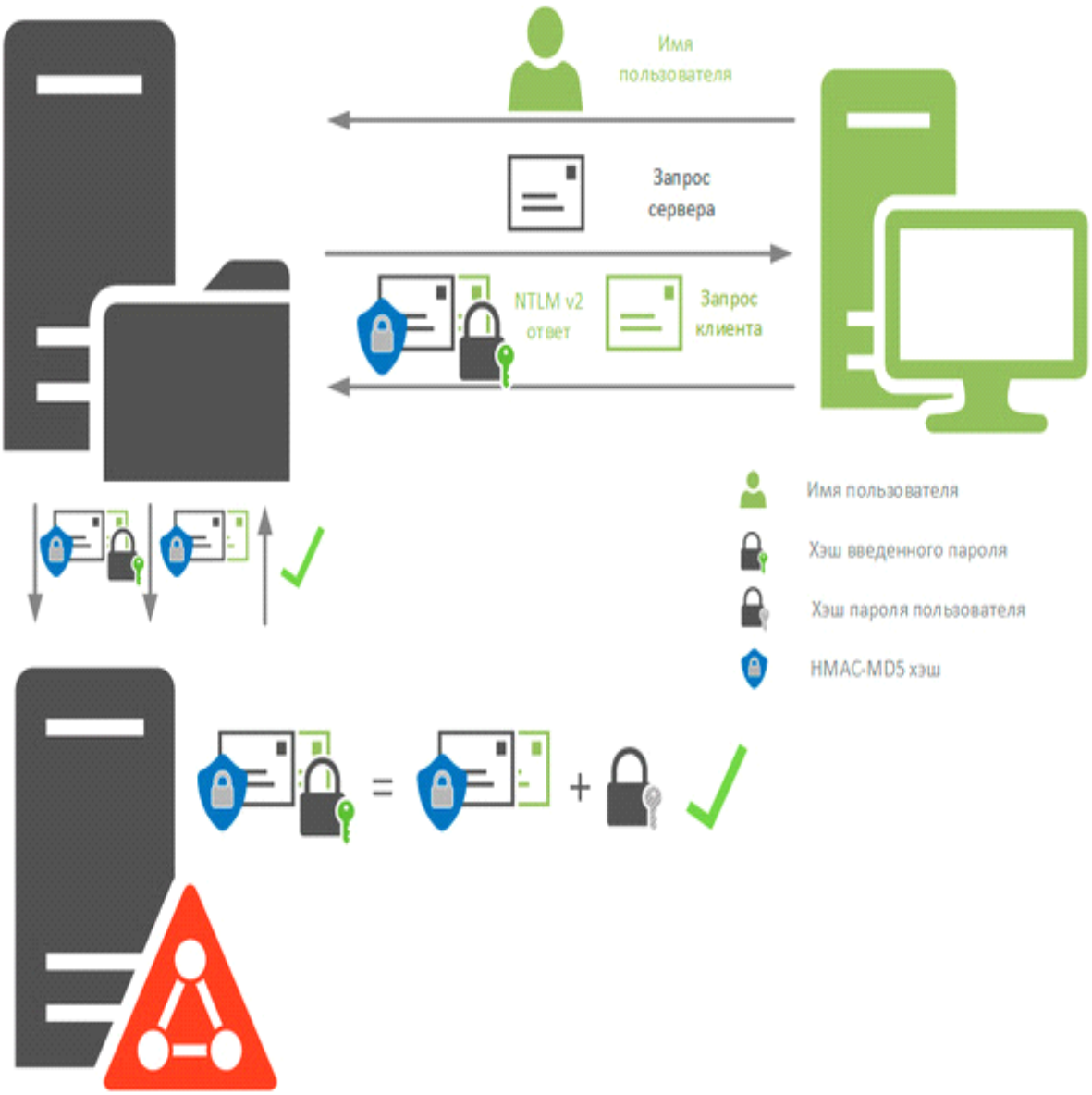
При локальной аутентификации пользователь хочет войти непосредственно на рабочую станцию, не входящую в домен. Что происходит после того, как пользователь ввел свой логин и пароль? Сразу после этого введенные данные передаются подсистеме локальной безопасности (LSA), которая сразу преобразует пароль в хэш. В открытом виде пароль нигде в системе не хранится и не фигурирует, пользователь - единственный кто его знает.

