Лекция 10. Средства защиты информации в ОС Windows и Unix

- Аудит событий безопасности в ОС Windows.
- Разграничение прав пользователей в ОС Unix.
- Разграничение доступа к объектам в ОС Unix.

Назначение аудита безопасности

- Определение истинных виновников компьютерных правонарушений и причин, способствовавших их возникновению.
- Обнаружение подготовительных действий к совершению компьютерного правонарушения.
- Немедленная реакция на событие, связанное с безопасностью компьютерной системы (в ОС Windows не реализовано).

Основные требования политики аудита

- Ассоциирование пользователя с любым событием аудита;
- обязательность аудита стандартного набора событий – идентификации и аутентификации, размещения объектов в адресном пространстве процессов, уничтожения объектов, действий привилегированного пользователя и др.;
- наличие необходимого набора атрибутов записи журнала аудита – даты и времени события, имени пользователя, типа события, признака успешного или неудачного завершения вызвавшего событие действия, имени связанного с событием объекта;
- возможность фильтрации записей журнала аудита;
- поддержка и защита от несанкционированного доступа к журналу аудита.

Аудит безопасности в ОС Windows

- Журнал аудита содержится в файле windows \ System32 \ Config \ secevent.evt, а доступ к нему осуществляется с помощью административной функции «Просмотр событий» Панели управления Windows.

Аудит безопасности в ОС Windows

Возможна регистрация следующих событий:

- вход пользователей в систему;
- доступ субъектов к объектам;
- доступ к службе каталогов Active Directory;
- изменение политики безопасности;
- использование привилегий;
- отслеживание процессов;
- системные события;
- попытки входа в систему;
- управление учетными записями пользователей и групп;
- доступ к глобальным системным объектам;
- использование прав на архивацию и восстановление объектов.

Аудит безопасности в ОС Windows

- Для каждой категории регистрируемых событий администратор может указать тип события (успешное и (или) неудачное завершение) либо отменить его регистрацию в журнале аудита.
- При аудите использования привилегий регистрируются попытки использования не всех возможных привилегий, а лишь тех, которые считаются потенциально опасными с точки зрения разработчиков подсистемы безопасности защищенных версий Windows (например, создание маркерного объекта или создание журналов безопасности). Следует отметить, что не все объективно опасные привилегии входят в этот список.

Аудит безопасности в ОС Windows

К системным событиям, которые могут регистрироваться в журнале аудита, относятся:

- перезагрузка операционной системы;
- завершение работы операционной системы;
- загрузка пакета аутентификации;
- запуск процесса входа (Winlogon);
- сбой при регистрации события в журнале аудита;
- очистка журнала аудита;
- загрузка пакета оповещения об изменении в списке пользователей.

Другие параметры аудита

- Максимальный размер журнала аудита.
- Реакция операционной системы на его переполнение:
  - затирать старые события при необходимости;
  - затирать старые события, которые произошли ранее установленного количества дней (в этом случае новые события не регистрируются, пока не истечет заданное количество дней с момента регистрации самого старого события, реакция по умолчанию);
  - не затирать события (очистка журнала вручную).

Аудит доступа к объектам

- Используется системный список контроля доступа SACL, содержимое которого формируется администратором системы.
- Элементы ACE списка SACL имеют один и тот же тип и содержат заголовок ACE, маску регистрируемых в журнале аудита прав доступа и SID пользователя или группы, чьи попытки доступа к объекту должны регистрироваться (если в ACE не указан SID, то регистрируются попытки доступа к объекту всех пользователей).

Администраторы и аудиторы

- Для обеспечения безопасности информации в КС целесообразно разделить полномочия администраторов КС и аудиторов (пользователей с правами доступа к файлу аудита). Если этого не сделать, то возникнет ситуация, при которой установка параметров политики безопасности и проверка ее соблюдения сосредоточатся в одних руках.

Администраторы и аудиторы

- В ОС Windows можно сделать так, что просматривать и очищать журнал аудита, а также управлять списками SACL объектов доступа смогут только члены группы аудиторов компьютерной системы.
- Но полномочия на изменение значений параметров политики аудита при этом сохранятся у членов группы администраторов компьютерной системы.

