Особенности безопасности распределенных вычислительных систем

Кафедра ИБ БГАРФ

Зензин Александр Степанович, к.т.н.

Обзор

1. Введение
2. Распределенная обработка данных.
3. Классификация сетей по способам распределения данных.
4. Особенности безопасности распределенных вычислительных систем.
5. Понятие защищенного канала.

Введение

Сети ЭВМ из достояния научных центров постепенно стали обязательным атрибутом процветающих торговых фирм, банков, милиции, таможни, налоговой службы и т.д. Если раньше главной проблемой было создание сети и обеспечение доступа к Интернет, то сегодня по мере увеличения размеров сети проблема безопасности выходит на лидирующие позиции. **Безопасность** - комплексное понятие, это и ограничение нежелательного доступа, и сохранность информации, и живучесть самой сети. Актуальность проблемы подтверждает количество RFC-документов, опубликованных за последнее время по данной тематике (см. ниже), а также включение этой проблемы в стандарты ИБ систем электросвязи страны. Данная дисциплина рассматривает вопросы безопасности, которые касаются обычных сетей, не содержащих конфиденциальную или тем более секретную информацию.

Уязвимость сетевых объектов в заметной степени определяется монопольным статусом основных видов программного обеспечения. Windows и UNIX занимают более 90% рынка ОС, Apache и IIS охватывают более 90% рынка программ WEB-сервиса. Когда на каждый десяток разработчиков программного обеспечения приходится миллион тесно сотрудничающих хакеров, сокращение числа объектов атаки (ОС и основные приложения) создает благоприятные условия для уязвимости потенциальных жертв. Единственным спасением является расширение многообразия ОС и приложений.

Существуют юридические аспекты сетевой безопасности, организационные и программно-технические. Проблема усугубляется тем, что законодательство различных стран, связанное с обеспечением конфиденциальности и безопасности, отличается значительно, а Интернет по своей природе носит всемирный характер. Следует помнить, что практически любые меры безопасности ограничивают возможности и свободу клиентов сети.

В данной дисциплине будут рассмотрены организационные и программно-технические подходы обеспечения безопасности сетей ЭВМ.

Введение

Рассмотрим сначала факторы, влияющие на надежность сети. Источниками ненадежности сети могут быть:

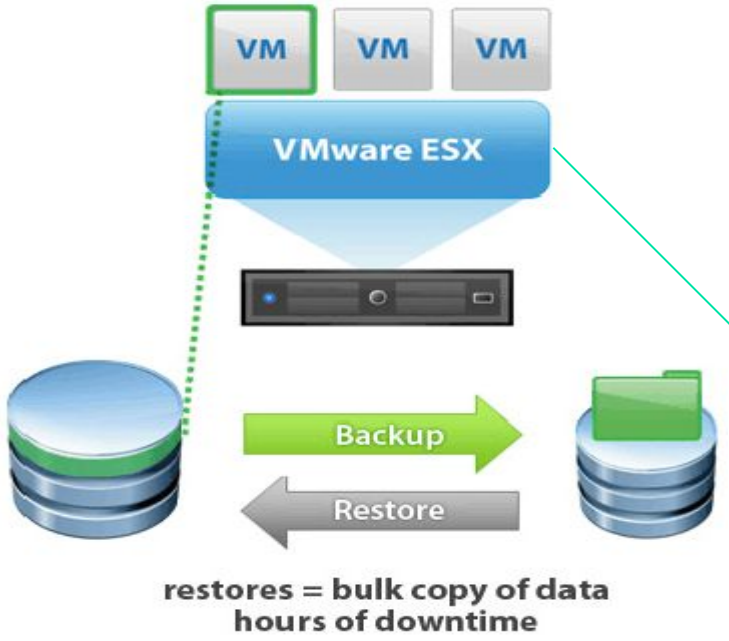
- стихийные явления, к которым можно отнести отказы оборудования или питания, а также некомпетентность обслуживающего персонала;
- несанкционированные действия операторов удаленных ЭВМ.

Основу стабильности сети составляют надежность ЭВМ и сетевого оборудования, а также устойчивость каналов связи. Каналы связи – это ответственность телекоммуникационных компаний.

В нашей власти правильная конфигурация узла, разумное распределение ответственности и качество сетевого питания (стабильность напряжения и частоты, амплитуда помех). Для решения последней проблемы используют специальные фильтры, мотор-генераторы и **UPS** (Uninterruptable Power Supply). Выбор того или иного решения зависит от конкретных условий, но для серверов использование UPS крайне желательно. При выборе UPS нужно учесть суммарную потребляемую мощность оборудования, подключаемого к источнику питания, и время, в течение которого UPS способен работать без напряжения в сети. При этом главная задача UPS - обеспечение завершения операций обмена с диском до того, как произойдет полное обесточивание сервера, или когда будет произведено переключение на резервный канал питания. Это осуществимо при использовании специального интерфейса и соответствующего программного обеспечения, работающего согласно протоколу SNMP. Указанный интерфейс обеспечит блокировку начала новых операций обмена или выполнит shutdown, если напряжение в сети упало ниже допустимого уровня. Сетевые фильтры являются желательными при работе с любыми ЭВМ, так как сеть в России сильно засорена высокочастотными помехами.