Сетевая аутентификация
NTLMv2

В случае входа пользователя в домен, для аутентификации используются иные механизмы, прежде всего протокол Kerberos, однако, если одна из сторон не может его использовать, по согласованию могут быть использованы протоколы NTLM и даже устаревший LM.

Начиная с Windows 7/Server 2008 R2 использование протоколов NTLM и LM по умолчанию выключено. Мы рассмотрим протокол NTLMv2, который вышел вместе с Windows 2000 и в настоящий момент остается актуальным.

Сразу рассмотрим схему с контроллером домена, в случае его отсутствия схема взаимодействия не меняется, только вычисления, производимые контроллером домена, выполняются непосредственно на сервере.



Клиент при обращении к серверу сообщает ему имя пользователя и имя домена, в ответ сервер передает ему случайное число - ***запрос сервера***. В ответ клиент генерирует также случайное число, куда, кроме прочего, добавляется метка времени, которое называется ***запрос клиента***. Наличие метки времени позволяет избежать ситуации, когда атакующий первоначально накапливает перехваченные данные, а потом с их помощью осуществляет атаку.

Запрос сервера объединяется с запросом клиента и от этой последовательности вычисляется HMAC-MD5 хэш.

После чего от данного хэша берется еще один HMAC-MD5 хэш, ключом в котором выступает NT-хэш пароля пользователя. Получившийся результат называется NTLMv2-ответом и вместе с запросом клиента пересылается серверу.

Сервер, получив NTLMv2-ответ и запрос клиента, объединяет последний с запросом сервера и также вычисляет HMAC-MD5 хэш, затем передает его вместе с ответом контроллеру домена. Тот извлекает из хранилища сохраненный хэш пароля пользователя и производит вычисления

над HMAC-MD5 хешем запросов сервера и клиента, сравнивая получившийся результат с переданным ему NTLMv2-ответом. В случае совпадения серверу возвращается ответ об успешной аутентификации.

Идентификация в Linux

Linux — система многопользовательская, а потому пользователь — ключевое понятие для организации всей системы доступа в Linux. Когда пользователь регистрируется в системе (проходит процедуру авторизации, например, вводя системное имя и пароль), он идентифицируется с учётной записью, в которой система хранит информацию о каждом пользователе: его системное имя и некоторые другие сведения, необходимые для работы с ним. Именно с учётными записями, а не с самими пользователями, и работает система.

Ниже приведён список этих сведений.

Системное имя (user name)

Это то имя, которое вводит пользователь в ответ на приглашение login:. Оно может содержать только латинские буквы и знак “_”. Это имя используется также в качестве имени учётной записи.

Идентификатор пользователя (UID)

Linux связывает системное имя с идентификатором пользователя в системе — UID (User ID). UID — это положительное целое число, по которому система и отслеживает пользователей. Обычно это число выбирается автоматически при регистрации учётной записи, однако оно не может быть совершенно произвольным. В Linux есть некоторые соглашения относительно того, каким типам пользователей могут быть выданы идентификаторы из того или иного диапазона. В частности, UID от “0” до “100” зарезервированы для псевдопользователей.

Идентификатор группы (GID)

Кроме идентификационного номера пользователя с учётной записью связан идентификатор группы. Группы пользователей применяются для организации доступа нескольких пользователей к некоторым ресурсам. У группы, так же, как и у пользователя, есть имя и идентификационный номер — GID (Group ID). В Linux каждый пользователь должен принадлежать как минимум к одной группе — группе по умолчанию. При создании учётной записи пользователя обычно создаётся и группа, имя которой совпадает с системным именем, именно эта группа будет использоваться как группа по умолчанию для этого пользователя.

Пользователь может входить более чем в одну группу, но в учётной записи указывается только номер группы по умолчанию. Группы позволяют регулировать доступ нескольких пользователей к различным ресурсам.

Полное имя (full name)

Помимо системного имени в учётной записи содержится и полное имя (имя и фамилия) использующего данную учётную запись человека.