Разграничение прав пользователей в ОС Unix

- Использование ограниченной оболочки.
- Использование ограниченной файловой системы.
- Использование подключаемых модулей аутентификации (Pluggable Authentication Module, PAM).

Использование ограниченной оболочки

Для ограничения полномочий пользователя в системе Unix администратор может создать в файле паролей учетную запись с правом выполнения единственной системной команды или программы (не командного процессора), например:

date::60000:100:Запуск программы date:/tmp:/sbin/date

Использование ограниченной оболочки

Могут также применяться ограниченные оболочки (restricted shell). После запуска ограниченной оболочки выполняются команды из файла $HOME/.profile, после чего:

- пользователь не может изменить текущий каталог;
- пользователь не может изменять значение системной переменной PATH;
- пользователь не может применять команды, содержащие /;
- пользователь не может перенаправлять вывод программ, применяя операции > или >>.

Использование ограниченной оболочки

- В качестве дополнительного ограничения пользователь не может прервать работу ограниченной оболочки во время обработки ею файла $HOME/.profile – в этом случае работа командного процессора немедленно завершается.
- Использование ограниченных оболочек требует внимательного выбора разрешенных для запуска пользователями программ. Многие системные команды и прикладные программы разрешают запуск других системных команд. Например, если разрешено выполнение команды man для чтения документации, то пользователь сможет запустить текстовый редактор, затем загрузить оболочку и выполнять любые программы.

Использование ограниченной файловой системы

- Другим способом ограничения полномочий отдельных пользователей является применение программы chroot, которая изменяет представление файловой системы для вызвавшего эту программу процесса. В этом случае в сеансе работы пользователя доступна только часть файловой системы.

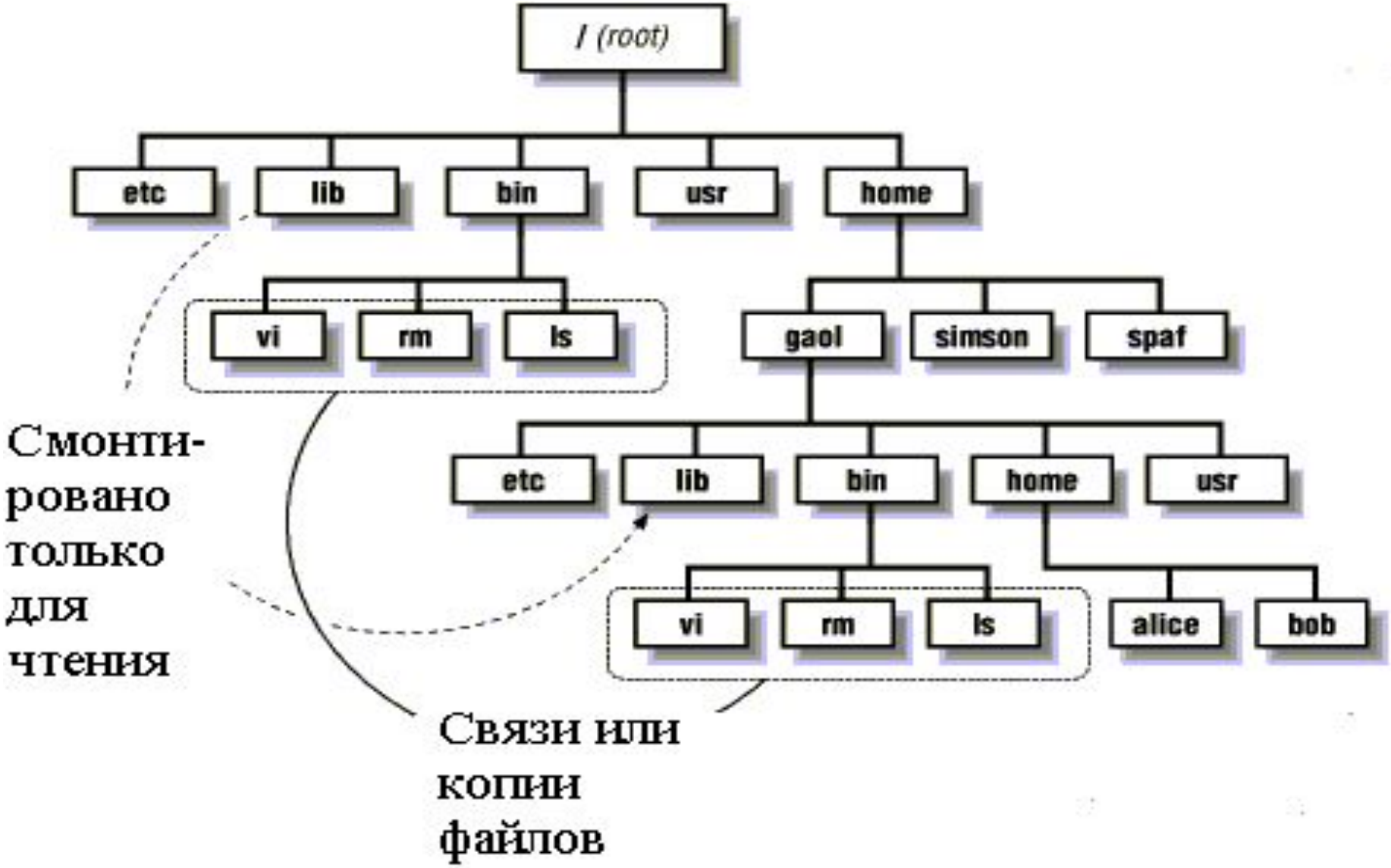
Использование ограниченной файловой системы

