Министерство образования РФ

Томский Государственный Университет
Факультет информатики
Кафедра теоретических основ информатики

Допустить к защите в ГАК
зав. кафедрой теоретических основ информатики,
кандидат технических наук, доцент

_____ Ю. Л. Костюк

"____" _____ 1999 г.

Кузьменко Алексей Васильевич

Исследование средств информационной безопасности систем и сетей на базе ОС UNIX и разработка методик ее обеспечения (Дипломная работа)

Научный руководитель:
Начальник информационно-аналитического центра
Томского Инновационного Центра Западной Сибири
_____ Голдаев Ю.С.

Исполнитель
студент группы 1441:
_____ Кузьменко А.В.

Томск 1999

Реферат

Дипломная работа, 184 стр., 7 рис., 13 табл.

ИНФОРМАЦИОННАЯ СИСТЕМА, БЕЗОПАСНОСТЬ, ВЫЧИСЛИТЕЛЬНАЯ СЕТЬ, СЕРВИС, ОПЕРАЦИОННАЯ СИСТЕМА, UNIX, SOLARIS, LINUX, МОДЕЛЬ, ЗАЩИТА ИНФОРМАЦИИ, ПОЛИТИКА БЕЗОПАСНОСТИ.

(1) Объект исследования: безопасность информационных систем и сетей.
(2) Цель работы: разработка методики создания безопасных информационных систем и оценки безопасности систем, уже существующих.
(3) Метод исследования: аналитический и экспериментальный на ЭВМ.
(4) Основные результаты: разработана методика создания и оценки безопасных систем. Разработана архитектурная модель безопасности информационной системы. Сформулированы требования к политике безопасности организации. Разработаны методы поддержания уровня безопасности системы в течение ее жизненного цикла и рекомендации по обнаружению и обработке нарушений политики безопасности. Разработана система обеспечения безопасности «SystemGuard» для ОС Linux.
(5) Новизна результатов: Имеющиеся на сегодняшний момент материалы в данной области морально устарели, не учитывают российской специфики, либо лишь выдвигают требования к системам, но не дают практических рекомендаций по обеспечению режима безопасности.
(6) Степень внедрения: построена система безопасности информационной системы Томского Инновационного Центра Западной Сибири. Разработана политика безопасности ИС ТИЦ, спланирована и реализована локальная сеть ТИЦ.
(8) Рекомендации по применению: Результаты могут быть использованы для обеспечения режима безопасности информации на предприятиях и в организациях. Некоторые главы могут быть полезны руководителям разных уровней для понимания важности и актуальности вопросов информационной безопасности.

Содержание

1. Введение

Проблема информационной безопасности возникла уже давно. Однако до сих пор не для всех очевидна ее важность и сложность. С каждым годом компьютеры и устройства с их применением все глубже проникают во все сферы жизни и деятельности человека. Все чаще электронные документы заменяют бумажные, все больше систем используют способность компьютера быстро обрабатывать информацию и хранить огромные объемы данных. Человечество стремится к модели информационного общества. Вместе с тем, по мере роста и усложнения информационных систем все более важной и одновременно сложной становится проблема обеспечения безопасности информации.

Вопросами информационной безопасности занимаются в разных странах и организациях не первый год. Первоначально эти проблемы интересовали только или почти только государственные органы. Под вопросами информационной безопасности понимались в основном, вопросы обеспечения секретности, конфиденциальности данных. Симптоматично, что первый документ, касающийся вопросов информационной безопасности, получивший широкую известность был выпущен Министерством обороны США. В 1983 году это ведомство выпустило книгу под названием «Критерии оценки надежных вычислительных систем» (Trusted Computer Systems Evaluation Criteria, TCSEC) [6]. Поскольку эта книга имела яркую оранжевую обложку, в обиходе она так и получила название – «Оранжевая книга». Эта книга положила начало систематическому подходу к оценке надежности информационных систем. Вместе с тем, нельзя не отметить, что подходы «Оранжевой книги» довольно специфичны, что, впрочем, легко объяснить, принимая во внимание, в недрах какого ведомства она была разработана.

В конце 80-х годов в Европейском экономическом сообществе были разработаны аналогичные по значению «Гармонизированные критерии» [7]. А в начале 90-х и в России государство обратило внимание на безопасность информации – Государственная техническая комиссия при Президенте России издала брошюры, посвященные проблеме защиты от несанкционированного доступа [1-5].

В последние несколько лет можно отметить рост интереса к проблемам построения безопасных информационных систем во всем мире, в том числе и в России. Это объясняется теми самыми причинами все более широкого распространения вычислительной техники, о которых было сказано ранее. Так как все больше бизнес-процессов оказываются зависящими от надежности информационных систем, соответственно, все больше внимания обращается на эту самую надежность.

К сожалению, отечественный рынок не балует нас изобилием книг, посвященных данной тематике. Большинство имеющейся литературы либо сильно устарело и не отражает современных реалий, либо является переводами соответствующих книг зарубежных авторов. Последние, несомненно, полезны в плане изучения зарубежного опыта, однако вряд ли применимы в наших условиях. Кроме того, такие книги как, например, упомянутая выше «Оранжевая книга» или Европейские критерии, лишь выдвигают требования к системам, но не дают практических рекомендаций по обеспечению режима безопасности.

Настоящая работа ставит перед собой две цели: с одной стороны, создать некую архитектурную модель безопасности информационной системы, с другой стороны, опираясь

на эту модель, рассмотреть практические аспекты создания безопасных информационных систем или оценки безопасности систем уже существующих. Применение и внедрение предложенных рекомендаций способно значительно снизить возможные потери в случае возникновения опасных ситуаций. Поскольку разработанные автором рекомендации в большинстве своем несут общий характер, их использование не ограничивается только операционной системой UNIX. Напротив, большинство из них применимо и в других системах. По сути своей, важен подход, модель безопасности, а не конкретная реализация.

Сразу отметим, что под современной информационной системой подразумевается система обработки информации, построенная на архитектуре клиент/сервер. Это связано как с ростом популярности этого подхода при построении и проектировании информационных систем в наши дни, так и с тем, что его применение позволяет легко декомпозировать любую систему на простые составляющие. Последнее свойство особенно важно, потому что современные информационные системы столь сложны, что рассмотрение их в целом, пожалуй, нереально.

Общее построение работы таково. Во второй главе приводится краткий обзор стандартов в области информационной безопасности, принятых в мире. Третья глава посвящена рассмотрению предлагаемой автором модели для анализа информационных систем с точки зрения безопасности. Впрочем, возможно такая модель будет применима не только в сфере безопасности. В четвертой главе приводятся практические рекомендации по построению безопасных систем, а так же по оценке безопасности уже существующих систем. Пятая глава содержит некоторые примеры реализации описанных в третьей и четвертой главах подходов на примере операционной системы Solaris. Наконец, шестая глава описывает реализацию вышеописанных подходов на примере ОС Linux и системы SystemGuard, разработанной для Linux и внедренной автором на www-сервере Томского Инновационного Центра Западной Сибири ТИЦ ЗС.

Предлагаемая работа основана на более чем двухлетнем опыте обеспечения и поддержания режима информационной безопасности в ТИЦ ЗС.

2. Обзор стандартов в области информационной безопасности

В этой главе рассматриваются стандарты и рекомендации, принятые в различных странах для оценки безопасности информационных систем. Разумеется, стандарты и рекомендации не способны дать практических советов о построении безопасных систем, однако, они полезны с точки зрения изучения основных требований, предъявляемых к таким системам. К тому же, знание критериев оценки безопасных систем полезно и при выборе и комплектовании программно-аппаратных конфигураций будущих информационных систем.

Знание критериев оценки полезно еще и для того, чтобы администраторы знали, на какие моменты следует обращать внимание при оценке безопасности собственных систем. В этом случае, приведенные здесь стандарты и рекомендации помогут правильно расставить акценты при выполнении такого рода работ.

2.1. Критерии оценки надежных информационных систем минобороны США

Данный труд, получивший в обиходе название «Оранжевая книга» из-за цвета обложки первоначального издания, был впервые опубликован Министерством обороны США в 1983 году. Его можно считать первой попыткой систематизации требований к информационным системам, работающим с критической информацией.

"Оранжевая книга" поясняет понятие безопасной системы, которая "управляет, посредством соответствующих средств, доступом к информации, так что только должным образом авторизованные лица или процессы, действующие от их имени, получают право читать, писать, создавать и удалять информацию". Очевидно, однако, что абсолютно безопасных систем не существует, что это абстракция. Любую систему можно "взломать", если располагать достаточно большими материальными и временными ресурсами. Есть смысл оценивать лишь степень доверия, которое разумно оказать той или иной системе.

2.1.1. Основные понятия

В "Оранжевой книге" надежная система определяется как "система, использующая достаточные аппаратные и программные средства, чтобы обеспечить одновременную обработку информации разной степени секретности группой пользователей без нарушения прав доступа". Степень доверия, или надежность систем, оценивается по двум основным критериям:

Политика безопасности - набор законов, правил и норм поведения, определяющих, как организация обрабатывает, защищает и распространяет информацию. В частности, правила определяют, в каких случаях пользователь имеет право оперировать с определенными наборами данных. Чем надежнее система, тем строже и многообразнее должна быть политика безопасности. В зависимости от сформулированной политики можно выбирать конкретные механизмы, обеспечивающие безопасность системы. Политика безопасности – это активный компонент защиты, включающий в себя анализ возможных угроз и выбор мер противодействия.

Гарантированность - мера доверия, которая может быть оказана архитектуре и реализации системы. Гарантированность может проистекать как из тестирования, так и из проверки (формальной или нет) общего замысла и исполнения системы в целом и ее компонентов.

Гарантированность показывает, насколько корректны механизмы, отвечающие за проведение в жизнь политики безопасности. Гарантированность можно считать пассивным компонентом защиты, надзирающим за самими защитниками.

Важным средством обеспечения безопасности является механизм подотчетности (протоколирования). Надежная система должна фиксировать все события, касающиеся безопасности. Ведение протоколов должно дополняться аудитом, то есть анализом регистрационной информации.

Концепция надежной вычислительной базы является центральной при оценке степени гарантированности, с которой систему можно считать надежной. Надежная вычислительная база - это совокупность защитных механизмов компьютерной системы (включая аппаратное и программное обеспечение), отвечающих за проведение в жизнь политики безопасности. Надежность вычислительной базы определяется исключительно ее реализацией и корректностью исходных данных, которые вводит административный персонал (например, это могут быть данные о степени благонадежности пользователей).

Вообще говоря, компоненты вне вычислительной базы могут не быть надежными, однако это не должно влиять на безопасность системы в целом. В результате, для оценки надежности компьютерной системы достаточно рассмотреть только ее вычислительную базу, которая, как можно надеяться, достаточно компактна.

Основное назначение надежной вычислительной базы - выполнять функции монитора обращений, то есть контролировать допустимость выполнения субъектами определенных операций над объектами. Монитор проверяет каждое обращение пользователя к программам или данным на предмет согласованности со списком действий, допустимых для пользователя.

От монитора обращений требуется выполнение трех свойств:

- Изолированность. Монитор должен быть защищен от отслеживания своей работы.
- Полнота. Монитор должен вызываться при каждом обращении, не должно быть способов его обхода.
- Верифицируемость. Монитор должен быть компактным, чтобы его можно было проанализировать и протестировать, будучи уверенным в полноте тестирования.

Реализация монитора обращений называется ядром безопасности. Ядро безопасности - это основа, на которой строятся все защитные механизмы. Помимо перечисленных выше свойств монитора обращений, ядро должно гарантировать собственную неизменность.

Границу надежной вычислительной базы называют периметром безопасности. Как уже указывалось, от компонентов, лежащих вне периметра безопасности, вообще говоря, не требуется надежности. С развитием распределенных систем понятию "периметр безопасности" все чаще придают другой смысл, имея в виду границу владений определенной организации. То, что внутри владений, считается надежным, а то, что вне - нет. Связь между внутренним и внешним мирами осуществляют посредством шлюзовой системы, которая по идее способна противостоять потенциально ненадежному или даже враждебному окружению.

2.1.2. Основные элементы политики безопасности

Согласно "Оранжевой книге", политика безопасности обязательно должна включать в себя следующие элементы:

- Произвольное управление доступом.
- Безопасность повторного использования объектов.
- Метки безопасности.
- Принудительное управление доступом.

Ниже следует подробное рассмотрение перечисленных элементов.

2.1.2.1. Произвольное управление доступом

Произвольное управление доступом - это метод ограничения доступа к объектам, основанный на учете личности субъекта или группы, в которую субъект входит. Произвольность управления состоит в том, что некоторое лицо (обычно владелец объекта) может по своему усмотрению давать другим субъектам или отбирать у них права доступа к объекту.