Начальная оболочка (login shell)

Важнейший способ взаимодействовать с системой Linux — командная строка. Начальная оболочка (login shell) запускается при входе пользователя в систему в текстовом режиме (например, на виртуальной консоли). Поскольку в Linux доступно несколько разных командных оболочек, в учётной записи указано, какую из командных оболочек нужно запустить для данного пользователя. Если специально не указывать начальную оболочку при создании учётной записи, она будет назначена по умолчанию, вероятнее всего это будет bash.

Все перечисленные данные об учётных записях хранятся в файле /etc/passwd.

Сведения о конкретной учётной записи пользователя можно получить с помощью утилиты getent

В современных версиях Linux применяются так называемые теневые файлы паролей – shadow и gshadow. Права на них назначены таким образом, что даже чтение этих фалов без прав суперпользователя невозможно. Нужно учесть, что нормальное функционирование системы при использовании теневых файлов подразумевает одновременно и наличие файлов passwd и group.

Файл shadow хранит защищенную информацию о пользователях, а также обеспечивает механизмы устаревания паролей и учетных записей.

В Linux, кроме обычных пользователей, существует один (и только один) пользователь с неограниченными правами. Идентификаторы UID и GID такого пользователя всегда 0. Его имя, как правило, root. Для пользователя root права доступа к файлам и процессам не проверяются системой.

Команда login используется при входе в систему. Она проверяет правильность ввода имени и пароля пользователя, меняет каталог на домашний, выстраивает окружение и запускает командный интерпретатор.

Слайд

В Linux хеш имеет формат idsalt$encrypted
Пример, значений id:
1 - MD5 (md5crypt) - 1000 вызовов стандартного md5
2 и 2a - Blowfish
5 - SHA-256

6 - SHA-512 -5000 вызовов sha512

Ранее использовался алгоритм DES (Unix).

В Windows с W2k используются NTLM-хеши. Ранее использовались LM-хеши.

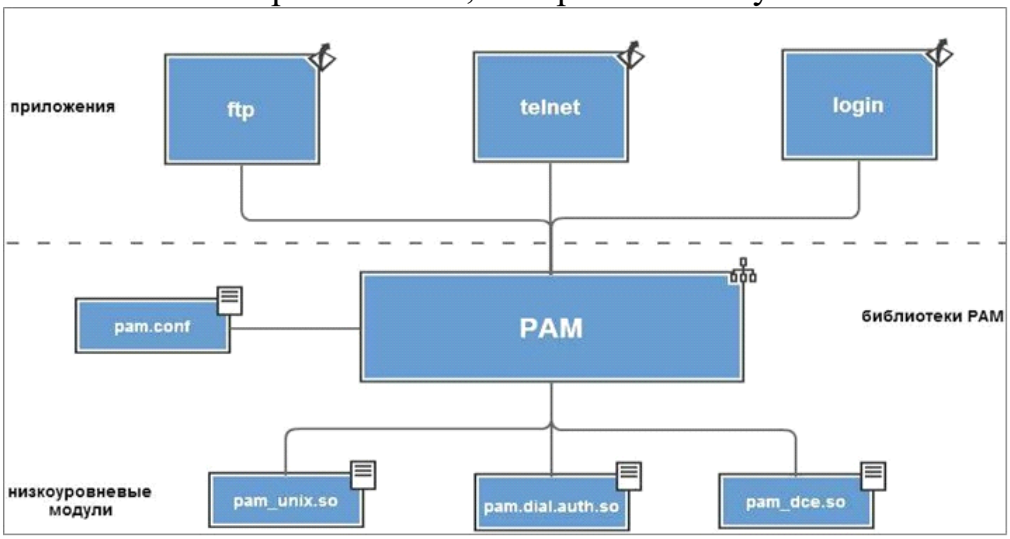
Вытащить хеш из SAM файла в Windows можно с помощью pwdump.

Пароли можно подобрать с помощью Hashcat, John the Ripper, EGB Bruteforcer, L0pthCrack, Windows Password Cracker.

Аутентификация в UNIX-подобных ОС

Подключаемые модули аутентификации (pluggable authentication modules, PAM) являются основной системой аутентификации в ОС семейства Unix, в т.ч. GNU/Linux. PAM пришел на смену встраиваемым механизмам аутентификации в различных приложениях (например, ранее в login была встроена собственная процедура аутентификации, проверяющая введенный пароль с паролем из /etc/passwd или хешем из /etc/shadow). Фактически PAM представляет из себя набор внешних модулей аутентификации, которые можно встроить в любые приложения, при этом в рамках самих приложениях нет необходимости беспокоиться об аутентификации пользователя достаточно использовать соответствующий PAM-модуль (т.е. это механизм внешней аутентификации).