Введение

Поскольку абсолютная надежность недостижима, одним из средств сохранения информации является дублирование носителей (напр. дисков), копирование и сохранение копий в надежном месте. Если раньше для этой цели годились гибкие диски или магнитные ленты, сегодня их пригодность может быть подвергнута сомнению. Конечно, ленты типа exabyte емкостью 2.5-20 Гбайт достаточно широко использовались, но относительно высокая стоимость таких накопителей ограничивает их применимость (да и скорость записи для них не слишком высока). Альтернативой им могут стать накопители с перезаписываемыми CD, где стоимость устройства несколько ниже, за то емкость одного диска для дешевых моделей пока не превосходит 1 Гбайт. Не исключено, что в скором времени основным средством сохранения информации станет ее дублирование на независимом жестком диске. Это может произойти при широком внедрении компактных жестких дисков емкостью порядка 10 Тбайт и более (а ведь еще несколько лет назад здесь стояла цифра 10Гбайт!).



Virtualization platforms, such VMware ESX and Microsoft Hyper-V, represent the first form of virtualization that is typically...

Рис. 1. Традиционная схема резервного копирования

Введение

Пожалуй самым ненадежном элементом персонального компьютеря является жесткий диск. Стоимость дисков падает и появилась возможность использования систем **RAID** (Redundant Array of Intelligent/Independent Drives). Такие системы автоматически дублируют все хранящиеся данные и мониторируют локальные сбои, что делает систему более устойчивой.

Отдельную проблему может составлять ***катострофоустойчивость*** информационной системы. Здесь имеется в виду сохранность данных в случае стихийных бедствий (пожаров, землятресений, наводнений или прорывов водопроводной трубы на вышерасположенном этаже). Если вся критическая информация находится на каких-то носителях в пределах одной комнаты или даже здания, при катастрофах указанного типа она может быть утрачена. Интернет предоставляет решение этой проблемы. Ведь можно организовать резервное копирование критически важных файлов на носители в удаленном здании или даже в другом городе. Понятно, что при резервном копировании через Интернет данные должны быть криптографически защищаться. Криптографическая защита делает возможным бизнес предоставления услуг резервного копирования (**SaaS**-технология - Software as a Service).

В этой схеме резервное копирование производится удаленно, а все файлы шифруются с помощью ключа владельца, так что их несанкционированное прочтение не возможно. Этот метод формирования backup становится новым направлением ИТ-бизнеса, который позволяет более экономно использовать ресурсы как клиента, так и фирмы, предоставляющей услуги. В настоящее время эта технология стала частью облачного компьютинга.

Введение

Во многих WEB-приложениях возникает необходимость хранения критически важных данных (паролей, номеров кредитных карт, номеров банковских счетов, персональной информации. Хранится такая информация в базах данных или просто в файлах. Для защиты таких данных обычно используется криптография. То что данные зашифрованы, делает часто их хозяев беззаботными, а напрасно, так как при этом сохраняется достаточно много уязвимостей, сопряженных с ошибками пользователя:

- сбой, который привел к тому, что данные оказались записаны незашифрованными;
- небезопасное хранение криптоключей, сертификатов и паролей;
- неправильное хранение секретов в памяти;
- плохой выбор алгоритма;
- плохой генератор псевдостучайных чисел;
- попытки изобрести новых алгоритм криптозащиты;
- некорректные процедуры смены ключей.

Решение многих из названных выше проблем можно реализовать достаточно простыми мерами:

- минимизировать применение криптографических методов. Хранить на дисках только ту информацию, которая абсолютно необходима. Например, вместо того, чтобы хранить номера кредитных карт в зашифрованном виде (избавляет от ввода при повторении сессии), предлагать пользователям вводить их каждый раз. Это же относится и к паролям;
- если использование криптографии неизбежно, применяйте библиотеку, которая широко опробована и не имеет известных уязвимостей. Ключ должен хранится в двух местах и извлекается оттуда непосредственно при исполнении программы.

Введение

Существует и другой аспект информационной безопасности. Как защитить персональные данные от разглашения людьми, которые с ними работают? Известно, что базы данных об автомобилях и их владельцах, мобильных телефонах, о прописках, банковских проводках и т.д. стали предметом широкой торговли. Здесь предстоит решить проблемы не только технологического порядка. Проблема эта, вероятно только не в таком масштабе, знакома и в других странах (Essential Guide to Identity & Access Management).