С концептуальной точки зрения текущее состояние прав доступа при произвольном управлении описывается матрицей, в строках которой перечислены субъекты, а в столбцах - объекты. В клетках, расположенных на пересечении строк и столбцов, записываются способы доступа, допустимые для субъекта по отношению к объекту - например, чтение, запись, выполнение, возможность передачи прав другим субъектам и т.п.

Очевидно, прямолинейное представление подобной матрицы невозможно (поскольку она очень велика). На практике, хранить данную матрицу целиком и не нужно, поскольку большинство клеток в ней пусты. В операционных системах более компактное представление матрицы доступа основывается на структурировании совокупности субъектов (владелец/группа/прочие в ОС UNIX) или на механизме списков управления доступом, то есть на представлении матрицы по столбцам, когда для каждого объекта перечисляются субъекты вместе с их правами доступа. За счет использования метасимволов можно компактно описывать группы субъектов, удерживая тем самым размеры списков управления доступом в разумных рамках.

Большинство операционных систем и систем управления базами данных реализуют именно произвольное управление доступом. Главное его достоинство - гибкость, главные недостатки - рассредоточенность управления и сложность централизованного контроля, а также оторванность прав доступа от данных, что позволяет копировать секретную информацию в общедоступные файлы.

2.1.2.2. Безопасность повторного использования объектов

Безопасность повторного использования объектов - важное на практике дополнение средств управления доступом, предохраняющее от случайного или преднамеренного извлечения секретной информации из "мусора". Безопасность повторного использования должна гарантироваться для областей оперативной памяти (в частности, для буферов с образами экрана, расшифрованными паролями и т.п.), для дисковых блоков и магнитных носителей в целом.

Важно обратить внимание на следующий момент. Поскольку информация о субъектах также представляет собой объект, необходимо позаботиться о безопасности "повторного использования субъектов". Когда пользователь покидает организацию, следует не только лишить его возможности входа в систему, но и запретить доступ ко всем объектам. В

противном случае, новый сотрудник может получить ранее использовавшийся идентификатор, а с ним и все права своего предшественника.

Современные интеллектуальные периферийные устройства усложняют обеспечение безопасности повторного использования объектов. Действительно, принтер может буферизовать несколько страниц документа, которые останутся в памяти даже после окончанию печати. Необходимо предпринять специальные меры, чтобы "вытолкнуть" их оттуда.

Впрочем, иногда организации защищаются от повторного использования слишком ревностно - путем уничтожения магнитных носителей. На практике троекратной записи на носитель случайных последовательностей бит вполне достаточно.

2.1.2.3. Метки безопасности

Для реализации принудительного управления доступом с субъектами и объектами ассоциируются метки безопасности. Метка субъекта описывает его благонадежность, метка объекта - степень закрытости содержащейся в нем информации.

Согласно "Оранжевой книге", метки безопасности состоят из двух частей - уровня секретности и списка категорий. Уровни секретности, поддерживаемые системой, образуют упорядоченное множество, которое может выглядеть, например, так:

- совершенно секретно,
- секретно,
- конфиденциально,
- несекретно.

Впрочем, для разных систем набор уровней секретности может различаться.

Категории образуют неупорядоченный набор. Их назначение - описать предметную область, к которой относятся данные. В военном окружении каждая категория может соответствовать, например, определенному виду вооружений. Механизм категорий позволяет разделить информацию по отсекам, что способствует лучшей защищенности. В следующем пункте мы подробно рассмотрим правила принудительного управления доступом, здесь же отметим, что субъект не может получить доступ к "чужим" категориям, даже если его уровень благонадежности - "совершенно секретно". Специалист по танкам не узнает тактико-технические данные самолетов.

Главная проблема, которую необходимо решать в связи с метками, это обеспечение их целостности. Во-первых, не должно быть непомеченных субъектов и объектов, иначе в меточной безопасности появятся легко используемые бреши. Во-вторых, при любых операциях с данными метки должны оставаться правильными. В особенности это относится к экспорту и импорту данных. Например, печатный документ должен открываться заголовком, содержащим текстовое и/или графическое представление метки безопасности. Аналогично, при передаче файла по каналу связи должна передаваться и ассоциированная с ним метка, причем в таком виде, чтобы удаленная система могла ее правильно интерпретировать, несмотря на возможные различия в уровнях секретности и наборе категорий.

Одним из средств обеспечения целостности меток безопасности является разделение устройств на многоуровневые и одноуровневые. На многоуровневых устройствах может храниться информация разного уровня секретности (точнее, лежащая в определенном

диапазоне уровней). Одноуровневое устройство можно рассматривать как вырожденный случай многоуровневого, когда допустимый диапазон состоит из одного уровня. Зная уровень устройства, система может решить, допустимо ли записывать на него информацию с определенной меткой. Например, попытка напечатать совершенно секретную информацию на принтере общего пользования с уровнем "несекретно" потерпит неудачу.

Метки безопасности, ассоциируемые с субъектами, более подвижны, чем метки объектов. Субъект может в течение сеанса работы с системой изменять свою метку, естественно, не выходя за предопределенные для него рамки. Иными словами, он может сознательно занижать свой уровень благонадежности, чтобы уменьшить вероятность непреднамеренной ошибки. Вообще, принцип минимизации привилегий - весьма разумное средство защиты.

2.1.2.4. Принудительное управление доступом

Принудительное управление доступом основано на сопоставлении меток безопасности субъекта и объекта.

Субъект может читать информацию из объекта, если уровень секретности субъекта не ниже, чем у объекта, а все категории, перечисленные в метке безопасности объекта, присутствуют в метке субъекта. В таком случае говорят, что метка субъекта доминирует над меткой объекта. Смысл сформулированного правила понятен - читать можно только то, что положено.

Субъект может записывать информацию в объект, если метка безопасности объекта доминирует над меткой субъекта. В частности, "конфиденциальный" субъект может писать в секретные файлы, но не может – в несекретные (разумеется, должны также выполняться ограничения на набор категорий). На первый взгляд подобное ограничение может показаться странным, однако оно вполне разумно. Ни при каких операциях уровень секретности информации не должен понижаться, хотя обратный процесс вполне возможен. Посторонний человек может случайно узнать секретные сведения и сообщить их куда следует, однако лицо, допущенное к работе с секретными документами, не имеет права раскрывать их содержание простому смертному.

Описанный способ управления доступом называется принудительным, поскольку он не зависит от воли субъектов (даже системных администраторов). После того, как зафиксированы метки безопасности субъектов и объектов, оказываются зафиксированными и права доступа. В терминах принудительного управления нельзя выразить предложение "разрешить доступ к объекту X еще и для пользователя Y". Конечно, можно изменить метку безопасности пользователя Y, но тогда он, скорее всего, получит доступ ко многим дополнительным объектам, а не только к X.

Принудительное управление доступом реализовано во многих вариантах операционных систем и СУБД, отличающихся повышенными мерами безопасности. В частности, такие варианты существуют и для ОС Solaris, и для различных СУБД. Независимо от практического использования, принципы принудительного управления являются удобным методологическим базисом для начальной классификации информации и распределения прав доступа. Удобнее мыслить в терминах уровней секретности и категорий, чем заполнять неструктурированную матрицу доступа. Впрочем, в реальной жизни произвольное и

принудительное управление доступом сочетается в рамках одной системы, что позволяет использовать сильные стороны обоих подходов.

2.1.3. Подотчетность

Если понимать политику безопасности узко, то есть как правила разграничения доступа, то механизм подотчетности является дополнением подобной политики. Цель подотчетности - в каждый момент времени знать, кто работает в системе и что он делает. Средства подотчетности делятся на три категории:

- Идентификация и аутентификация.
- Предоставление надежного пути.
- Анализ регистрационной информации.

Рассмотрим эти категории подробнее.

2.1.3.1. Идентификация и аутентификация

Каждый пользователь, прежде чем получить право совершать какие-либо действия в системе, должен идентифицировать себя. Обычный способ идентификации - ввод имени пользователя при входе в систему. В свою очередь, система должна проверить подлинность личности пользователя, то есть что он является именно тем, за кого себя выдает. Стандартное средство проверки подлинности (аутентификации) - пароль, хотя в принципе могут использоваться также разного рода личные карточки, биометрические устройства (сканирование роговицы или отпечатков пальцев) или их комбинация.

Идентификация и аутентификация - первый и важнейший программно-технический рубеж информационной безопасности. Если не составляет проблемы получить доступ к системе под любым именем, то другие механизмы безопасности, например, управление доступом, очевидно, теряют смысл. Очевидно и то, что без идентификации пользователей невозможно протоколирование их действий. В силу перечисленных причин проверке подлинности должно придаваться первостепенное значение. Существует целая серия публикаций правительственных ведомств США, разъясняющих вопросы аутентификации и, в частности, проблемы, связанные с паролями. Например, декларируется, что пользователю позволено менять свой пароль, что пароли, как правило, должны быть машинно-сгенерированными (а не выбранными "вручную"), что пользователю должна предоставляться некоторая регистрационная информация (дата и время последнего входа в систему и т.п.).

2.1.3.2. Предоставление надежного пути

Надежный путь связывает пользователя непосредственно с надежной вычислительной базой, минуя другие, потенциально опасные компоненты системы. Цель предоставления надежного пути - дать пользователю возможность убедиться в подлинности обслуживающей его системы.

Относительно несложно реализовать надежный путь, если используется неинтеллектуальный терминал - достаточно иметь зарезервированную управляющую последовательность (при условии защищенности линии связи между терминалом и системой). Если же пользователь общается с интеллектуальным терминалом, персональным компьютером или рабочей станцией, задача обеспечения надежного пути становится чрезвычайно сложной, если вообще разрешимой. Как гарантировать, что пользователь

общается с подлинной программой login, а не с "Троянским конем"? Возможно, по этой причине о предоставлении надежного пути упоминают редко, хотя на практике данный аспект весьма важен.

2.1.3.3. Анализ регистрационной информации

Аудит имеет дело с действиями (событиями), так или иначе затрагивающими безопасность системы. К числу таких событий относятся:

- Вход в систему (успешный или нет).
- Выход из системы.
- Обращение к удаленной системе.
- Операции с файлами (открыть, закрыть, переименовать, удалить).
- Смена привилегий или иных атрибутов безопасности (режима доступа, уровня благонадежности пользователя и т.п.).

Можно назвать и другие события - например, смену набора регистрируемых действий. Полный перечень событий, потенциально подлежащих регистрации, зависит от избранной политики безопасности и от специфики системы.

Если фиксировать все события, объем регистрационной информации, скорее всего, будет расти слишком быстро, а ее эффективный анализ станет невозможным. "Оранжевая книга" предусматривает наличие средств выборочного протоколирования, как в отношении пользователей, так и в отношении событий.

Протоколирование помогает следить за пользователями и реконструировать прошедшие события. Слежка важна в первую очередь как профилактическое средство. Можно надеяться, что многие воздержатся от нарушений безопасности, зная, что их действия фиксируются. Реконструкция событий позволяет проанализировать случаи нарушений, понять, почему они стали возможны, оценить размеры ущерба и принять меры по недопущению подобных нарушений в будущем.

При протоколировании события обязательно должна быть записана следующая информация:

- Дата и время события.
- Уникальный идентификатор пользователя - инициатора действия.
- Тип события.
- Результат действия (успех или неудача).
- Источник запроса (например, имя терминала).
- Имена затронутых объектов (например, открываемых или удаляемых файлов).
- Описание изменений, внесенных в базы данных защиты (например, новая метка безопасности объекта).
- Метки безопасности субъектов и объектов события.

Необходимо подчеркнуть важность не только сбора информации, но и ее регулярного и целенаправленного анализа. В плане анализа выгодное положение занимают средства аудита СУБД, поскольку к регистрационной информации могут естественным образом применяться произвольные SQL-запросы. Следовательно, появляется возможность для выявления подозрительных действий применять сложные запросы.

2.1.4. Гарантированность

Гарантированность - это мера уверенности, с которой можно утверждать, что для проведения в жизнь сформулированной политики безопасности выбран подходящий набор средств, и что каждое из этих средств правильно исполняет отведенную ему роль.

В "Оранжевой книге" рассматривается два вида гарантированности - операционная и технологическая. Операционная гарантированность относится к архитектурным и реализационным аспектам системы, в то время как технологическая - к методам построения и сопровождения.

2.1.4.1. Операционная гарантированность

Операционная гарантированность включает в себя проверку следующих элементов:

- Архитектура системы.
- Целостность системы.
- Анализ тайных каналов передачи информации.
- Надежное администрирование.
- Надежное восстановление после сбоев.