Соответственно при любом изменении процедуры или последовательности идентификации и аутентификации (и/а) пропадает необходимость изменять само приложение - достаточно изменить PAM и и/а изменится во всех приложениях, которые используют этот PAM-модуль.



Модули PAM классифицируются по типу модуля. Каждый модуль должен выполнять функции хотя бы одного из четырех типов:

- Модуль аутентификации используется для аутентификации пользователей или создания и удаления учетных данных.
- Модуль управления учетными записями выполняет действия, связанные с доступом, истечением учетных данных или записей, правилами и ограничениями для паролей и т. д.
- Модуль управления сеансами используется для создания и завершения сеансов.
- Модуль управления паролями выполняет действия, связанные с изменением и обновлением пароля.

PAM обеспечивает различные функциональные возможности, такие как: аутентификация с однократной регистрацией, управление доступом и другие.

Их реализация обеспечивается различными модулями:

- pam_access обеспечивает управление входом в систему в виде протоколируемой службы при помощи имени пользователя и домена в зависимости от правил, указанных заранее в файле /etc/security/access.conf.
- pam_cracklib проверяет пароли на соответствие правилам для паролей.
- pam_env sets/unsets устанавливает и сбрасывает переменные среды из файла /etc/security/pam_env_conf.
- pam_debug выполняет отладку PAM.
- pam_deny блокирует модули PAM.
- pam_echo выводит сообщения.
- pam_exec выполняет внешнюю команду.
- pam_ftp модуль для анонимного доступа.
- pam_localuser проверяет наличие имени пользователя в файле /etc/passwd.
- pam_unix выполняет обычную аутентификацию на основе пароля из файла /etc/passwd.

Существует множество других модулей (pam_userdb, pam_warn, pam_xauth),), перехватывающих набор возвращаемых значений.

Для аутентификации по сети часто используются протоколы LDAP, SASL, Kerberos.

Протоколы аутентификации.

PAP (Password Authentication Protocol) — протокол простой проверки подлинности, предусматривающий отправку имени пользователя и пароля на сервер удаленного доступа открытым текстом (без шифрования).

Картинка

CHAP (Challenge Handshake Authentication Protocol) — протокол проверки подлинности, предусматривающий передачу не самого пароля пользователя, а косвенных сведений о нем. Например, хеша или хеша взятого от произвольных данных с паролем.

Протокол CHAP определен в RFC 1994

Картинка

MS-CHAP (Microsoft Challenge Handshake Authentication Protocol) — протокол, разработанный корпорацией Microsoft для выполнения процедур проверки подлинности удаленных рабочих станций Windows. Является модификацией CHAP.

MS-CHAP v1, RFC 2433

MS-CHAP v2, RFC 2759

EAP (*Extensible Authentication Protocol*, Расширяемый Протокол Аутентификации) — расширяемая инфраструктура аутентификации, которая определяет формат посылки и описана документом RFC 3748. Широко применяется как в проводных, так и в беспроводных сетях.

EAP использует механизм произвольной проверки подключения удаленного доступа. Точная схема проверки согласовывается клиентом удаленного доступа и устройством проверки подлинности (сервером удаленного доступа или сервером RADIUS).