Возможным решение проблем сохранности критических данных (помимо криптозащиты) может стать мониторинг и журналирование операций с файлами, содержащими такую информацию. Кроме того, можно реализовать программы фильтрующие конфиденциальные данные. Например, номера кридитных карт имеют известный формат и обычно снабжены контрольной суммой. Т.о. легко можно выявлять и блокировать попытки передачи по сети номеров кредитных карт.

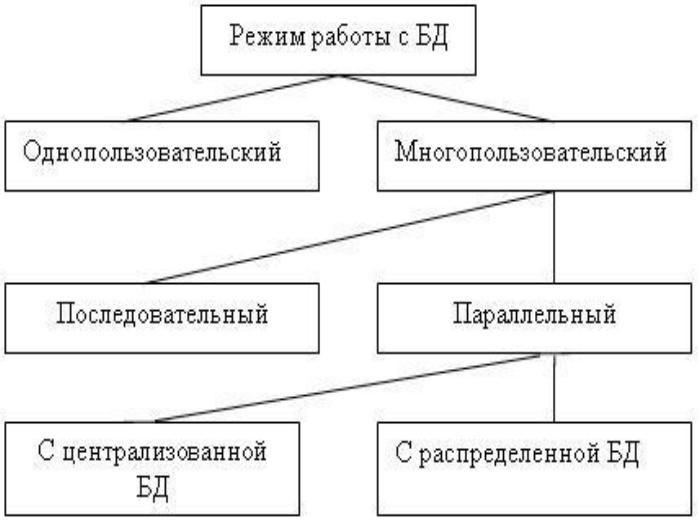
Широкое внедрение SaaS ставит ряд новых проблем. Среди них сопоставимость средств и политик защиты, а также оборудования и программ противодействия вторжениям и DoS-атакам. Важным пунктом является и согласование методов аутентификации и алгоритмов проверки целостности и неискаженности пересылаемых файлов. Большинство провайдеров предлагают 128-битную криптозащиту, здесь требуется выработка единого стандарта.

К сожалению, помимо объективных причин на надежность и устойчивость работы сети влияет и субъективный фактор. Это, прежде всего некомпетентный персонал, различные компьютерные вирусы и хакеры. Скрытая безработица среди программистов порождает хакерство в России. Практическое отсутствие сетевых вирусов связано с ограниченностью сетей в России и со сложностью их написания, но этот барьер будет скоро преодолен и к этому следует готовиться уже сегодня.

Распределенная обработка данных

Параллельный доступ к одной БД нескольких пользователей, в том случае если БД расположена на одной машине, соответствует режиму распределенного доступа к централизованной БД. (Такие системы называются *системами распределенной обработки данных)*.

Если же БД распределена по нескольким компьютерам, расположенным в сети, и к ней возможен параллельный доступ нескольких пользователей, то мы имеем дело с параллельным доступом к распределенной БД. Подобные системы называются *системами распределенных баз данных.* В общем случае режимы использования БД можно представить в виде, изображенным на рисунке.



Системы распределенной обработки данных в основном связаны с первым поколением БД, которые строились на мультипрограммных операционных системах и использовали централизованное хранение БД на устройствах внешней памяти центральной ЭВМ и терминальный многопользовательский режим доступа к ней. При этом пользовательские терминалы не имели собственных ресурсов — то есть процессоров и памяти, которые могли бы использоваться для хранения и обработки данных. Первой полностью реляционной системой, работающей в многопользовательском режиме, была СУБД SYSTEM R, разработанная фирмой IBM, именно в ней были реализованы как язык манипулирования данными SQL, так и основные принципы синхронизации, применяемые при распределенной обработке данных, которые до сих пор являются базисными практически во всех коммерческих СУБД.

Распределенная обработка данных

Общая тенденция движения от отдельных mainframe-систем к открытым распределенным системам, объединяющим компьютеры среднего класса, получила название DownSizing. Этот процесс оказал огромное влияние на развитие архитектур СУБД и поставил перед их разработчиками ряд сложных задач. Главная проблема состояла в технологической сложности перехода от централизованного управления данными на одном компьютере и СУБД, использовавшей собственные модели, форматы представления данных и языки доступа к данным и т. д., к *распределенной обработке данных в неоднородной вычислительной среде*, состоящей из соединенных в глобальную сеть компьютеров различных моделей и производителей.

В то же время происходил встречный процесс — UpSizing. Бурное развитие персональных компьютеров, появление локальных сетей также оказали серьезное влияние на эволюцию СУБД. Высокие темпы роста производительности и функциональных возможностей PC привлекли внимание разработчиков профессиональных СУБД, что привело к их активному распространению на платформе настольных систем.

Сегодня возобладала тенденция создания информационных систем на такой платформе, которая точно соответствовала бы ее масштабам и задачам. Она получила название RightSizing (помещение ровно в тот размер, который необходим).