Операционная гарантированность - это способ убедиться в том, что архитектура системы и ее реализация действительно проводят в жизнь избранную политику безопасности.

Архитектура системы должна способствовать реализации мер безопасности или прямо поддерживать их. Примеры подобных архитектурных решений в рамках аппаратуры и операционной системы – разделение команд по уровням привилегированности, защита различных процессов от взаимного влияния за счет выделения каждому своего виртуального пространства, особая защита ядра ОС.

В принципе меры безопасности не обязательно должны быть заранее встроены в систему – достаточно принципиальной возможности дополнительной установки защитных продуктов. Так, сугубо ненадежная система MS-DOS может быть улучшена за счет средств проверки паролей доступа к компьютеру и/или жесткому диску, за счет борьбы с вирусами путем отслеживания попыток записи в загрузочный сектор CMOS-средствами и т.п. Тем не менее, по-настоящему надежная система должна изначально проектироваться с упором на механизмы безопасности.

Среди архитектурных решений, предусматриваемых "Оранжевой книгой", упомянем следующие:

Деление аппаратных и системных функций по уровням привилегированности и контроль обмена информацией между уровнями.

- Защита различных процессов от взаимного влияния за счет механизма виртуальной памяти.
- Наличие средств управления доступом.
- Структурированность системы, явное выделение надежной вычислительной базы, обеспечение компактности этой базы.
- Следование принципу минимизации привилегий - каждому компоненту дается ровно столько привилегий, сколько необходимо для выполнения им своих функций.
- Сегментация (в частности, сегментация адресного пространства процессов) как средство повышения надежности компонентов.

- Целостность системы в данном контексте означает, что аппаратные и программные компоненты надежной вычислительной базы работают должным образом и что имеется аппаратное и программное обеспечение для периодической проверки целостности.
- Анализ тайных каналов передачи информации - тема, специфичная для режимных систем, когда главное - обеспечить конфиденциальность информации. Тайным называется канал передачи информации, не предназначенный для обычного использования. Шпионская аналогия - горшок с геранью в окне как сигнал опасности.

Различают тайные каналы с памятью и временные (ударение на "ы"). Тайные каналы с памятью используют изменения хранимых объектов. Тайным знаком может быть размер файла, имя файла (составленное, например, из входного имени и пароля атакуемого субъекта), число пробелов между словами и т.д. Тайный канал считается быстрым, если с его помощью можно передавать 100 или более бит в секунду.

Временные каналы передают информацию за счет изменения временных характеристик процессов – например, времени обработки запроса.

Обычно тайные каналы используются не столько для передачи информации от одного злоумышленника другому, сколько для получения злоумышленником сведений от внедренного в систему "Троянского коня".

Не очень понятно, как на практике, в распределенной системе, выявлять тайные каналы (хотя, после выявления, пропускную способность оценить можно). Как показывают предыдущие рассмотрения, тайным каналом может служить почти все, что угодно, а скорости современных процессоров и периферийных устройств делают опасными даже прямолинейные способы передачи. Вероятно, только для статичной конфигурации можно с разумной полнотой описать возможные тайные каналы передачи информации.

Надежное администрирование в трактовке "Оранжевой книги" означает всего лишь, что должны быть логически выделены три роли - системного администратора, системного оператора и администратора безопасности. Физически эти обязанности может выполнять один человек, но, в соответствии с принципом минимизации привилегий, в каждый момент времени он должен выполнять только одну из трех ролей. Конкретный набор обязанностей администраторов и оператора зависит от специфики организации.

Надежное восстановление после сбоев - вещь необходимая, но ее реализация может быть сопряжена с серьезными техническими трудностями. Прежде всего, должна быть сохранена целостность информации и, в частности, целостность меток безопасности. В принципе возможна ситуация, когда сбой приходится на момент записи нового файла с совершенно секретной информацией. Если файл окажется с неправильной меткой, информация может быть скомпрометирована. Далее, на период восстановления система не должна оставаться беззащитной. Нельзя допускать промежуточных состояний, когда защитные механизмы полностью или частично отключены, а доступ пользователей разрешен.

Вообще говоря, надежное восстановление включает в себя два вида деятельности - подготовку к сбою (отказу) и собственно восстановление. Подготовка к сбою - это и регулярное выполнение резервного копирования, и выработка планов действий в экстренных

случаях, и поддержание запаса резервных компонентов. Восстановление, вероятно, связано с перезагрузкой системы и выполнением ремонтных и/или административных процедур.

2.1.4.2. Технологическая гарантированность

Технологическая гарантированность охватывает весь жизненный цикл системы, то есть периоды проектирования, реализации, тестирования, продажи и сопровождения. Все перечисленные действия должны выполняться в соответствии с жесткими стандартами, чтобы обезопаситься от утечки информации и нелегальных "закладок".

Первое, на что обычно обращают внимание, это тестирование. Изготовитель или поставщик выполняет набор тестов, документирует его и предоставляет на рассмотрение аттестационной комиссии, которая проверяет полноту набора и, быть может, выполняет свои тесты. Вообще говоря, тестированию подлежат как собственно механизмы безопасности, так и пользовательский интерфейс к ним. Тесты должны показать, что защитные механизмы функционируют в соответствии со своим описанием и что не существует очевидных способов обхода или разрушения защиты. Тесты должны продемонстрировать действенность средств управления доступом, защищенность регистрационной и аутентификационной информации. Должна быть уверенность, что надежную вычислительную базу нельзя привести в состояние, когда она перестанет обслуживать пользовательские запросы (пожалуй, это единственное упоминание в "Оранжевой книге" такого аспекта информационной безопасности, как доступность).

Верификация описания архитектуры - это выполненное автоматически формальное доказательство того, что архитектура системы соответствует сформулированной политике безопасности. Национальный центр компьютерной безопасности США располагает двумя системами для проведения подобных формальных доказательств - Gypsy Verification Environment (GVE) компании Computational Logic, Inc. и Formal Development Methodology (FDM) корпорации UNISYS.

Средства конфигурационного управления защищают надежную систему в процессе проектирования, реализации и сопровождения. Конфигурационное управление включает в себя идентификацию, протоколирование и анализ всех изменений, вносимых в надежную вычислительную базу (независимо от того, идет ли речь об аппаратуре или программах), а также (что прямо следует из названия) управление процессом внесения изменений.

Конфигурационное управление давно и широко используется разработчиками программного обеспечения отнюдь не только (и не столько) по соображениям безопасности. Специфика подхода "Оранжевой книги" - в тотальном контроле изменений и в строгой дисциплине их проведения.

Надежное распределение защищает систему в процессе ее передачи от поставщика клиенту. Оно включает в себя два комплекса мер - по защите и по проверке. Защитная часть работает на пути от поставщика к клиенту. Она позволяет поставщику утверждать, что клиент получил именно то, поставщик отгрузил, что передана нужная версия, все последние изменения, и что по дороге система не была вскрыта и в нее не были внесены изменения. Среди защитных механизмов - надежная упаковка, предохраняющая от вредного воздействия окружающей среды, надежная транспортировка и, наконец, надежная инсталляция аппаратуры и программ.

Проверочные меры применяются клиентом, чтобы убедиться, что он получил именно то, что заказал, и что система не подверглась нелегальным изменениям. Клиент должен убедиться, что полученный им продукт – точная копия эталонного варианта, имеющегося у поставщика. Для этого существует целый спектр методов, начиная от проверок серийных номеров аппаратных компонентов и кончая верификацией контрольных сумм программ и данных. Следует отметить, что современная практика поставки программного обеспечения на CD-ROM существенно затрудняет (по сравнению с поставкой на дискетах) внесение нелегальных изменений.

2.1.5. Документация

Документация - необходимое условие гарантированной надежности системы и, одновременно, - инструмент проведения политики безопасности. Без документации люди не будут знать, какой политике следовать и что для этого нужно делать.

Согласно "Оранжевой книге", в комплект документации надежной системы должны входить следующие тома:

- Руководство пользователя по средствам безопасности.
- Руководство администратора по средствам безопасности.
- Тестовая документация.
- Описание архитектуры.

Разумеется, на практике требуется еще минимум одна книга - письменное изложение политики безопасности данной организации.

Руководство пользователя по средствам безопасности предназначено для обычных, непривилегированных людей. Оно должно содержать сведения о механизмах безопасности и способах их использования. Руководство должно давать ответы по меньшей мере на следующие вопросы:

Как входить в систему? Как вводить имя и пароль? Как менять пароль? Как часто это нужно делать? Как выбирать новый пароль? Как защищать файлы и другую информацию? Как задавать права доступа к файлам? Из каких соображений это нужно делать? Как импортировать и экспортировать информацию, не нарушая правил безопасности? Как уживаться с системными ограничениями? Почему эти ограничения необходимы? Какой стиль работы сделает ограничения необременительными?

Руководство администратора по средствам безопасности предназначено и для системного администратора, и для администратора безопасности. В Руководстве освещаются вопросы начального конфигурирования системы, перечисляются текущие обязанности администратора, анализируется соотношения между безопасностью и эффективностью функционирования.

Типичное оглавление Руководства администратора включает в себя следующие пункты:

Каковы основные защитные механизмы? Как администрировать средства идентификации и аутентификации? В частности, как заводить новых пользователей и удалять старых? Как администрировать средства произвольного управления доступом? Как защищать системную информацию? Как обнаруживать слабые места? Как администрировать средства протоколирования и аудита? Как выбирать регистрируемые события? Как анализировать результаты? Как администрировать средства принудительного управления

доступом? Какие уровни секретности и категории выбрать? Как назначать и менять метки безопасности? Как генерировать новую, переконфигурированную надежную вычислительную базу? Как безопасно запускать систему и восстанавливать ее после сбоев и отказов? Как организовать резервное копирование? Как разделить обязанности системного администратора и оператора?

Тестовая документация содержит описания тестов и их результаты. По идее она проста, но зачастую весьма пространна. Кроме того (вернее, перед тем), тестовая документация должна содержать план тестирования и условия, налагаемые на тестовое окружение.

Описание архитектуры в данном контексте обязательно должно включать в себя сведения о внутреннем устройстве надежной вычислительной базы. Вообще говоря, это описание должно быть формальным, допускающим автоматическое сопоставление с политикой безопасности на предмет соответствия требованиям последней. Объем описания архитектуры может оказаться сопоставимым с объемом исходных текстов программной реализации системы.

2.1.6. Классы безопасности

"Критерии" Министерства обороны США открыли путь к ранжированию информационных систем по степени надежности. В "Оранжевой книге" определяется четыре уровня безопасности (надежности) - D, C, B и A. Уровень D предназначен для систем, признанных неудовлетворительными. В настоящее время он пуст и ситуация едва ли когда-нибудь изменится. По мере перехода от уровня C к A к надежности систем предъявляются все более жесткие требования. Уровни C и B подразделяются на классы (C1, C2, B1, B2, B3) с постепенным возрастанием надежности. Таким образом, всего имеется шесть классов безопасности - C1, C2, B1, B2, B3, A1. Чтобы система в результате процедуры сертификации могла быть отнесена к некоторому классу, ее политика безопасности и гарантированность должны удовлетворять приводимым ниже требованиям. Поскольку при переходе к каждому следующему классу требования только добавляются, мы будем выписывать лишь то новое, что присуще данному классу, группируя требования в согласии с предшествующим изложением.

Итак, ниже следуют критерии оценки надежных компьютерных систем.

2.1.6.1. Требования к политике безопасности

Требования к политике безопасности, проводимой системой, подразделяются в соответствии с основными направлениями политики, предусматриваемыми "Оранжевой книгой".

Произвольное управление доступом:

- Класс C1 - надежная вычислительная база должна управлять доступом именованных пользователей к именованным объектам. Механизм управления (права для владельца/группы/прочих, списки управления доступом) должен позволять пользователям специфицировать разделение файлов между индивидами и/или группами.
- Класс C2 - в дополнение к C1, права доступа должны гранулироваться с точностью до пользователя. Механизм управления должен ограничивать распространение прав

доступа - только авторизованный пользователь (например, владелец объекта) может предоставлять права доступа другим пользователям. Все объекты должны подвергаться контролю доступа.

- Класс B3 - в дополнение к C2, должны обязательно использоваться списки управления доступом с указанием разрешенных режимов. Должна быть возможность явного указания пользователей или их групп, доступ которых к объекту запрещен.
- (Примечание. Поскольку классы B1 и B2 не упоминаются, требования к ним в плане произвольного управления доступом те же, что и для C2. Аналогично, требования к классу A1 те же, что и для B3.)

Повторное использование объектов:

- Класс C2 - при выделении хранимого объекта из пула ресурсов надежной вычислительной базы необходимо ликвидировать все следы предыдущих использований.