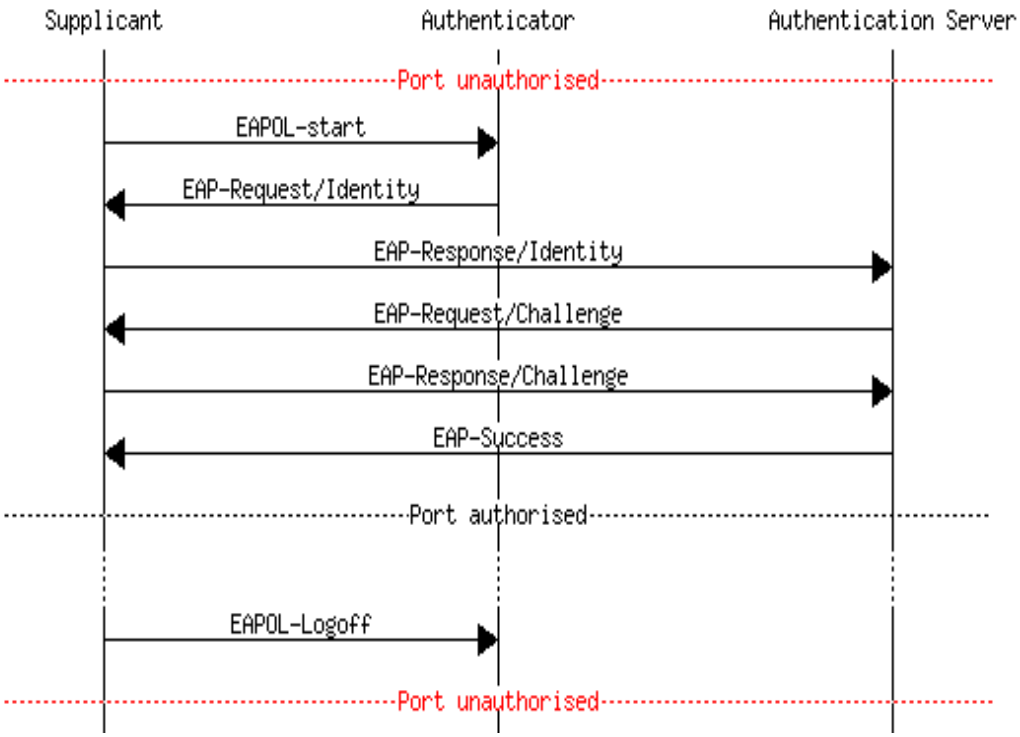
Служба маршрутизации и удаленного доступа по умолчанию поддерживает протокол EAP-TLS. Для поддержки других способов проверки EAP можно подключить другие модули EAP к серверу, на котором запущена служба маршрутизации и удаленного доступа.

Существует порядка 40 типов EAP. Для беспроводных сетей актуальны EAP-TLS, EAP-MD5, EAP-SIM, EAP-AKA, PEAP, LEAP и EAP-TTLS.

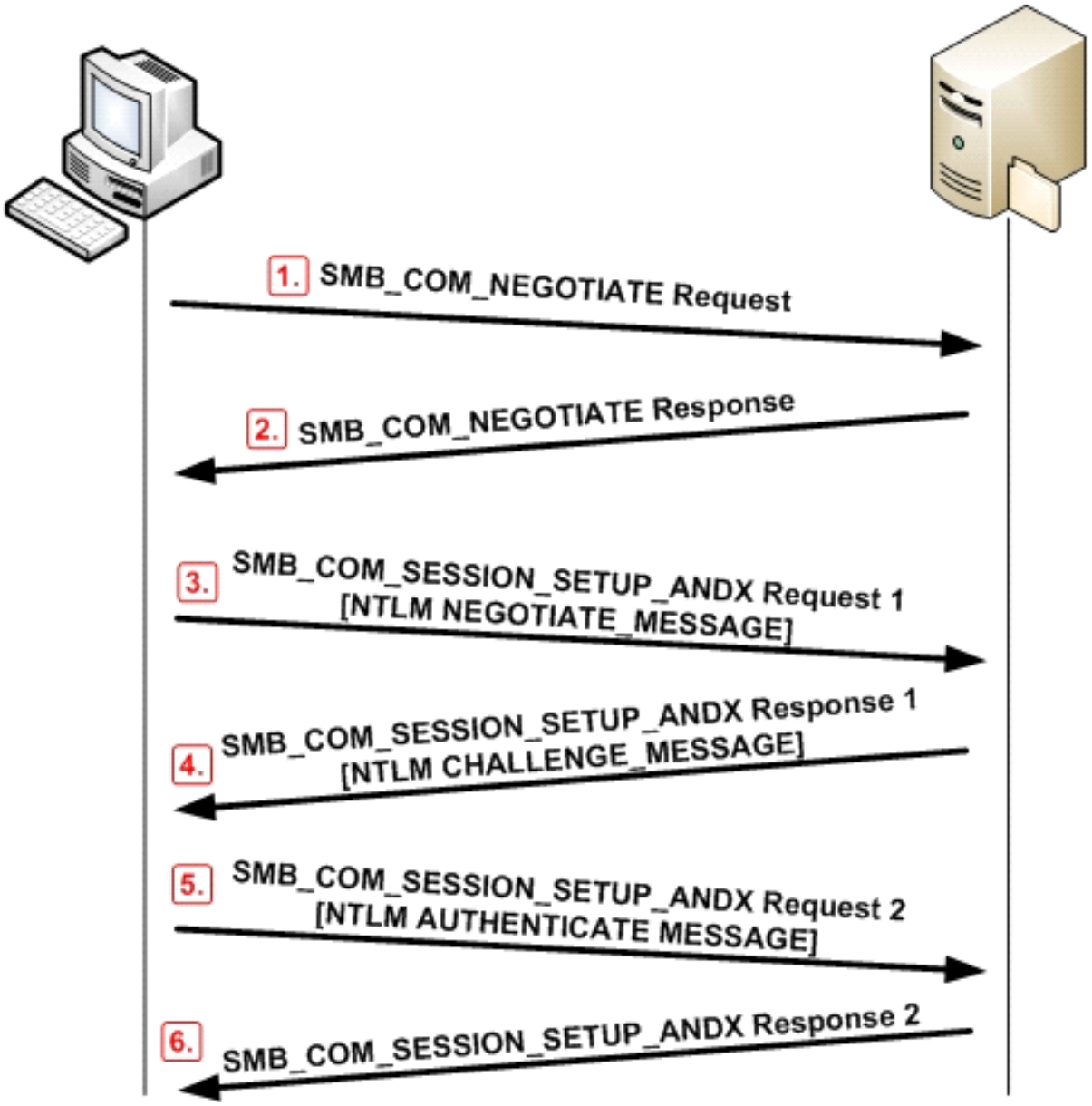
LEAP (*Lightweight Extensible Authentication Protocol*) — Облегченный расширяемый протокол аутентификации, версия протокола EAP, разработанная компанией Cisco и совместимая с продукцией семейства Cisco Aironet.