Однако и в настоящее время большие ЭВМ сохраняются и сосуществуют с современными ***открытыми системами***. Причина этого проста — в свое время в аппаратное и программное обеспечение больших ЭВМ были вложены огромные средства: в результате многие продолжают их использовать, несмотря на морально устаревшую архитектуру. В то же время перенос данных и программ с больших ЭВМ на компьютеры нового поколения сам по себе представляет сложную техническую проблему и требует значительных затрат.

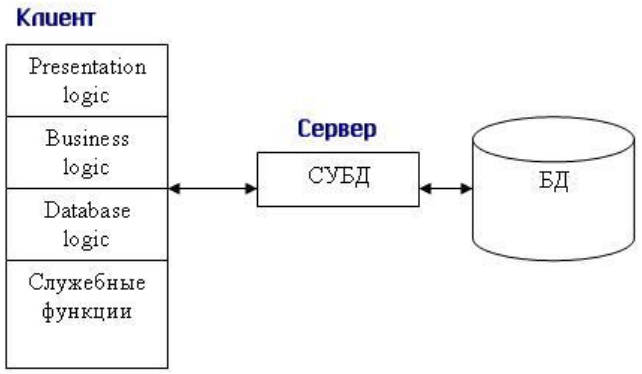
Классификация сетей по способам распределения данных

Вычислительная модель **«клиент—сервер»** исходно связана с парадигмой открытых систем, которая появилась в 90-х годах и быстро эволюционировала. Сам термин «клиент-сервер» исходно применялся к архитектуре программного обеспечения, которое описывало распределение процесса выполнения по принципу взаимодействия двух программных процессов, один из которых в этой модели назывался «клиентом», а другой — «сервером». Клиентский процесс запрашивал некоторые услуги, а серверный процесс обеспечивал их выполнение. При этом предполагалось, что один серверный процесс может обслужить множество клиентских процессов.

Основной принцип технологии «клиент—сервер» применительно к *технологии баз данных* заключается в разделении функций стандартного интерактивного приложения на 5 групп, имеющих различную природу:

- функции ввода и отображения данных (Presentation Logic);
- прикладные функции, определяющие основные алгоритмы решения задач приложения (Business Logic);
- функции обработки данных внутри приложения (Database Logic),
- функции управления информационными ресурсами (Database Manager System);
- служебные функции, играющие роль связок между функциями первых четырех групп.

Структура типового приложения, работающего с базой данных, приведена на рисунке.



Классификация сетей по способам распределения данных

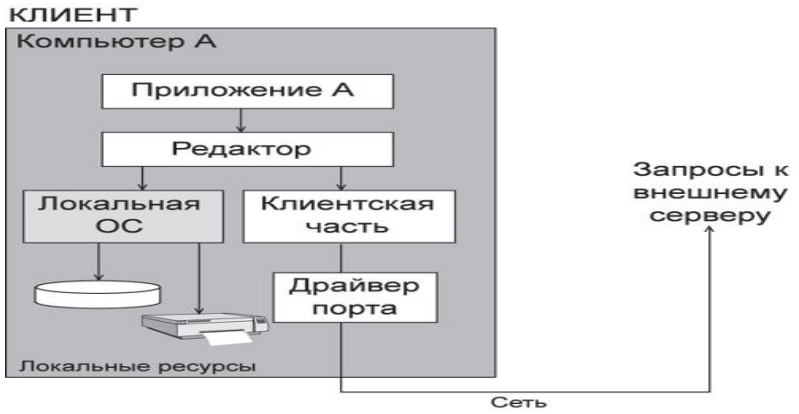
Как отмечалось выше *распределенная обработка данных в неоднородной вычислительной среде*, требует **соединения в вычислительную сеть** компьютеров различных моделей.

Вычислительная сеть — это многослойный комплекс взаимосвязанных и согласованно функционирующих программных и аппаратных компонентов: *компьютеров, коммуникационного оборудования, операционных систем, сетевых приложений*.

В зависимости от того, как распределены функции между компьютерами сети, они могут выступать в трех разных ролях:

1. Компьютер, занимающийся исключительно обслуживанием запросов других компьютеров, играет роль выделенного сервера сети.

2. Компьютер, обращающийся с запросами к ресурсам другой машины, играет роль узла-клиента.



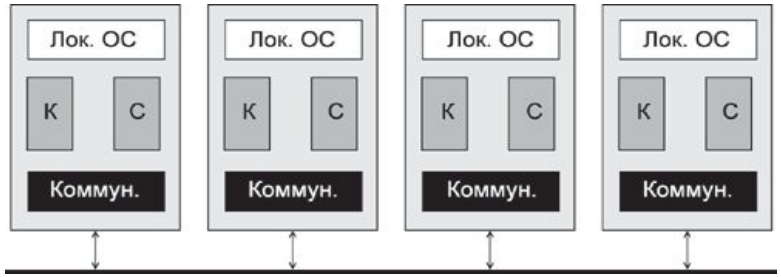
Классификация сетей по способам распределения данных

3. Компьютер, совмещающий функции клиента и сервера, является одноранговым узлом.