Метки безопасности:

- Класс B1 - надежная вычислительная база должна управлять метками безопасности, ассоциируемыми с каждым субъектом и хранимым объектом. Метки являются основой функционирования механизма принудительного управления доступом. При импорте непомеченной информации соответствующий уровень секретности должен запрашиваться у авторизованного пользователя и все такие действия следует протоколировать.
- Класс B2 - в дополнение к B1, помечаться должны все ресурсы системы (например, ПЗУ), прямо или косвенно доступные субъектам.

Целостность меток безопасности:

- Класс B1 - метки должны адекватно отражать уровни секретности субъектов и объектов. При экспорте информации метки должны преобразовываться в точное и однозначно трактуемое внешнее представление, сопровождающее данные. Каждое устройство ввода/вывода (в том числе коммуникационный канал) должно трактоваться как одноуровневое или многоуровневое. Все изменения трактовки и ассоциированных уровней секретности должны протоколироваться.
- Класс B2 - в дополнение к B1, надежная вычислительная база должна немедленно извещать терминального пользователя об изменении его метки безопасности. Пользователь может запросить информацию о своей метке. Надежная вычислительная база должна поддерживать присваивание всем подключенным физическим устройствам минимального и максимального уровня секретности. Эти уровни должны использоваться при проведении в жизнь ограничений, налагаемых физической конфигурацией системы (например, расположением устройств).

Принудительное управление доступом:

- Класс B1 - надежная вычислительная база должна обеспечить проведение в жизнь принудительного управления доступом всех субъектов ко всем хранимым объектам.

Субъектам и объектам должны быть присвоены метки безопасности, являющиеся комбинацией упорядоченных уровней секретности, а также категорий. Метки являются основой принудительного управления доступом. Надежная вычислительная база должна поддерживать как минимум два уровня секретности. Субъект может читать объект, если его (субъекта) метка безопасности доминирует над меткой безопасности объекта, то есть уровень секретности субъекта не меньше уровня секретности объекта и все категории объекта входят в метку безопасности субъекта. Субъект может писать в объект, если метка безопасности объекта доминирует над меткой субъекта. Надежная вычислительная база должна контролировать идентификационную и аутентификационную информацию. При создании новых субъектов (например, процессов) их метки безопасности не должны доминировать над меткой породившего их пользователя.

- Класс B2 - в дополнение к B1, все ресурсы системы (в том числе ПЗУ, устройства ввода/вывода) должны иметь метки безопасности и служить объектами принудительного управления доступом.

2.1.6.2. Требования к подотчетности

Идентификация и аутентификация:

- Класс C1 - пользователи должны идентифицировать себя, прежде чем выполнять какие-либо иные действия, контролируемые надежной вычислительной базой. Для аутентификации должен использоваться какой-либо защитный механизм, например, пароли. Аутентификационная информация должна быть защищена от несанкционированного доступа.
- Класс C2 - в дополнение к C1, каждый пользователь системы должен уникальным образом идентифицироваться. Каждое регистрируемое действие должно ассоциироваться с конкретным пользователем.
- Класс B1 - в дополнение к C2, надежная вычислительная база должна поддерживать метки безопасности пользователей.

Предоставление надежного пути:

- Класс B2 - надежная вычислительная база должна поддерживать надежный коммуникационный путь к себе для пользователя, выполняющего операции начальной идентификации и аутентификации. Инициатива в общении по этому пути должна исходить исключительно от пользователя.
- Класс B3 - в дополнение к B2, надежный коммуникационный путь может формироваться по запросу, исходящему как от пользователя, так и от самой базы. Надежный путь может использоваться для начальной идентификации и аутентификации, для изменения текущей метки безопасности пользователя и т.п. Общение по надежному пути должно быть логически отделено и изолировано от других информационных потоков.

Аудит:

- Класс C2 - надежная вычислительная база должна создавать, поддерживать и защищать журнал регистрационной информации, относящейся к доступу к объектам, контролируемым базой. Должна быть возможность регистрации следующих событий:
  - использование механизма идентификации и аутентификации,
  - внесение объектов в адресное пространство пользователя (например, открытие файла, запуск программы),
  - удаление объектов,
  - действия системных операторов, системных администраторов, администраторов безопасности,
  - другие события, затрагивающие информационную безопасность.
  - Каждая регистрационная запись должна включать следующие поля:
  - дата и время события,
  - идентификатор пользователя,
  - тип события,
  - результат действия (успех или неудача).

Для событий идентификации и аутентификации регистрируется также идентификатор устройства (например, терминала). Для действий с объектами регистрируются имена объектов. Системный администратор может выбирать набор регистрируемых событий для каждого пользователя.

- Класс B1 - в дополнение к C2, должны регистрироваться операции выдачи на печать и ассоциированные внешние представления меток безопасности. При операциях с объектами, помимо имен, регистрируются их метки безопасности. Набор регистрируемых событий может различаться в зависимости от уровня секретности объектов.
- Класс B2 - в дополнение к B1, должна быть возможность регистрировать события, связанные с организацией тайных каналов с памятью.
- Класс B3 - в дополнение к B2, должна быть возможность регистрации появления или накопления событий, несущих угрозу политике безопасности системы.

Администратор безопасности должен немедленно извещаться о попытках нарушения политики безопасности. А система, в случае продолжения попыток, должна пресекать их наименее болезненным способом.

2.1.6.3. Требования к гарантированности

Операционная гарантированность:

Архитектура системы:

- Класс C1 - надежная вычислительная база должна поддерживать область для собственного выполнения, защищенную от внешних воздействий (в частности, от изменения команд и/или данных) и от попыток слежения за ходом работы. Ресурсы, контролируемые базой, могут составлять определенное подмножество всех субъектов и объектов системы.

- Класс C2 - в дополнение к C1, надежная вычислительная база должна изолировать защищаемые ресурсы в той мере, как это диктуется требованиями контроля доступа и подотчетности.
- Класс B1 - в дополнение к C2, надежная вычислительная база должна обеспечивать взаимную изоляцию процессов путем разделения их адресных пространств.
- Класс B2 - в дополнение к B1, надежная вычислительная база должна быть внутренне структурирована на хорошо определенные, относительно независимые модули. Надежная вычислительная база должна эффективно использовать имеющееся оборудование для отделения элементов, критически важных с точки зрения защиты, от прочих компонентов системы. Модули базы должны проектироваться с учетом принципа минимизации привилегий. Для защиты логически раздельных хранимых объектов должны использоваться аппаратные средства, такие как сегментация. Должен быть полностью определен пользовательский интерфейс к надежной вычислительной базе и все элементы базы.
- Класс B3 - в дополнение к B2, надежная вычислительная база должна быть спроектирована и структурирована таким образом, чтобы использовать полный и концептуально простой защитный механизм с точно определенной семантикой. Этот механизм должен играть центральную роль во внутренней структуризации надежной вычислительной базы и всей системы. База должна активно использовать разделение по уровням, абстракцию и инкапсуляцию данных. Значительные инженерные усилия должны быть направлены на уменьшение сложности надежной вычислительной базы и на вынесение из нее модулей, не являющихся критически важными с точки зрения защиты.

Целостность системы:

- Класс C1 - должны быть в наличии аппаратные и/или программные средства, позволяющие периодически проверять корректность функционирования аппаратных и микропрограммных компонентов надежной вычислительной базы.

Анализ тайных каналов передачи информации:

- Класс B2 - системный архитектор должен тщательно проанализировать возможности по организации тайных каналов с памятью и оценить максимальную пропускную способность каждого выявленного канала.
- Класс B3 - в дополнение к B2, аналогичная процедура должна быть проделана для временных каналов.
- Класс A1 - в дополнение к B3, для анализа должны использоваться формальные методы.

Надежное администрирование:

- Класс B2 - система должна поддерживать разделение функций оператора и администратора.
- Класс B3 - в дополнение к B2, должна быть специфицирована роль администратора безопасности. Получить права администратора безопасности можно только после

выполнения явных, протоколируемых действий. Не относящиеся к защите действия администратора безопасности должны быть по возможности ограничены.

Надежное восстановление:

- Класс B3 - должны существовать процедуры и/или механизмы, позволяющие произвести восстановление после сбоя или иного нарушения работы без ослабления защиты.

Технологическая гарантированность:

Тестирование:

- Класс C1 - защитные механизмы должны быть протестированы на предмет соответствия их поведения системной документации. Тестирование должно подтвердить, что у неавторизованного пользователя нет очевидных способов обойти или разрушить средства защиты надежной вычислительной базы.
- Класс C2 - в дополнение к C1, тестирование должно подтвердить отсутствие очевидных недостатков в механизмах изоляции ресурсов и защиты регистрационной информации.
- Класс B1 - в дополнение к C2, группа специалистов, полностью понимающих конкретную реализацию надежной вычислительной базы, должна подвергнуть описание архитектуры, исходные и объектные коды тщательному анализу и тестированию. Цель должна состоять в выявлении всех дефектов архитектуры и реализации, позволяющих субъекту без должной авторизации читать, изменять, удалять информацию или приводить базу в состояние, когда она перестает обслуживать запросы других субъектов. Все выявленные недостатки должны быть исправлены или нейтрализованы, после чего база подвергается повторному тестированию, чтобы убедиться в отсутствии старых или новых недостатков.
- Класс B2 - в дополнение к B1, должна быть продемонстрирована относительная устойчивость надежной вычислительной базы к попыткам проникновения.
- Класс B3 - в дополнение к B2, должна быть продемонстрирована устойчивость надежной вычислительной базы к попыткам проникновения. Не должно быть выявлено архитектурных недостатков. Допускается выявление лишь небольшого числа исправимых недостатков реализации. Должна существовать обоснованная уверенность, что немногие недостатки остались невыявленными.
- Класс A1 - в дополнение к B3, тестирование должно продемонстрировать, что реализация надежной вычислительной базы соответствует формальным спецификациям верхнего уровня.

Основу тестирования средств защиты от проникновения в систему должно составлять ручное или иное отображение спецификаций на исходные тексты.

Верификация спецификаций архитектуры:

- Класс B1 - должна существовать неформальная или формальная модель политики безопасности, поддерживаемой надежной вычислительной базой. Модель должна

соответствовать основным посылкам политики безопасности на протяжении всего жизненного цикла системы.

- Класс B2 - в дополнение к B1, модель политики безопасности должна быть формальной. Для надежной вычислительной базы должны существовать описательные спецификации верхнего уровня, точно и полно определяющие ее интерфейс.
- Класс B3 - в дополнение к B2, должны быть приведены убедительные аргументы соответствия между спецификациями и моделью.
- Класс A1 - в дополнение к B3, помимо описательных, должны быть представлены формальные спецификации верхнего уровня, относящиеся к аппаратным и/или микропрограммным элементам, составляющим интерфейс надежной вычислительной базы. Комбинация формальных и неформальных методов должна подтвердить соответствие между спецификациями и моделью. Должны использоваться современные методы формальной спецификации и верификации систем, доступные Национальному центру компьютерной безопасности США. Ручное или иное отображение формальных спецификаций на исходные тексты должно подтвердить корректность реализации надежной вычислительной базы.

Конфигурационное управление:

- Класс B2 - в процессе разработки и сопровождения надежной вычислительной базы должна использоваться система конфигурационного управления, обеспечивающая контроль изменений в описательных спецификациях верхнего уровня, иных архитектурных данных, реализационной документации, исходных текстах, работающей версии объектного кода, тестовых данных и документации. Конфигурационное управление должно обеспечивать соответствие друг другу всех аспектов текущей версии надежной вычислительной базы. Должны предоставляться средства генерации новых версий базы по исходным текстам и средства для сравнения версий, чтобы убедиться в том, что произведены только запланированные изменения.
- Класс A1 - в дополнение к B2, механизм конфигурационного управления должен распространяться на весь жизненный цикл и все компоненты системы, имеющие отношение к обеспечению безопасности, включая спецификации и документацию. Для защиты эталонной копии материалов, использующихся для генерации надежной вычислительной базы, должна использоваться комбинация физических, административных и технических мер.

Надежное распространение:

- Класс A1 - должна поддерживаться целостность соответствия между эталонными данными, описывающими текущую версию вычислительной базы, и эталонной копией текстов этой версии. Должны существовать процедуры, подтверждающие соответствие между поставляемыми клиентам аппаратными и программными компонентами и эталонной копией.

2.1.6.4. Требования к документации

Руководство пользователя по средствам безопасности:

- Класс C1 - отдельный фрагмент документации (глава, том) должен описывать защитные механизмы, предоставляемые надежной вычислительной базой, и их взаимодействие между собой, содержать рекомендации по их использованию.