EAP определяет формат сообщений, а каждый протокол использующий EAP определяет способ инкапсуляции сообщений EAP в свой формат.

EAP-MD5



Защищенный протокол расширенной проверки подлинности PEAP (Protected Extensible Authentication Protocol) — это новый член семейства протоколов расширенной проверки подлинности (EAP). В протоколе PEAP используется протокол безопасности TLS (Transport Level Security) для создания зашифрованного канала между клиентом, подлинность которого проверяется по протоколу PEAP (например, беспроводным компьютером), и сервером проверки подлинности PEAP (например, сервером службы проверки подлинности в Интернете (IAS) или службы RADIUS (Remote Authentication Dial-In User Service). Протокол PEAP не определяет метод проверки подлинности, однако обеспечивает дополнительную безопасность для других протоколов проверки подлинности EAP, например EAP-MSCHAPv2, которые могут работать через зашифрованный канал протокола TLS, обеспечиваемый протоколом PEAP. PEAP используется как метод проверки подлинности беспроводных клиентов 802.11, но он не поддерживается для виртуальных частных сетей (VPN) или других клиентов удаленного доступа.



NTLMv2 считается достаточно надежным, хотя сейчас предпочтительный протокол — Kerberos. NTLMv2 по-прежнему широко используется для локальной регистрации и в некоторых других случаях. NTLMv2 похож на NTLM, но в хеше пароля NTLMv2 используется аутентификация сообщений HMAC-MD5, а последовательности запрос—ответ присваивается метка времени, чтобы предотвратить атаки, в ходе которых взломщик записывает учетные данные и впоследствии их использует.

В целом NTLMv2 более устойчив к атакам с применением «грубой силы», нежели NTLM, так как в протоколе применяется 128-разрядный ключ шифрования.

Аутентификация на основе внешних носителей ключа, биометрических характеристик пользователя, методы повышения защищенности ОС от подбора паролей

Ввод пароля можно подсмотреть. Иногда для подглядывания используются даже оптические приборы. Пароли нередко сообщают коллегам, чтобы те могли, например, подменить на некоторое время владельца пароля. Пароль можно угадать "методом грубой силы", используя, скажем, словарь. Если файл паролей зашифрован, но доступен для чтения, его можно скачать к себе на компьютер и попытаться подобрать пароль, запрограммировав полный перебор (предполагается, что алгоритм шифрования известен).