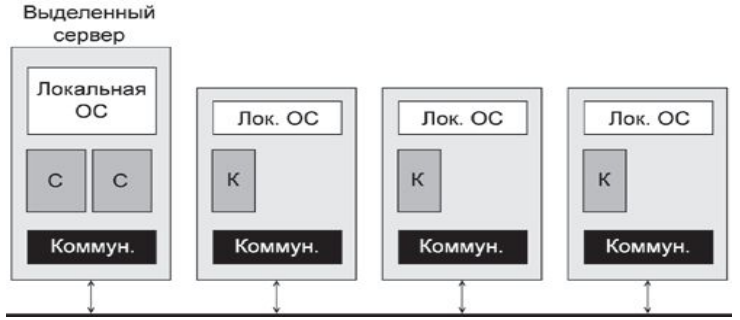
Очевидно, что сеть не может состоять только из клиентских или только из серверных узлов. Сеть может быть построена по одной из трех схем:

- сеть на основе **одноранговых узлов** — **одноранговая сеть**;
- сеть на основе клиентов и серверов — **сеть с выделенными серверами**;
- сеть, включающая узлы всех типов — **гибридная сеть**.

Каждая из этих схем имеет свои достоинства и недостатки, определяющие их области применения.



а) одноранговая сеть



б) сеть с выделенным сервером

Классификация сетей по способам распределения данных

Существует несколько типов систем, различающихся по характеру распределения данных и их использованию:

1. Системы с *централизованными* данными. При наличии нескольких клиентов они могут либо находиться в том же месте, где размещены и данные, либо быть удалены от них.
2. *Иерархические* системы. В схеме иерархии **зависимых данных** данные в машинах нижнего уровня тесно связаны с данными в машине верхнего уровня. Зачастую они могут быть подмножествами данных верхнего уровня, используемыми в локальных приложениях. Эталонная копия данных при этом может храниться на верхнем уровне. При внесении изменений в данные на нижнем уровне эти изменения *должны передаваться* в машину верхнего уровня - иногда немедленно, иногда позднее, в цикле обновления. В других системах такого типа нижний уровень может содержать те же данные, что и верхний, и еще свои собственные, которые никогда не передаются наверх.

В схеме иерархии **независимых данных** все процессоры представляют собой независимые замкнутые системы обработки данных. Структура данных на машинах нижнего уровня сильно отличается от их структуры на верхнем уровне. Наиболее распространенным примером отношений такого вида могут служить системы, в которых нижние уровни предназначены для рутинных повторяющихся (массовых) операций: приема заказов, контроля за выпуском продукции, управления складом и т. п.

В машине верхнего уровня, находится информационная система, которая должна снабжать необходимой информацией руководство, планирующие подразделения, отделы прогнозирования, разработчиков новых изделий и стратегий. Все данные могут быть извлечены из нижних уровней, но они суммируются, редактируются, реорганизуются с помощью вторичных индексов или иных методов поиска, чтобы обеспечить ответы на разнообразные, часто заранее непредвиденные вопросы.

Классификация сетей по способам распределения данных

3. Системы с *расщепленными* данными. Это несколько систем с идентичными структурами данных. Система в районе А хранит данные района А, система в районе В хранит данные района В и т. д. Большинству обрабатываемых транзакций требуются только те данные, которые находятся в обрабатывающей системе, но в некоторых случаях для обработки транзакции, возникшей в одном районе, могут потребоваться данные из другого района. При этом объектом передачи из одного района в другой через сеть может стать *либо транзакция, либо данные*. Во многих организациях установлено большое число персональных компьютеров с одинаковыми расщепленными файлами в каждой, а сеть объединяет их в единую систему. В системах с расщепленными данными прикладные программы и структуры данных одни и те же. Программирование для всех машин выполняет одна общая группа разработчиков (В системах же с разделенными данными объединенные в сеть подсистемы содержат разные данные и разные программы, как правило, создаются разными группами разработчиков).
4. Системы с *реплицированными* данными. Идентичные копии данных хранятся в разных местах, потому что дублирование памяти позволяет избежать передачи больших объемов данных, и это оказывается дешевле. Такая организация имеет смысл только в тех случаях, когда объем обновлений невелик.
5. *Гетерогенная* система. Она состоит из независимых вычислительных систем, установленных различными организациями для решения своих специфических задач и объединенных через универсальную сеть. Каждый компьютер хранит только собственные данные, и никакого сходства или единства форм организации данных здесь нет. Пользователь может получить доступ к любой машине в сети, но он должен в деталях знать, как организованы данные на этой конкретной машине.

Особенности безопасности распределенных вычислительных систем

Все эти типы систем, классифицируются как по характеру распределения данных, так и по способам поддержки этих способов распределения схемами организации вычислительных сетей: одноранговыми, с выделенными серверами и гибридными. Например, может быть организована одноранговая система с расщепленными данными.