Руководство администратора по средствам безопасности:

- Класс C1 - руководство должно содержать сведения о функциях и привилегиях, которыми управляет системный администратор посредством механизмов безопасности.
- Класс C2 - в дополнение к C1, должны описываться процедуры обработки регистрационной информации и управления файлами с такой информацией, а также структура записей для каждого типа регистрируемых событий.
- Класс B1 - в дополнение к C2, руководство должно описывать функции оператора и администратора, затрагивающие безопасность, в том числе действия по изменению характеристик пользователей. Должны быть представлены рекомендации по согласованному и эффективному использованию средств безопасности, их взаимодействию друг с другом, по безопасной генерации новых версий надежной вычислительной базы.
- Класс B2 - в дополнение к B1, должны быть указаны модули надежной вычислительной базы, содержащие механизмы проверки обращений. Должна быть описана процедура безопасной генерации новой версии базы после внесения изменений в исходные тексты.
- Класс B3 - в дополнение к B2, должна быть описана процедура, обеспечивающая безопасность начального запуска системы и возобновления ее работы после сбоя.

Тестовая документация:

- Класс C1 - разработчик системы должен представить экспертному совету документ, содержащий план тестов, процедуры прогона тестов и результаты тестов.
- Класс B2 - в дополнение к C1, тесты должны подтверждать действенность мер по уменьшению пропускной способности тайных каналов передачи информации.
- Класс A1 - в дополнение к B2, должно быть описано соответствие между формальными спецификациями верхнего уровня и исходными текстами.

Описание архитектуры:

- Класс C1 - должны быть описаны подход к безопасности, используемый производителем, и применение этого подхода при реализации надежной вычислительной базы. Если база состоит из нескольких модулей, должен быть описан интерфейс между ними.
- Класс B1 - в дополнение к C1, должно быть представлено неформальное или формальное описание модели политики безопасности, проводимой в жизнь надежной вычислительной базой. Необходимо наличие аргументов в пользу

достаточности избранной модели для реализации политики безопасности. Должны быть описаны защитные механизмы базы и их место в модели.

- Класс B2 - в дополнение к B1, модель политики безопасности должна быть формальной и доказательной. Должно быть показано, что описательные спецификации верхнего уровня точно отражают интерфейс надежной вычислительной базы. Должно быть показано, как база реализует концепцию монитора обращений, почему она устойчива к попыткам отслеживания ее работы, почему ее нельзя обойти. Должна быть доказана также корректность реализации базы. Должна быть описана структура базы, чтобы облегчить ее тестирование и проверку соблюдения принципа минимизации привилегий. Документация должна содержать результаты анализа тайных каналов передачи информации и описание мер протоколирования, помогающих выявлять каналы с памятью.
- Класс B3 - в дополнение к B2, должно быть неформально продемонстрировано соответствие между описательными спецификациями верхнего уровня и реализацией надежной вычислительной базы.
- Класс A1 - в дополнение к B3, должно быть неформально продемонстрировано соответствие между формальными спецификациями верхнего уровня и реализацией надежной вычислительной базы.

2.1.7. Некоторые комментарии

Поучительно проследить, как именно требования к политике безопасности и к гарантированности распределены по классам безопасности. В "младших" классах политика довольно быстро ужесточается, достигая пика к классу B1. Напротив, меры гарантированности отнесены в основном в "старшие" классы, начиная с B2. Это подтверждает независимость двух основных групп критериев надежности и методологическую целесообразность их разделения по Европейскому образцу (см. следующий раздел).

Распределение требований по классам вызывает ряд конкретных возражений. Неоправданно далеко отодвинуты такие очевидные требования, как извещение о нарушении защиты, конфигурационное управление, безопасный запуск и восстановление после сбоев. Возможно, это оправдано в физически защищенной военной среде, но никак не в коммерческой, когда постоянное слежение за перемещениями сотрудников может быть очень дорогим удовольствием.

В представленном виде "Критерии" полностью игнорируют коммуникационный аспект, присущий современным распределенным системам. Далее мы покажем, сколь специфична эта область, сколько потенциальных угроз безопасности она содержит, какие новые защитные механизмы следует использовать. Примечательно, что изданные позднее толкования "Критериев" для сетевых конфигураций примерно в три раза толще самой "Оранжевой книги".

Очень важный методологический недостаток "Оранжевой книги" - явная ориентация на производителя и оценщика, а не на покупателя систем. Она не дает ответ на вопрос, как безопасным образом строить систему, как наращивать отдельные компоненты и конфигурацию в целом. "Критерии" рассчитаны на статичные, замкнутые системы, которые,

вероятно, доминируют в военной среде, но крайне редки в среде коммерческой. Покупателям нужны более динамичные и структурированные критерии.

Тем не менее, следует подчеркнуть, что публикация "Оранжевой книги" без всякого преувеличения стала эпохальным событием в области защиты коммерческих информационных систем. Появился общепризнанный понятийный базис, без которого даже обсуждение проблем безопасности было бы затруднительным. Именно в этом видится главная ценность «Критериев оценки надежных компьютерных систем» Министерства обороны США.

Отметим, что огромный идейный потенциал "Оранжевой книги" пока во многом остается невостребованным. Прежде всего, это касается концепции технологической гарантированности, охватывающей весь жизненный цикл системы - от выработки спецификаций до фазы эксплуатации. При современной технологии программирования результирующая система не содержит информации, присутствующей в исходных спецификациях. В то же время, наличие подобной информации (быть может, в предварительно обработанном виде) на этапе выполнения, позволило бы по-новому поставить и решить многие проблемы информационной безопасности. Например, знание монитором обращений того, к каким объектам (или классам объектов) может осуществлять доступ программа, существенно затруднило бы создание "троянских коней" и распространение вирусов. К сожалению, пока для принятия решения о допустимости того или иного действия используется скудная и, в основном, косвенная информация (как правило - идентификатор владельца процесса), не имеющая отношения к семантике действия.

2.2. Гармонизированные критерии европейских стран

Следуя по пути интеграции, Европейские страны приняли согласованные критерии оценки безопасности информационных технологий (Information Technology Security Evaluation Criteria, ITSEC). Данное изложение основывается на версии 1.2 этих Критериев, опубликованной в июне 1991 года от имени соответствующих органов четырех стран - Франции, Германии, Нидерландов и Великобритании [7]. Выгода от использования согласованных критериев очевидна для всех - и для производителей, и для потребителей, и для самих органов сертификации.

Принципиально важной чертой Европейских Критериев является отсутствие априорных требований к условиям, в которых должна работать информационная система. (Напомним, что в "Критериях" Министерства обороны США очевидна привязка к условиям правительственной системы, обрабатывающей секретную информацию.) Так называемый спонсор, то есть организация, запрашивающая сертификационные услуги, формулирует цель оценки, то есть описывает условия, в которых должна работать система, возможные угрозы ее безопасности и предоставляемые ею защитные функции. Задача органа сертификации - оценить, насколько полно достигаются поставленные цели, то есть насколько корректны и эффективны архитектура и реализация механизмов безопасности в описанных спонсором условиях. Таким образом, в терминологии "Оранжевой книги", Европейские Критерии относятся к гарантированности безопасной работы системы. Требования к политике безопасности и к наличию защитных механизмов не являются составной частью Критериев. Впрочем, чтобы облегчить формулировку цели оценки, Критерии содержат в качестве

приложения описание десяти примерных классов функциональности, типичных для правительственных и коммерческих систем.

Перейдем к систематическому изложению подхода, развиваемого в Европейских Критериях.

2.2.1. Основные понятия

Европейские Критерии рассматривают следующие составляющие информационной безопасности:

- конфиденциальность, то есть защиту от несанкционированного получения информации;
- целостность, то есть защиту от несанкционированного изменения информации;
- доступность, то есть защиту от несанкционированного удержания информации и ресурсов.

В Критериях проводится различие между системами и продуктами. Система - это конкретная аппаратно-программная конфигурация, построенная с вполне определенными целями и функционирующая в известном окружении. Продукт - это аппаратно-программный "пакет", который можно купить и по своему усмотрению встроить в ту или иную систему. Таким образом, с точки зрения информационной безопасности основное отличие между системой и продуктом состоит в том, что система имеет конкретное окружение, которое можно определить и изучить сколь угодно детально, а продукт должен быть рассчитан на использование в различных условиях. Угрозы безопасности системы носят вполне конкретный и реальный характер. Относительно угроз продукту можно лишь строить предположения. Разработчик может специфицировать условия, пригодные для функционирования продукта; дело покупателя обеспечить выполнение этих условий.

Из практических соображений важно обеспечить единство критериев оценки продуктов и систем - например, чтобы облегчить и удешевить оценку системы, составленной из ранее сертифицированных продуктов. В этой связи для систем и продуктов вводится единый термин - объект оценки. В соответствующих местах делаются оговорки, какие требования относятся исключительно к системам, а какие - только к продуктам.

Каждая система и/или продукт предъявляет свои требования к обеспечению конфиденциальности, целостности и доступности. Чтобы удовлетворить эти требования, необходимо предоставить соответствующий набор функций (сервисов) безопасности, таких как идентификация и аутентификация, управление доступом или восстановление после сбоев.

Сервисы безопасности реализуются посредством конкретных механизмов. Например, для реализации функции идентификации и аутентификации можно использовать такой механизм, как сервер аутентификации Kerberos.

Чтобы объект оценки можно было признать надежным, необходима определенная степень уверенности в наборе функций и механизмов безопасности. Степень уверенности мы будем называть гарантированностью. Гарантированность может быть большей или меньшей в зависимости от тщательности проведения оценки.

Гарантированность затрагивает два аспекта - эффективность и корректность средств безопасности. При проверке эффективности анализируется соответствие между целями,

сформулированными для объекта оценки, и имеющимся набором функций безопасности. Точнее говоря, рассматриваются вопросы адекватности функциональности, взаимной согласованности функций, простоты их использования, а также возможные последствия эксплуатации известных слабых мест защиты. Кроме того, в понятие эффективности входит способность механизмов защиты противостоять прямым атакам (мощность механизма). Определяется три градации мощности - базовая, средняя и высокая.

Под корректностью понимается правильность реализации функций и механизмов безопасности. В Критериях определяется семь возможных уровней гарантированности корректности - от E0 до E6 (в порядке возрастания). Уровень E0 обозначает отсутствие гарантированности (аналог уровня D "Оранжевой книги"). При проверке корректности анализируется весь жизненный цикл объекта оценки - от проектирования до эксплуатации и сопровождения.

Общая оценка системы складывается из минимальной мощности механизмов безопасности и уровня гарантированности корректности. Теоретически эти два аспекта независимы, хотя на практике нет смысла проверять правильность реализации "по высшему разряду", если механизмы безопасности не обладают даже средней мощностью.

2.2.2. Функциональность

В Европейских Критериях средства, имеющие отношение к информационной безопасности, рассматриваются на трех уровнях детализации. Наиболее абстрактный взгляд касается лишь целей безопасности. На этом уровне мы получаем ответ на вопрос, зачем нужны функции безопасности. Второй уровень содержит спецификации функций безопасности. Мы узнаем, какая функциональность на самом деле обеспечивается. Наконец, на третьем уровне содержится информация о механизмах безопасности. Мы видим, как реализуется декларированная функциональность.

Спецификации функций безопасности - важнейшая часть описания объекта оценки. Критерии рекомендуют выделить в этих спецификациях разделы со следующими заголовками:

- Идентификация и аутентификация.
- Управление доступом.
- Подотчетность.
- Аудит.
- Повторное использование объектов.
- Точность информации.
- Надежность обслуживания.
- Обмен данными.

Большинство из перечисленных тем мы рассматривали при анализе "Оранжевой книги". Здесь мы остановимся лишь на моментах, специфичных для Европейских Критериев.

Под идентификацией и аутентификацией понимается не только проверка подлинности пользователей в узком смысле, но и функции для регистрации новых пользователей и удаления старых, а также функции для генерации, изменения и проверки аутентификационной информации, в том числе средства контроля целостности. Сюда же относятся функции для ограничения числа повторных попыток аутентификации.

Средства управления доступом также трактуются Европейскими Критериями достаточно широко. В этот раздел попадают, помимо прочих, функции, обеспечивающие временное ограничение доступа к совместно используемым объектам с целью поддержания целостности этих объектов - мера, типичная для систем управления базами данных. В этот же раздел попадают функции для управления распространением прав доступа и для контроля получения информации путем логического вывода и агрегирования данных (что также типично для СУБД).

Под точностью в Критериях понимается поддержание определенного соответствия между различными частями данных (точность связей) и обеспечение неизменности данных при передаче между процессами (точность коммуникаций). Точность выступает как один из аспектов целостности информации.