- Ограниченная файловая система, корнем которой будет являться, например, домашний каталог пользователя, должна иметь все необходимые файлы и команды для входа пользователя и работы его прикладных программ. Естественно, что в «новых» файлах паролей и групп не должны содержаться те же самые хеш-значения паролей, что и в «настоящих» регистрационных файлах.

Использование ограниченной файловой системы

- В каталогах /etc, /lib и /usr/lib, /bin в ограниченной файловой системе не должны содержаться те же файлы, которые содержатся в стандартных каталогах. В них должны быть копии или связи только необходимых для работы данного пользователя файлов. При этом символические связи будут ограничены доступной частью файловой системы. Можно выполнить подключение файловой системы в режиме «только чтение» (с помощью команды mount) для обхода этого ограничения.

Использование ограниченной файловой системы



Использование PAM-модулей

Существуют четыре типа модулей PAM:

- Auth – выполнение функций аутентификации;
- Account – устанавливает возможность аутентификации при выполнении некоторых условий (например, вход в компьютерную систему может быть разрешен пользователю только в будние дни и рабочие часы);
- Password – задает ограничения на пароли пользователей;
- Session – устанавливает возможность доступа пользователя к определенным сервисам после разрешения входа модулем Account.

Использование PAM-модулей

- Возможно объединение различных модулей одного типа для выполнения нескольких процедур аутентификации.
- Конфигурационные файлы PAM размещаются в каталоге / etc / pam.d.

Разграничение доступа к объектам в ОС Unix

- Разграничение доступа к объектам в операционных системах семейства Unix – файлам, каталогам, связям и специальным файлам (символьным или блочным устройствам ввода-вывода и именованным каналам) – осуществляется на основе хранящихся в индексе соответствующего объекта сведений о владельце объекта (UID) и его группе (GID), а также векторе доступа к объекту.
- Индекс файла – его управляющий блок, хранящийся в области индексов, отделенной от области файлов.

Структура каталога

Каталог – файл, состоящий из записей, соответствующих включенным в каталог файлам. Каждая запись состоит из номера индекса, связанного с данным файлом, и имени файла. Возможно создание множества файлов, связанных с одним и тем же индексом, т.е. с одной и той же областью внешней памяти.