Особенности безопасности распределенных вычислительных систем.

Чем сложнее сеть, тем острее встают вопросы ***управления сетью*.** Современные программные средства управления ЛВС в большинстве своем состоят из различных *утилит*, из которых компонуются комплексы (функции) управления, в том числе функции безопасности, такие как **защиты от несанкционированного доступа**, которые обычно обеспечивают:

- ограничение доступа в определенное время, и (или) для определенных станций, и (или) определенное число раз;
- ограничение совокупности доступных конкретному пользователю каталогов;
- ограничение для конкретного пользователя списка возможных действий;
- ограничение доступа к конкретным файлам.

Отказоустойчивость определяется наличием в сети автономного источника питания, отображением или дублированием информации в дисковых накопителях. Отображение заключается в хранении двух копий данных на двух дисках, подключенных к одному контроллеру, а дублирование означает подключение каждого из этих двух дисков к разным контроллерам. Сетевая ОС, реализующая дублирование дисков, обеспечивает более высокий уровень отказоустойчивости. Дальнейшее повышение отказоустойчивости связано с дублированием серверов.

Особенности безопасности распределенных вычислительных систем

Безопасность вещь относительная. Для того чтобы конкретизировать обсуждаемый предмет, введены четыре уровня безопасности *безопасности операционных систем*, обозначенные латинскими буквами A-D. Каждый из уровней поделен на субуровни, обозначаемые соответствующей буквой и цифрой (например, A1, A2, A3 или D1, D2 и т.д.). Уровню D1 соответствует DOS, где пользователь имеет доступ ко всем системным ресурсам и программам. Владельцем файлов является текущий пользователь. C-уровень предлагает большую безопасность, чем D-уровень. На C-уровне пользователь должен быть идентифицирован, прежде чем он получит доступ к своим файлам. Стандартная система UNIX с канонической схемой доступа (login/password) относится к уровню безопасности C1. Уровень C2 включает возможность блокировать для пользователя исполнение некоторых команд, если не выполняются определенные условия. При этом пользователь не может также контролировать определенные операции, которые имеют место. Многие современные UNIX-системы, в частности SCO-UNIX, могут обеспечивать безопасность на уровне C2. B-уровень дает еще большие гарантии безопасности, запрещая пользователю изменять условия доступа для файлов. Очень не многие операционные системы и практически ни одна коммерчески доступная система не обеспечивают такого уровня безопасности. Но при выборе уровня безопасности следует учитывать, что повышение надежности неизбежно уменьшает удобство пользования системой, так что обычно приходится идти на определенный компромисс.

Для коммерческих предприятий безопасность является экономической категорией (см. Risk Management: Bridging Policies and Procedures - Fundamental Security Concepts или Understanding the Basic Configuration of the Adaptive Security Appliance (ASA)) и нужно научится оценивать эффективность средств защиты. Чрезмерные издержки здесь могут быть столь же плохи, как и недостаточные вложения.

Изучение этих вопросов и есть цель данной дисциплины.

Понятие защищенного канала

Известно, что задачу защиты данных можно разделить на две подзадачи: защиту данных внутри компьютера и защиту данных в процессе их передачи от одного компьютера в другой. Для обеспечения безопасности данных при их передаче по публичным сетям используются различные *технологии защищенного канала*.

Технология защищенного канала обеспечивает защиту трафика между двумя точками в открытой транспортной сети, например в Интернете. Защищенный канал подразумевает выполнение трех основных функций:

- взаимная аутентификация абонентов при установлении соединения, которая может быть выполнена, например, путем обмена паролями;
- защита передаваемых по каналу сообщений от несанкционированного доступа, например, путем шифрования;
- подтверждение целостности поступающих по каналу сообщений, например, путем передачи одновременно с сообщением его дайджеста.

В зависимости от местарасположения программного обеспечения защищенного канала различает две схемы его образования:

- схема с конечными узлами, взаимодействующими через публичную сеть (рис. 2, а);
- схема с оборудованием поставщика услуг публичной сети, расположенным на границе между частной и публичной сетями (рис. 2, б).