Функции надежности обслуживания должны гарантировать, что действия, критичные по времени, будут выполнены ровно тогда, когда нужно - не раньше и не позже, и что некритичные действия нельзя перевести в разряд критичных. Далее, должна быть гарантия, что авторизованные пользователи за разумное время получат запрашиваемые ресурсы. Сюда же относятся функции для обнаружения и нейтрализации ошибок, необходимые для минимизации простоев, а также функции планирования, позволяющие гарантировать время реакции на внешние события.

К области обмена данными относятся функции, обеспечивающие коммуникационную безопасность, то есть безопасность данных, передаваемых по каналам связи. Здесь Европейские Критерии следуют в фарватере рекомендаций X.800, предлагая следующие подзаголовки:

- Аутентификация.
- Управление доступом.
- Конфиденциальность данных.
- Целостность данных.
- Невозможность отказаться от совершенных действий.

Перечисленные темы мы рассмотрим позднее, в разделе, посвященном особенностям информационной безопасности компьютерных сетей.

Набор функций безопасности может специфицироваться с использованием ссылок на заранее определенные классы функциональности. В Европейских Критериях таких классов десять. Пять из них (F-C1, F-C2, F-B1, F-B2, F-B3) соответствуют классам безопасности "Оранжевой книги".

Класс F-IN предназначается для объектов оценки с высокими потребностями по обеспечению целостности данных и программ, что типично для систем управления базами данных. При описании класса F-IN вводится понятие роли, выдвигается требование по предоставлению доступа к определенным объектам только с помощью предопределенных процессов. Должны различаться следующие виды доступа: чтение, запись, добавление, удаление, переименование (для всех объектов), выполнение, удаление, переименование (для выполняемых объектов), создание и удаление объектов.

Класс F-AV характеризуется повышенными требованиями к доступности. Это существенно, например, для систем управления технологическими процессами. В разделе "Надежность обслуживания" описания этого класса специфицируется, что объект оценки должен восстанавливаться после отказа отдельного аппаратного компонента таким образом,

чтобы все критически важные функции оставались постоянно доступными. То же должно быть верно для вставки отремонтированного компонента, причем после этого объект оценки возвращается в состояние, устойчивое к одиночным отказам. Независимо от уровня загрузки должно гарантироваться время реакции на определенные события и отсутствие тупиков.

Класс F-DI характеризуется повышенными требованиями к целостности передаваемых данных. Перед началом общения стороны должны быть в состоянии проверить подлинность друг друга. При получении данных должна предоставляться возможность проверки подлинности источника. При обмене данными должны предоставляться средства контроля ошибок и их исправления. В частности, должны обнаруживаться все повреждения или намеренные искажения адресной и пользовательской информации. Знание алгоритма обнаружения искажений не должно давать возможность производить нелегальную модификацию. Должны обнаруживаться и трактоваться как ошибки попытки воспроизведения ранее переданных сообщений.

Класс FDC характеризуется повышенными требованиями к конфиденциальности передаваемой информации. Перед поступлением данных в каналы связи должно автоматически выполняться шифрование с использованием сертифицированных средств. На приемном конце также автоматически производится расшифровка. Ключи шифрования должны быть защищены от несанкционированного доступа.

Класс F-DX характеризуется повышенными требованиями и к целостности, и к конфиденциальности информации. Его можно рассматривать как объединение классов F-DI и F-DC с дополнительными возможностями шифрования, действующими из конца в конец, и с защитой от анализа трафика по определенным каналам. Должен быть ограничен доступ к ранее переданной информации, которая в принципе может способствовать нелегальной расшифровке.

2.2.3. Гарантированность эффективности

Для получения гарантий эффективности средств безопасности рассматриваются следующие вопросы:

- Соответствие набора функций безопасности провозглашенным целям, то есть их пригодность для противодействия угрозам, перечисленным в описании объекта оценки;
- Взаимная согласованность различных функций и механизмов безопасности;
- Способность механизмов безопасности противостоять прямым атакам;
- Возможность практического использования слабостей в архитектуре объекта оценки, то есть наличие способов отключения, обхода, повреждения и обмана функций безопасности;
- Возможность небезопасного конфигурирования или использования объекта оценки при условии, что администраторы и/или пользователи имеют основание считать ситуацию безопасной;
- Возможность практического использования слабостей в функционировании объекта оценки.

Важнейшей частью проверки эффективности является анализ слабых мест в защите объекта оценки. Цель анализа - найти все возможности отключения, обхода, повреждения, обмана средств защиты. Оценивается также способность всех критически важных защитных механизмов противостоять прямым атакам –мощность механизмов. Защищенность системы или продукта не может быть выше мощности самого слабого из критически важных механизмов, поэтому в Критериях имеется в виду минимальная гарантированная мощность.

Для нее определены три уровня - базовый, средний и высокий.

Согласно Критериям, мощность можно считать базовой, если механизм способен противостоять отдельным случайным атакам.

Мощность можно считать средней, если механизм способен противостоять злоумышленникам с ограниченными ресурсами и возможностями.

Наконец, мощность можно считать высокой, если есть уверенность, что механизм может быть побежден только злоумышленником с высокой квалификацией, набор возможностей и ресурсов которого выходит за пределы практичности.

Важной характеристикой является простота использования продукта или системы. Должны существовать средства, информирующие персонал о переходе объекта в небезопасное состояние (что может случиться в результате сбоя, ошибок администратора или пользователя). Ситуации, когда в процессе функционирования объекта оценки появляются слабости, допускающие практическое использование, в то время как администратор об этом не знает, должны быть исключены.

Эффективность защиты признается неудовлетворительной, если выявляются слабые места, допускающие практическое использование, и эти слабости не исправляются до окончания процесса оценки. В таком случае объекту присваивается уровень гарантированности E0.

Обратим внимание на то, что анализ слабых мест производится в контексте целей, декларируемых для объекта оценки. Например, можно примириться с наличием тайных каналов передачи информации, если отсутствуют требования к конфиденциальности. Далее, слабость конкретного защитного механизма может не иметь значения, если она компенсируется другими средствами обеспечения безопасности, то есть если механизм не является критически важным.

2.2.4. Гарантированность корректности

При проверке корректности объекта оценки применяются две группы критериев. Первая группа относится к конструированию и разработке системы или продукта, вторая - к эксплуатации. Оцениваются следующие аспекты:

Процесс разработки:

- требования к объекту оценки;
- общая архитектура;
- детализированная архитектура;
- реализация.

Среда разработки:

- средства конфигурационного управления;

- используемые языки программирования и компиляторы;
- безопасность среды разработки (ее физическая защищенность, методы подбора персонала и т.п.).

Эксплуатационная документация:

- руководство пользователя;
- руководство администратора.

Операционное окружение:

- доставка и конфигурирование системы или продукта;
- запуск и эксплуатация.

Уровни корректности от E1 до E6 выстроены по нарастанию требований к тщательности оценки. Так, на уровне E1 анализируется лишь общая архитектура объекта - вся остальная уверенность может быть следствием функционального тестирования. На уровне E3 к анализу привлекаются исходные тексты программ и схемы аппаратуры. На уровне E6 требуется формальное описание функций безопасности, общей архитектуры, а также модели политики безопасности. В общем распределение требований по уровням гарантированности в Европейских Критериях соответствует аналогичному распределению для классов безопасности C1 - A1 из "Оранжевой книги".

Мы не будем останавливаться на детальном описании уровней корректности.

2.3. Руководящие документы по защите от НСД Гостехкомиссии при Президенте РФ

В 1992 году Гостехкомиссия при Президенте РФ опубликовала пять Руководящих документов [1-5], посвященных проблеме защиты от несанкционированного доступа (НСД) к информации. Мы рассмотрим важнейшие из них.

2.3.1. Концепция защиты от несанкционированного доступа к информации

Идейной основой набора Руководящих документов является «Концепция защиты СВТ и АС от НСД к информации». Концепция «излагает систему взглядов, основных принципов, которые закладываются в основу проблемы защиты информации от несанкционированного доступа (НСД), являющейся частью общей проблемы безопасности информации».

В Концепции различаются понятия средств вычислительной техники (СВТ) и автоматизированной системы (АС), аналогично тому, как в Европейских Критериях проводится деление на продукты и системы. Более точно, «Концепция предусматривает существование двух относительно самостоятельных и, следовательно, имеющих отличие направлений в проблеме защиты информации от НСД. Это - направление, связанное с СВТ, и направление, связанное с АС.

Отличие двух направлений порождено тем, что СВТ разрабатываются и поставляются на рынок лишь как элементы, из которых в дальнейшем строятся функционально ориентированные АС, и поэтому, не решая прикладных задач, СВТ не содержат пользовательской информации.

Помимо пользовательской информации при создании АС появляются такие отсутствующие при разработке СВТ характеристики АС, как полномочия пользователей, модель нарушителя, технология обработки информации».

Существуют различные способы покушения на информационную безопасность - радиотехнические, акустические, программные и т.п. Среди них НСД выделяется как доступ к информации, нарушающий установленные правила разграничения доступа, с использованием штатных средств, предоставляемых СВТ или АС

Под штатными средствами понимается совокупность программного, микропрограммного и технического обеспечения СВТ или АС.

В Концепции формулируются следующие основные принципы защиты от НСД к информации:

- Защита СВТ обеспечивается комплексом программно-технических средств.
- Защита АС обеспечивается комплексом программно-технических средств и поддерживающих их организационных мер.
- Защита АС должна обеспечиваться на всех технологических этапах обработки информации и во всех режимах функционирования, в том числе при проведении ремонтных и регламентных работ.

Программно-технические средства защиты не должны существенно ухудшать основные функциональные характеристики АС (надежность, быстродействие, возможность изменения конфигурации АС).

Неотъемлемой частью работ по защите является оценка эффективности средств защиты, осуществляемая по методике, учитывающей всю совокупность технических характеристик оцениваемого объекта, включая технические решения и практическую реализацию средств защиты.

Защита АС должна предусматривать контроль эффективности средств защиты от НСД. Этот контроль может быть либо периодическим, либо инициироваться по мере необходимости пользователем АС или контролирующими органами.

Концепция ориентируется на физически защищенную среду, проникновение в которую посторонних лиц считается невозможным, поэтому нарушитель определяется как субъект, имеющий доступ к работе со штатными средствами АС и СВТ как части АС. Нарушители классифицируются по уровню возможностей, предоставляемых им штатными средствами АС и СВТ. Выделяется четыре уровня этих возможностей. Классификация является иерархической, т.е. каждый следующий уровень включает в себя функциональные возможности предыдущего:

- Первый уровень определяет самый низкий уровень возможностей ведения диалога в АС - запуск задач (программ) из фиксированного набора, реализующих заранее предусмотренные функции по обработке информации.
- Второй уровень определяется возможностью создания и запуска собственных программ с новыми функциями по обработке информации.
- Третий уровень определяется возможностью управления функционированием АС, т.е. воздействием на базовое программное обеспечение системы и на состав и конфигурацию ее оборудования.
- Четвертый уровень определяется всем объемом возможностей лиц, осуществляющих проектирование, реализацию и ремонт технических средств АС,

вплоть до включения в состав СВТ собственных технических средств с новыми функциями по обработке информации.

В своем уровне нарушитель является специалистом высшей квалификации, знает все о АС и, в частности, о системе и средствах ее защиты.

В качестве главного средства защиты от НСД к информации в Концепции рассматривается система разграничения доступа (СРД) субъектов к объектам доступа. Основными функциями СРД являются:

- реализация правил разграничения доступа (ПРД) субъектов и их процессов к данным;
- реализация ПРД субъектов и их процессов к устройствам создания твердых копий;
- изоляция программ процесса, выполняемого в интересах субъекта, от других субъектов;
- управление потоками данных с целью предотвращения записи данных на носители несоответствующего грифа;
- реализация правил обмена данными между субъектами для АС и СВТ, построенных по сетевым принципам.

Кроме того, Концепция предусматривает наличие обеспечивающих средств для СРД, которые выполняют следующие функции:

- идентификацию и опознание (аутентификацию) субъектов и поддержание привязки субъекта к процессу, выполняемому для субъекта;
- регистрацию действий субъекта и его процесса;
- предоставление возможностей исключения и включения новых субъектов и объектов доступа, а также изменение полномочий субъектов;
- реакцию на попытки НСД, например, сигнализацию, блокировку, восстановление после НСД;
- тестирование;
- очистку оперативной памяти и рабочих областей на магнитных носителях после завершения работы пользователя с защищаемыми данными;
- учет выходных печатных и графических форм и твердых копий в АС;
- контроль целостности программной и информационной части как СРД, так и обеспечивающих ее средств.