Вектор доступа

Представляет собой список контроля доступа фиксированной (а не произвольной, как в ОС Windows) длины. Первый элемент списка определяет права доступа к объекту его владельца, второй – членов его первичной группы, а третий – всех остальных пользователей системы. Суперпользователь root имеет полный доступ ко всем объектам в системе. Каждый элемент вектора доступа имеет длину 3 или 4 бита.

Виды доступа к объекту

- Возможны три вида доступа к объекту: чтение (r), запись (w) и выполнение (x). Для каталогов запись определяет создание, переименование и (или) удаление файлов, а выполнение – поиск файла в каталоге по заданному имени.
- Пример вектора доступа к файлу:

rwxr-xr--(или 0754 в 8-й форме)

(владелец файла имеет право на полный доступ к нему, члены группы владельца – на чтение и выполнение файла, а все остальные пользователи – только на чтение файла).

Надежность разграничения доступа к объектам в Unix

Для того чтобы без использования системных команд изменить права доступа к объекту для конкретного пользователя, необходимо иметь доступ к области индексов файловой системы, которые определены в специальном файле (например, / dev / root). Но индекс этого файла также хранится в области индексов. Поэтому, если не изменять права доступа ко всем системным объектам, которые заданы по умолчанию при установке операционной системы (что может сделать только суперпользователь), то можно гарантировать безопасность работы подсистемы разграничения доступа.

Дополнительные биты в подвекторах доступа

- Если четвертый бит установлен в элементе вектора для владельца файла (SUID), то программный файл будет выполняться в сеансе любого пользователя с правами владельца этого файла. Это необходимо, например, при вызове команды passwd пользователем для изменения своего пароля. Обычный пользователь может иметь право смены своего пароля, но не может иметь право записи в файл паролей.

Дополнительные биты в подвекторах доступа

- Если четвертый бит установлен в элементе вектора доступа для членов группы владельца (SGID), то данный программный файл будет выполняться в сеансе любого пользователя с правами членов группы владельца данного файла. Если SGID установлен в векторе доступа к каталогу, то все создаваемые пользователем файлы в этом каталоге будут иметь такой же идентификатор группы владельца, как и у каталога.

Угрозы при использовании SUID

С помощью SUID или SGID возможна попытка получения постоянных полномочий администратора КС, если удастся хотя бы один раз получить (перехватить) его пароль. Например:

- нарушитель создает копию командного процессора в своем домашнем каталоге или еще где-нибудь;
- назначает владельцем созданного файла администратора и с применением его полномочий (после входа в систему под его учетной записью) устанавливает SUID (под своей учетной записью он это сделать не может, т.к. при смене владельца все дополнительные биты сбрасываются);
- до тех пор, пока созданный файл не будет уничтожен, нарушитель будет пользоваться правами администратора.

Защита от угрозы

Для предотвращения приведенной выше угрозы необходимо регулярно проверять файловую систему на наличие незарегистрированных файлов с установленными битами SUID или SGID.

Дополнительные биты в подвекторах доступа

- Если четвертый бит установлен в элементе вектора доступа для всех остальных пользователей (Sticky), то операционная система создает специальный текстовый образ программного файла. Чаще этот бит используется для каталогов и определяет запрет на удаление или переименование файлов других пользователей в этом каталоге. Это особенно важно для каталогов /tmp и /usr/tmp, чтобы одни пользователи не могли повредить работе других. Бит Sticky для каталогов может быть установлен только администратором.

Права доступа к вновь создаваемым объектам

Определяются на основе значения системной переменной umask, которое устанавливается в файлах пользователей .login, .cshrc или .profile либо в системном файле / etc / profile. Значение umask определяет сбрасываемые биты в элементах вектора доступа к создаваемому объекту.

Права доступа к вновь создаваемым объектам

Например, чтобы у всех вновь создаваемых объектов вектор доступа был равен 0755 (владелец объекта имел бы полный доступ к нему, а члены группы владельца и все остальные пользователи – право на чтение и выполнение объекта), значением переменной umask должно быть 0022. Для того чтобы только владелец создаваемого объекта имел к нему полный доступ, а всем остальным пользователям доступ к объекту был запрещен, следует установить umask в значение 0077.