Понятие защищенного канала

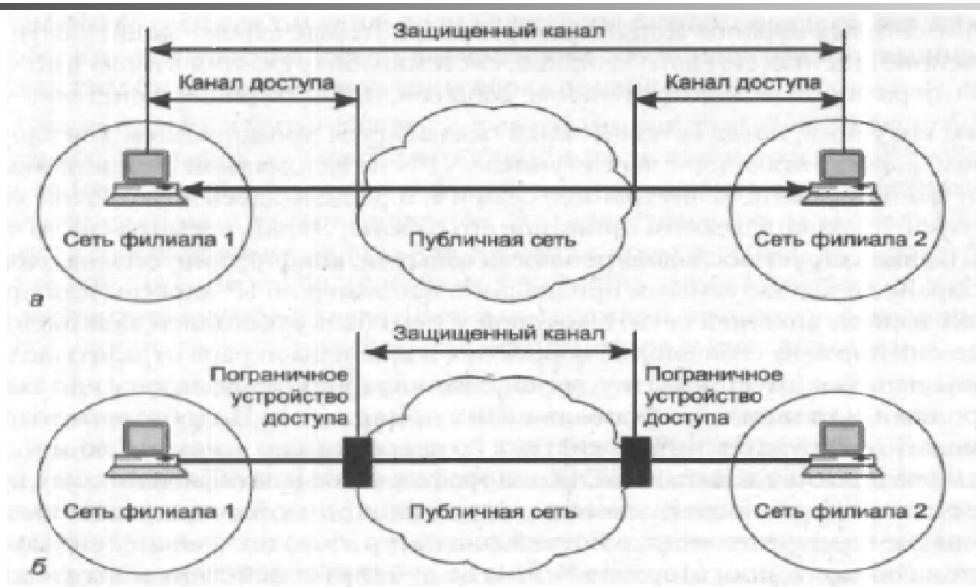


Рис. 2. Два подхода к образованию защищенного канала

В первом случае защищенный канал образуется программными средствами, установленными на двух удаленных компьютерах, принадлежащих двум разным локальным сетям одного предприятия и связанных между собой через публичную сеть. Преимуществом этого подхода является полная защищенность канала вдоль всего пути следования, а также возможность использования любых протоколов создания защищенных каналов, лишь бы на конечных точках канала поддерживался один и тот же протокол. Недостатки заключаются в *избыточности* и децентрализованности решения. Избыточность состоит в том, что вряд ли стоит создавать защищенный канал на всем пути следования данных: уязвимыми для злоумышленников обычно являются сети с коммутацией пакетов, а не каналы телефонной сети или выделенные каналы, через которые локальные сети подключены к территориальной сети. Поэтому защиту каналов доступа к публичной сети можно считать избыточной.

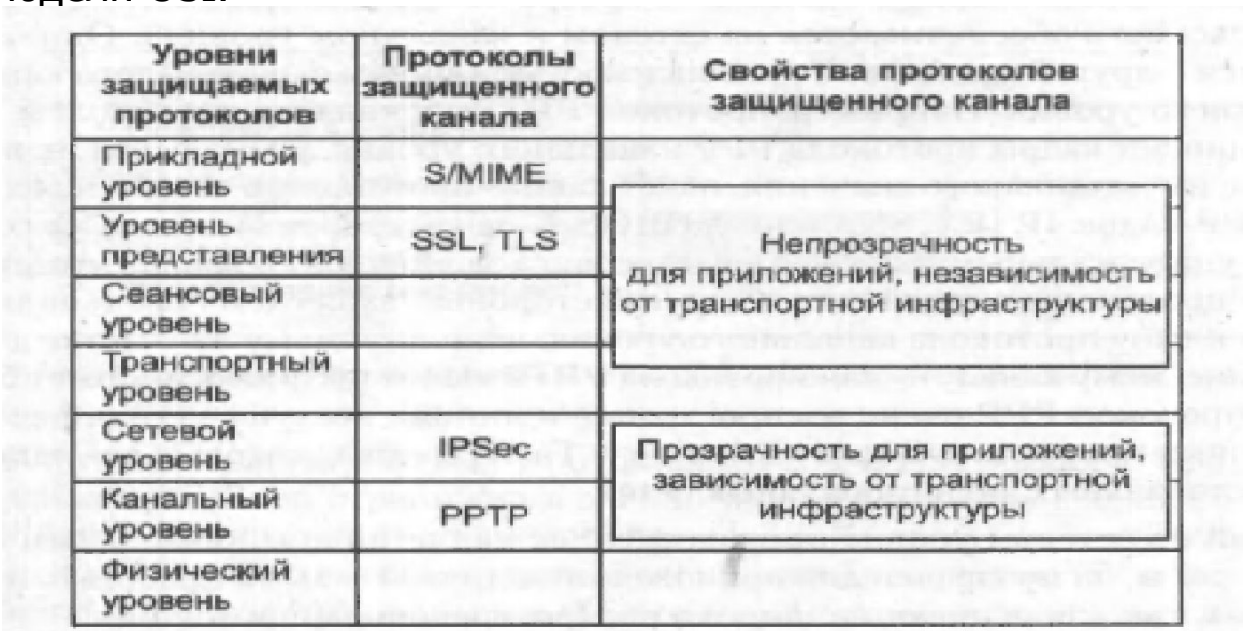
Понятие защищенного канала

Децентрализация заключается в том, что для каждого компьютера, которому требуется предоставить услуги защищенного канала, необходимо отдельно устанавливать, конфигурировать и администрировать программные средства защиты данных. Подключение каждого нового компьютера к защищенному каналу требует выполнять эти трудоемкие операции заново.