Мы видим, что функции системы разграничения доступа и обеспечивающих средств, предлагаемые в Концепции близки к аналогичным положениям "Оранжевой книги". Это вполне естественно, поскольку близки и исходные посылки - защита от несанкционированного доступа к информации в условиях физически безопасного окружения.

Технические средства защиты от НСД, согласно Концепции, должны оцениваться по следующим основным параметрам:

- степень полноты охвата ПРД реализованной СРД и ее качество;
- состав и качество обеспечивающих средств для СРД;

- гарантии правильности функционирования СРД и обеспечивающих ее средств.

Здесь усматривается аналогия с гарантированностью эффективности и корректности в Европейских гармонизированных Критериях, что можно только приветствовать.

2.3.2. Классификация СВТ по уровню защищенности от НСД

Таблица 2.1.

Наименование показателя	Класс защищенности					
	6	5	4	3	2	1
1. Произвольный контроль доступа	+	+	+	=	+	=
2. Принудительный контроль доступа	-	-	+	=	=	=
3. Очистка памяти после использования	-	+	+	+	=	=
4. Изоляция модулей системы	-	-	+	=	+	=
5. Маркировка документов	-	-	+	=	=	=
6. Защита ввода и вывода на сменный носитель	-	-	+	=	=	=
7. Сопоставление пользователя с устройством	-	-	+	=	=	=
8. Идентификация и аутентификация	+	=	+	=	=	=
9. Гарантия проектирования	-	+	+	+	+	+
10. Регистрация	-	+	+	+	=	=
11. Взаимодействие пользователя с КСЗ	-	-	-	+	=	=
12. Надежное восстановление	-	-	-	+	=	=
13. Целостность КСЗ	-	+	+	+	=	=
14. Контроль модификации	-	-	-	-	+	=
15. Контроль дистрибуции	-	-	-	-	+	=
16. Гарантии архитектуры	-	-	-	-	-	+
17. Тестирование	+	+	+	+	+	=
18. Руководство пользователя	+	=	=	=	=	=
19. Руководство по КСЗ	+	+	=	+	+	=
20. Текстовая документация	+	+	+	+	+	=
21. Проектная документация	+	+	+	+	+	+

Обозначения:
“-” – нет требований к данному классу
“+” – новые или дополнительные требования
“=” – требования к классу совпадают с требованиями предыдущего класса
“КСЗ” – Комплекс Средств Защиты

Переходя к рассмотрению предлагаемой Гостехкомиссией при Президенте РФ классификации средств вычислительной техники по уровню защищенности от несанкционированного доступа к информации, отметим ее близость к классификации "Оранжевой книги". Процитируем соответствующий Руководящий документ:

«Устанавливается семь классов защищенности СВТ от НСД к информации. Самый низкий класс - седьмой, самый высокий - первый.

Классы подразделяются на четыре группы, отличающиеся качественным уровнем защиты:

- первая группа содержит только один седьмой класс;
- вторая группа характеризуется дискреционной защитой и содержит шестой и пятый классы;

- третья группа характеризуется мандатной защитой и содержит четвертый, третий и второй классы;
- четвертая группа характеризуется верифицированной защитой и содержит только первый класс».

Седьмой класс присваивают СВТ, к которым предъявлялись требования по защите от НСД к информации, но при оценке защищенность СВТ оказалась ниже уровня требований шестого класса.

Приведем сводную таблицу распределения показателей защищенности по шести классам СВТ (табл. 1).

2.3.3. Классификация АС по уровню защищенности от НСД

Классификация автоматизированных систем устроена иначе. Снова обратимся к соответствующему Руководящему документу:

"1.8. Устанавливается девять классов защищенности АС от НСД к информации.

Каждый класс характеризуется определенной минимальной совокупностью требований по защите. Классы подразделяются на три группы, отличающиеся особенностями обработки информации в АС. В пределах каждой группы соблюдается иерархия требований по защите в зависимости от ценности (конфиденциальности) информации и, следовательно, иерархия классов защищенности АС.

1.9. Третья группа классифицирует АС, в которых работает один пользователь, допущенный ко всей информации АС, размещенной на носителях одного уровня конфиденциальности. Группа содержит два класса - 3Б и 3А.

Вторая группа классифицирует АС, в которых пользователи имеют одинаковые права доступа (полномочия) ко всей информации АС, обрабатываемой и (или) хранимой на носителях различного уровня конфиденциальности.

Группа содержит два класса - 2Б и 2А.

Первая группа классифицирует многопользовательские АС, в которых одновременно обрабатывается и (или) хранится информация разных уровней конфиденциальности, и не все пользователи имеют право доступа ко всей информации АС.

Группа содержит пять классов - 1Д, 1Г, 1В, 1Б и 1А."

Сведем в таблицу требования ко всем девяти классам защищенности АС (табл. 2).

Таблица 2.2.

Подсистемы и требования		Классы защищенности								
		3Б	3А	2Б	2А	1Д	1Г	1В	1Б	1А
1. Подсистема управления доступом										
1.1.	Идентификация, проверка подлинности и контроль доступа субъектов в систему	+	+	+	+	+	+	+	+	+
	к терминалам, ЭВМ, узлам сети ЭВМ, каналам связи, внешним устройствам ЭВМ				+		+	+	+	+
	к программам				+		+	+	+	+
	к томам, каталогам, записям, полям				+		+	+	+	+

	записей									
1.2.	Управление потоками информации				+			+	+	+
2. Подсистема регистрации и учета										
2.1.	Регистрация и учет: входа/выхода субъектов доступа в/из системы (узла сети)	+	+	+	+	+	+	+	+	+
	выдачи печатных (графических) выходных документов		+		+		+	+	+	+
	запуска/завершения программ и процессов (заданий, задач)				+		+	+	+	+
	доступа программ субъектов к защищаемым файлам, включая их создание, удаление и передачу по линиям и каналам связи				+		+	+	+	+
	доступа к терминалам, ЭВМ, узлам сети ЭВМ, каналам связи, внешним устройствам ЭВМ, программам, томам, каталогам, записям, полям записей				+		+	+	+	+
	изменения полномочий субъектов доступа							+	+	+
	создаваемых защищаемых объектов доступа				+			+	+	+
2.2.	Учет носителей информации	+	+	+	+	+	+	+	+	+
2.3.	Очистка (обнуление, обезличивание) освобождаемых областей оперативной памяти ЭВМ и внешних накопителей		+		+		+	+	+	+
2.4.	Сигнализация попыток нарушеня защиты							+	+	+
3. Криптографическая подсистема										
3.1.	Шифрование конфиденциальной информации				+				+	+
3.2.	Шифрование информации, принадлежащей разным субъектам доступа (группам субъектов), на разных ключах									+
3.3.	Использование аттестованых (сертифицированных) криптографических средств				+				+	+
4. Подсистема обеспечения целостности										
4.1.	Обеспечение целостности программных средств и обрабатываемой информации	+	+	+	+	+	+	+	+	+
4.2.	Физическая охрана средств вычислительной техники и носителей информации	+	+	+	+	+	+	+	+	+
4.3.	Наличие администратора (службы) защиты информации в АС				+			+	+	+
4.4.	Периодическое тестирование средств защиты от НСД	+	+	+	+	+	+	+	+	+
4.5.	Наличие средств восстановления	+	+	+	+	+	+	+	+	+

	СЗИ от НСД									
4.6.	Использование сертифицированных средств защиты		+		+			+	+	+

Обозначения:

“+” – есть требования к данному классу

Приведем подробное изложение требований к достаточно представительному классу защищенности - 1В. Мы позволяем себе многостраничное цитирование по двум причинам. Во-первых, данные требования, несомненно, важны с практической точки зрения. Лица, отвечающие за информационную безопасность, должны сопоставлять свои действия с Руководящими указаниями, чтобы обеспечить систематичность защитных мер. Во-вторых, брошюры Гостехкомиссии при Президенте РФ являются библиографической редкостью, и ознакомиться с ними в подлиннике затруднительно.

«2.13. Требования к классу защищенности 1В:

Подсистема управления доступом:

должна осуществляться идентификация и проверка подлинности субъектов доступа при входе в систему по идентификатору (коду) и паролю условно-постоянного действия длиной не менее шести буквенно-цифровых символов;

должна осуществляться идентификация терминалов, ЭВМ, узлов сети ЭВМ, каналов связи, внешних устройств ЭВМ по логическим именам и/или адресам;

должна осуществляться идентификация программ, томов, каталогов, файлов, записей, полей записей по именам;

должен осуществляться контроль доступа субъектов к защищаемым ресурсам в соответствии с матрицей доступа;

должно осуществляться управление потоками информации с помощью меток конфиденциальности. При этом уровень конфиденциальности накопителей должен быть не ниже уровня конфиденциальности записываемой на него информации.

Подсистема регистрации и учета:

должна осуществляться регистрация входа/выхода субъектов доступа в систему/из системы, либо регистрация загрузки и инициализации операционной системы и ее программного останова;

должна осуществляться регистрация выдачи печатных (графических) документов на "твердую" копию;

должна осуществляться регистрация запуска/завершения программ и процессов (заданий, задач), предназначенных для обработки защищаемых файлов;

должна осуществляться регистрация попыток доступа программных средств к следующим дополнительным защищаемым объектам доступа: терминалам, ЭВМ, узлам сети ЭВМ, линиям (каналам) связи, внешним устройствам ЭВМ, программам, томам, каталогам, файлам, записям, полям записей;

должна осуществляться регистрация изменений полномочий субъектов доступа и статуса объектов доступа;

должен осуществляться автоматический учет создаваемых защищаемых файлов с помощью их дополнительной маркировки, используемой в подсистеме управления доступом. Маркировка должна отражать уровень конфиденциальности объекта;

должен проводиться учет всех защищаемых носителей информации с помощью их любой маркировки;

должна осуществляться очистка (обнуление, обезличивание) освобождаемых областей оперативной памяти ЭВМ и внешних накопителей. Очистка осуществляется произвольной двукратной записью в любую освобождаемую область памяти, использованную для хранения защищаемой информации;

должна осуществляться сигнализация попыток нарушения защиты.

Подсистема обеспечения целостности:

должна быть обеспечена целостность программных средств СЗИ НСД, а также неизменность программной среды, при этом:

целостность СЗИ НСД проверяется при загрузке системы по контрольным суммам компонент СЗИ, целостность программной среды обеспечивается использованием трансляторов с языков высокого уровня и отсутствием средств модификации объектного кода программ при обработке и (или) хранении защищаемой информации;

должна осуществляться физическая охрана СВТ (устройств и носителей информации), предусматривающая постоянное наличие охраны территории и здания, где размещается АС, с помощью технических средств охраны и специального персонала, использование строгого пропускного режима, специальное оборудование помещений АС;

должен быть предусмотрен администратор (служба) защиты информации, ответственный за ведение, нормальное функционирование и контроль работы СЗИ НСД. Администратор должен иметь свой терминал и необходимые средства оперативного контроля и воздействия на безопасность АС;

должно проводиться периодическое тестирование всех функций СЗИ НСД с помощью специальных программных средств не реже одного раза в год;

должны быть в наличии средства восстановления СЗИ НСД, предусматривающие ведение двух копий программных средств СЗИ НСД и их периодическое обновление и контроль работоспособности;

должны использоваться сертифицированные средства защиты».

По существу перед нами - минимум требований, которым необходимо следовать, чтобы обеспечить конфиденциальность защищаемой информации.

2.4. Особенности информационной безопасности вычислительных сетей

"Оранжевая книга" Министерства обороны США и Руководящие документы Гостехкомиссии при Президенте РФ создавались в расчете на централизованные конфигурации, основу которых составляют большие машины. Распределенная организация современных информационных систем требует внесения существенных изменений и дополнений как в политику безопасности, так и в способы проведения ее в жизнь. Появились новые угрозы, для противодействия которым нужны новые функции и механизмы защиты.

Основополагающим документом в области защиты распределенных систем стали рекомендации X.800 [9]. В данном разделе мы рассмотрим эту работу, а также интерпретацию "Критериев" Министерства обороны США для сетевых конфигураций [8].

2.4.1. Рекомендации X.800

Рекомендации X.800 - документ довольно обширный. Мы сосредоточим внимание на специфически сетевых функциях (сервисах) безопасности, а также на необходимых для их реализации защитных механизмах. Одновременно мы познакомимся с основными понятиями данной области информационной безопасности.