Тем не менее, следующие меры позволяют значительно повысить надежность парольной защиты:

- наложение технических ограничений (пароль должен быть не слишком коротким, он должен содержать буквы, цифры, знаки пунктуации и т.п.);
- управление сроком действия паролей, их периодическая смена;
- ограничение доступа к файлу паролей;
- ограничение числа неудачных попыток входа в систему (это затруднит применение "метода грубой силы");
- обучение пользователей;
- использование программных генераторов паролей (программа, основываясь на несложных правилах, порождает только благозвучные и, следовательно, запоминающиеся пароли).

Рассмотренные выше пароли можно назвать ***многоразовыми***; их раскрытие позволяет злоумышленнику действовать от имени легального пользователя. Гораздо более сильным средством, устойчивым к пассивному прослушиванию сети, являются ***одноразовые пароли***.

Наиболее известным программным генератором одноразовых паролей является ***система S/KEY*** компании Bellcore.

Идея этой системы состоит в следующем. Пусть имеется односторонняя функция f (т.е. функция, вычислить обратную которой за приемлемое время не представляется возможным). Эта функция известна и пользователю, и серверу аутентификации.

Пусть, далее, имеется секретный ключ K, известный только пользователю.

На этапе начального администрирования пользователя функция f применяется к ключу K n раз, после чего результат сохраняется на сервере. После этого процедура проверки подлинности пользователя выглядит так:

- сервер присылает на пользовательскую систему число (n – 1);
- пользователь применяет функцию f к секретному ключу K (n – 1) раз и отправляет результат по сети на сервер аутентификации;
- сервер применяет функцию f к полученному от пользователя значению и сравнивает результат с ранее сохраненной величиной.

В случае совпадения подлинность пользователя считается установленной, сервер запоминает новое значение (присланное пользователем) и уменьшает на единицу счетчик (n).

Поскольку функция f необратима, перехват пароля, равно как и получение доступа к серверу аутентификации, не позволяют узнать секретный ключ K и предсказать следующий одноразовый пароль.

Система S/KEY имеет статус Internet-стандарта (RFC 1938).

Другой подход к надежной аутентификации состоит в генерации нового пароля через небольшой промежуток времени (например, каждые 60 секунд), для чего могут использоваться программы или специальные интеллектуальные карты. Серверу аутентификации должен быть известен алгоритм генерации паролей и ассоциированные с ним параметры; кроме того, часы клиента и сервера должны быть синхронизированы.

Биометрическая аутентификация представляет собой совокупность автоматизированных методов идентификации и/или аутентификации людей на основе их физиологических и поведенческих характеристик. К числу физиологических характеристик принадлежат особенности отпечатков пальцев, сетчатки и роговицы глаз, геометрия руки и лица и т.п. К поведенческим характеристикам относятся динамика подписи (ручной), стиль работы с клавиатурой. На стыке физиологии и поведения находятся анализ особенностей голоса и распознавание речи.

В общем виде работа с биометрическими данными организована следующим образом. Сначала создается и поддерживается база данных характеристик потенциальных пользователей. Для этого биометрические характеристики пользователя снимаются, обрабатываются, и результат обработки (называемый биометрическим шаблоном) заносится в базу данных (исходные данные, такие как результат сканирования пальца или роговицы, обычно не хранятся).

В дальнейшем для идентификации (и одновременно аутентификации) пользователя процесс снятия и обработки повторяется, после чего производится поиск в базе данных шаблонов. В случае успешного поиска личность пользователя и ее подлинность считаются установленными. Для аутентификации достаточно произвести сравнение с одним биометрическим шаблоном, выбранным на основе предварительно введенных данных.

Необходимо стремиться к многофакторной аутентификации.

Отключить поддержку устаревших протоколов аутентификации (LM, NTLM).

Указать минимальную длину пароля.

Указать срок действия пароля.

Использовать стойкие алгоритмы хеширования и аутентификации.

Вывод

В данной лекции мы рассмотрели такой важный механизм безопасности как идентификация и аутентификация. Изучили конкретные реализации механизмов аутентификации в ОС Linux, Windows, FreeBSD.