Во втором случае клиенты и серверы не участвуют в создании защищенного канала — он прокладывается только внутри публичной сети с коммутацией пакетов, например внутри Интернета. Так, канал может быть проложен между сервером удаленного доступа поставщика услуг публичной сети и пограничным маршрутизатором корпоративной сети. Это хорошо масштабируемое решение, управляемое централизовано администраторами как корпоративной сети, так и сети поставщика услуг. Для компьютеров корпоративной сети канал прозрачен — программное обеспечение этих конечных узлов остается без изменений. Такой гибкий подход позволяет легко образовывать новые каналы защищенного взаимодействия между компьютерами независимо от места их расположения. Реализация этого подхода сложнее — нужен **стандартный протокол образования защищенного канала**, требуется установка у всех поставщиков услуг программного обеспечения, поддерживающего такой протокол, необходима поддержка протокола производителями пограничного коммуникационного оборудования. Однако вариант, когда все заботы по поддержанию защищенного канала берет на себя поставщик услуг публичной сети, оставляет сомнения в надежности защиты: во-первых, незащищенными оказываются каналы доступа к публичной сети, во-вторых, потребитель услуг чувствует себя в полной зависимости от надежности поставщика услуг.

Иерархия технологий защищенного канала

Защищенный канал можно построить с помощью системных средств, реализованных на разных уровнях модели OSI.



Уровни защищаемых протоколов	Протоколы защищенного канала	Свойства протоколов защищенного канала
Прикладной уровень	S/MIME	Непрозрачность для приложений, независимость от транспортной инфраструктуры
Уровень представления	SSL, TLS	
Сеансовый уровень		
Транспортный уровень		
Сетевой уровень	IPSec	Прозрачность для приложений, зависимость от транспортной инфраструктуры
Канальный уровень	PPTP	
Физический уровень		

Если защита данных осуществляется средствами верхних уровнеи (прикладного, представления или сеансового), то такой способ защиты не зависит от технологий транспортировки данных (IP или IPX, Ethernet или ATM), что можно считать несомненным достоинством. В то же время приложения при этом становятся зависимыми от конкретного протокола защищенного канала, так как в них должны быть встроены явные вызовы функций этого протокола.

Защищенный канал, реализованный на самом высоком (прикладном) уровне, защищает только вполне определенную сетевую службу, например файловую, гипертекстовую или почтовую. Так, протокол S/MIME защищает исключительно сообщения электронной почты. При таком подходе для каждой службы необходимо разрабатывать собственную защищенную версию протокола.

Иерархия технологий защищенного канала

Популярный протокол SSL (Secure Socket Layer — слой защищенных сокетов) работает на уровне представления и создает защищенный канал, используя следующие технологии безопасности:

- взаимная аутентификация приложений на обоих концах защищенного канала выполняется путем обмена сертификатами (стандарт X.509);
- для контроля целостности передаваемых данных используются дайджесты;
- секретность обеспечивается шифрацией со средствами симметричных ключей сеанса.

Протокол SSL разработан компанией Netscape Communications для защиты данных, передаваемых между веб-сервером и веб-браузером, но он может быть использован и любыми другими приложениями. Работа протокола защищенного канала на уровне представления делает его более универсальным средством, чем протокол безопасности прикладного уровня. Однако для того чтобы приложение смогло воспользоваться протоколом уровня представления, в него по-прежнему приходится вносить исправления, хотя и не столь существенные, как в случае протокола прикладного уровня. Модификация приложения в данном случае сводится к встраиванию явных обращений к API соответствующего протокола безопасности.

Средства защищенного канала становятся прозрачными для приложений в тех случаях, когда безопасность обеспечивается на сетевом и канальном уровнях. Однако здесь мы сталкиваемся с другой проблемой — зависимостью сервиса защищенного канала от протокола нижнего уровня.

Иерархия технологий защищенного канала

Например, протокол PPTP, не являясь протоколом канального уровня, защищает кадры протокола PPP канального уровня, упаковывая их в IP-пакеты. При этом не имеет никакого значения, пакет какого протокола, в свою очередь, упакован в данном PPP-кадре: IP, IPX, SNA или NetBIOS. С одной стороны, это делает сервис PPTP достаточно универсальным, так как клиент сервиса защищенного канала может задействовать любые протоколы в своей сети. С другой стороны, такая схема предъявляет жесткие требования к типу протокола канального уровня, используемому на участке доступа клиента к защищенному каналу — для протокола PPTP таким протоколом может быть только PPP. Хотя протокол PPP очень распространен в линиях доступа, сегодня конкуренцию ему составляют протоколы Gigabit Ethernet и Fast Ethernet, которые все чаще работают не только в локальных, но и глобальных сетях.

Работающий на сетевом уровне **протокол IPSec** является компромиссным вариантом. С одной стороны, он прозрачен для приложений, с другой — может "работать практически во всех сетях, так как основан на широко распространенном протоколе IP и использует любую технологию канального уровня (PPP, Ethernet, ATM и т. д.).