Чтобы почувствовать специфику распределенных систем, достаточно рассмотреть такое стандартное средство защиты, как подотчетность. Помимо других целей, записи в регистрационном журнале могут служить доказательством того, что определенный пользователь совершил то или иное действие (точнее, действие было совершено от его имени). В результате пользователь не может отказаться от совершённого им действия и несет полную ответственность за результат. В распределенных системах действие порой совершается на нескольких компьютерах и, вообще говоря, не исключено, что их регистрационные журналы противоречат друг другу. Так бывает, когда злоумышленнику удается подделать сетевой адрес и имя другого пользователя. Значит, нужны иные средства обеспечения «неотказуемости» (невозможности отказаться от совершенных действий).

2.4.1.1. Функции (сервисы) безопасности

Перечислим сервисы безопасности, характерные для распределенных систем, и роли, которые они могут играть. Вопросы реализации этих сервисов рассматриваются в следующем пункте.

Аутентификация. Данная функция обеспечивает аутентификацию партнеров по общению и аутентификацию источника данных. Аутентификация партнеров по общению используется как при установлении соединения, так и периодически во время сеанса. Она служит для предотвращения таких угроз, как маскарад и повтор предыдущего сеанса связи. Аутентификация источника данных - это подтверждение подлинности источника отдельной порции данных. Функция не обеспечивает защиты против повторной передачи данных.

Управление доступом. Управление доступом обеспечивает защиту от несанкционированного использования ресурсов, доступных по сети.

Конфиденциальность данных. Данная функция обеспечивает защиту от несанкционированного получения информации. Различают следующие виды конфиденциальности:

- конфиденциальность данных при общении с установлением соединения (в этом и следующем случаях защищаются вся пользовательская информация);
- конфиденциальность данных при общении без установления соединения;
- конфиденциальность отдельных полей данных (избирательная конфиденциальность);
- конфиденциальность трафика (защита информации, которую можно получить, анализируя трафик).

Целостность данных. Данная функция подразделяется на подвиды в зависимости от того, какой тип общения используют партнеры – с установлением соединения или без

такового, защищаются ли все данные или только отдельные поля, обеспечивается ли восстановление в случае нарушения целостности.

Неотказуемость. Данная функция (невозможность отказаться от совершенных действий) обеспечивает два вида услуг:

- неотказуемость с подтверждением подлинности источника данных;
- неотказуемость с подтверждением доставки.

Побочным продуктом неотказуемости является аутентификация источника данных.

Таблица 2.3.

Функция безопасности	Уровень модели ISO						
	1	2	3	4	5	6	7
Аутентификация			+	+			+
Управление доступом			+	+			+
Конфиденциальность соединения	+	+	+	+		+	+
Конфиденциальность вне соединения		+	+	+		+	+
Избирательная конфиденциальность						+	+
Конфиденциальность трафика	+		+				+
Целостность с восстановлением				+			+
Целостность без восстановления			+	+			+
Избирательная целостность							+
Целостность вне соединения			+	+			+
Неотказуемость							+

Обозначения:

“+” – данный уровень может предоставить функцию безопасности

В таблице 3 указаны уровни эталонной семиуровневой модели OSI, на которых могут быть реализованы функции безопасности. Отметим, что прикладные процессы в принципе могут взять на себя поддержку всех защитных сервисов.

2.4.1.2. Механизмы безопасности

Для реализации функций безопасности могут использоваться следующие механизмы и их комбинации.

Шифрование. Шифрование подразделяется на симметричное (с секретным ключом, когда знание ключа шифрования влечет знание ключа расшифровки) и асимметричное (с открытым ключом, когда знание ключа шифрования не позволяет узнать ключ расшифровки). Различают также обратимое и необратимое шифрование. Последнее может использоваться для вычисления криптографических контрольных сумм (хэш-функций, дайджестов, имитовставок).

Электронная (цифровая) подпись. Механизм электронной подписи включает в себя две процедуры:

- выработку подписи;
- проверку подписанной порции данных.

Процедура выработки подписи использует информацию, известную только субъекту, подписывающему порцию данных. Процедура проверки подписи является общедоступной, она не должна позволять найти секретный ключ подписывающего субъекта.

Механизмы управления доступом. При принятии решений по поводу предоставления запрашиваемого типа доступа могут использоваться следующие виды и источники информации:

- Базы данных управления доступом. В такой базе, поддерживаемой централизованно или на оконечных системах, могут храниться списки управления доступом или структуры аналогичного назначения.
- Пароли или иная аутентификационная информация. Токены, билеты или иные удостоверения, предъявление которых свидетельствует о наличии прав доступа.
- Метки безопасности, ассоциированные с субъектами и объектами доступа.
- Время запрашиваемого доступа.
- Маршрут запрашиваемого доступа.
- Длительность запрашиваемого доступа.

Механизмы управления доступом могут располагаться на любой из общающихся сторон или в промежуточной точке. В промежуточных точках целесообразно проверять права доступа к коммуникационным ресурсам. Очевидно, требования механизма, расположенного на приемном конце, должны быть известны заранее, до начала общения.

Механизмы контроля целостности данных. Различают два аспекта целостности: целостность отдельного сообщения или поля информации и целостность потока сообщений или полей информации. Вообще говоря, контроль двух видов целостности осуществляется различными механизмами, хотя контролировать целостность потока, не проверяя отдельные сообщения, едва ли имеет смысл. Процедура контроля целостности отдельного сообщения (поля) включает в себя два процесса - один на передающей стороне, другой на приемной. На передающей стороне к сообщению добавляется избыточная информация, которая является функцией от сообщения (та или иная разновидность контрольной суммы). На приемной стороне независимо генерируется контрольная сумма полученного сообщения с последующим сравнением результатов. Данный механизм сам по себе не защищает от дублирования сообщений. Для проверки целостности потока сообщений (то есть для защиты от кражи, переупорядочивания, дублирования и вставки сообщений) используются порядковые номера, временные штампы, криптографическое связывание (когда результат шифрования очередного сообщения зависит от предыдущего) или иные аналогичные приемы. При общении в режиме без установления соединения использование временных штампов может обеспечить ограниченную форму защиты от дублирования сообщений.

Механизмы аутентификации. Аутентификация может достигаться за счет использования паролей, личных карточек или иных устройств аналогичного назначения, криптографических методов (когда демонстрируется знание секретного ключа), устройств измерения и анализа биометрических характеристик. Аутентификация бывает односторонней (обычно клиент доказывает свою подлинность серверу) и двусторонней (взаимной). Пример односторонней аутентификации - процедура входа пользователя в систему. Для защиты от дублирования аутентификационной информации могут использоваться временные штампы и синхронизация часов в узлах сети.

Механизмы дополнения трафика. Механизмы дополнения трафика, разумеется, эффективны только в сочетании со средствами обеспечения конфиденциальности, поскольку в противном случае злоумышленнику будет очевиден фиктивный характер дополнительных сообщений.

Механизмы управления маршрутизацией. Маршруты могут выбираться статически или динамически. Оконечная система, зафиксировав неоднократные атаки на определенном маршруте, может отказаться от его использования. На выбор маршрута способна повлиять метка безопасности, ассоциированная с передаваемыми данными.

Механизмы нотаризации. Механизм нотаризации служит для заверения таких коммуникационных характеристик, как целостность, время, личности отправителя и получателей. Заверение обеспечивается надежной третьей стороной, которая обладает достаточной информацией, чтобы ее заверениям можно было доверять. Обычно нотаризация опирается на механизм электронной подписи.

Таблица 2.4.

Механизмы / Функции безопасности	Шифрование	Электронная подпись	Управление	Целостность	Аутентификация	Дополнение	Управление	Нотаризация
Аутентификация партнеров	+	+			+			
Аутентификация источника	+	+						
Управление доступом			+					
Конфиденциальность	+						+	
Избирательная конфиденциальность	+							
Конфиденциальность трафика	+					+	+	
Целостность соединения	+			+				
Целостность вне соединения	+	+		+				
Неотказуемость		+		+				+

Обозначения:

“+” – механизм пригоден для реализации данной функции безопасности

В таблице 4 сведены функции и механизмы безопасности. Таблица показывает, какие механизмы (по одиночке или в комбинации с другими) могут использоваться для реализации той или иной функции.

2.4.1.3. Администрирование средств безопасности

Администрирование средств безопасности включает в себя распространение информации, необходимой для работы функций и механизмов безопасности, а также сбор и анализ информации об их функционировании. Примерами могут служить распространение криптографических ключей, установка значений параметров защиты, ведение регистрационного журнала и т.п.

Концептуальной основой администрирования является информационная база управления безопасностью. Эта база может не существовать как единое (распределенное) хранилище, но каждая из оконечных систем должна располагать информацией, необходимой для проведения в жизнь избранной политики безопасности.

Усилия администратора средств безопасности должны распределяться по трем направлениям:

- администрирование системы в целом;
- администрирование функций безопасности;

- администрирование механизмов безопасности.

Среди действий, относящихся к системе в целом, отметим поддержание актуальности политики безопасности, взаимодействие с другими административными службами, реагирование на происходящие события, аудит и безопасное восстановление.

Администрирование функций безопасности включает в себя определение защищаемых объектов, выработку правил подбора механизмов безопасности (при наличии альтернатив), комбинирование механизмов для реализации функции безопасности, взаимодействие с другими администраторами для обеспечения согласованной работы.

Обязанности администратора механизмов безопасности определяются перечнем задействованных механизмов. Типичный список таков:

Управление ключами (генерация и распределение). Вероятно, многие аспекты управления ключами (например, их доставка) выходят за пределы среды OSI.

Управление шифрованием (установка и синхронизация криптографических параметров). К управлению шифрованием можно отнести и администрирование механизмов электронной подписи. Управление целостностью, если оно обеспечивается криптографическими средствами, также тяготеет к данному направлению.

Администрирование управления доступом (распределение информации, необходимой для управления - паролей, списков доступа и т.п.).

Управление аутентификацией (распределение информации, необходимой для аутентификации - паролей, ключей и т.п.).

Управление дополнением трафика (выработка и поддержание правил, задающих характеристики дополняющих сообщений - частоту отправки, размер и т.п.). Характеристики могут варьироваться по заданному закону в зависимости от даты и времени.

Управление маршрутизацией (выделение надежных путей).

Управление нотаризацией (распространение информации о нотариальных службах, администрирование этих служб).

Мы видим, что администрирование средств безопасности в распределенной среде имеет много особенностей по сравнению с централизованными системами.

2.4.2. Интерпретация “ОРАНЖЕВОЙ КНИГИ” для сетевых конфигураций

В 1987 году Национальный центр компьютерной безопасности США выпустил в свет интерпретацию "Оранжевой книги" для сетевых конфигураций [8]. Данный документ состоит из двух частей. Первая содержит собственно интерпретацию, во второй рассматриваются сервисы безопасности, специфичные или особенно важные для сетевых конфигураций.

2.4.2.1. Интерпретация

В первой части вводится минимум новых понятий. Важнейшее из них - надежная сетевая вычислительная база, распределенный аналог надежной вычислительной базы изолированных систем. Надежная сетевая вычислительная база формируется из всех частей всех компонентов сети, обеспечивающих информационную безопасность. Надежная сетевая система должна обеспечивать такое распределение защитных механизмов, чтобы общая

политика безопасности проводилась в жизнь, несмотря на уязвимость коммуникационных путей и асинхронную работу компонентов.

Не существует прямой зависимости между вычислительными базами компонентов, рассматриваемых как изолированные системы, и фрагментами сетевой вычислительной базы. Более того, нет прямой зависимости и между уровнями безопасности отдельных компонентов и уровнем безопасности всей сетевой конфигурации. Например, в результате объединения двух систем класса B1, обладающих несовместимыми правилами кодирования меток безопасности, получается сеть, не удовлетворяющая требованию обеспечения целостности меток. В качестве противоположного примера рассмотрим объединение двух компонентов, один из которых сам не обеспечивает протоколирование действий пользователя, но передает необходимую информацию другому компоненту, который и ведет протокол. В таком случае сеть в целом, несмотря на слабость компонента, удовлетворяет требованию подотчетности.

Чтобы понять суть положений, вошедших в первую часть, рассмотрим интерпретацию требований к классу безопасности C2. Первое требование к этому классу - поддержка произвольного управления доступом. Интерпретация предусматривает различные варианты распределения надежной сетевой вычислительной базы по компонентам и, соответственно, различные варианты распределения механизмов управления доступом. В частности, некоторые компоненты, закрытые от прямого доступа пользователей (например, коммутаторы пакетов, оперирующие на третьем уровне семиуровневой модели OSI), могут вообще не содержать подобных механизмов.

Пользователь осуществляет доступ к удаленному ресурсу посредством суррогатного процесса, выполняющегося на удаленной системе от его имени. Данный процесс подвергается стандартным локальным процедурам контроля доступа. Интерпретация предусматривает различные способы ассоциирования идентификатора пользователя с суррогатным процессом. Может существовать единая идентификационная база данных, доступная каждому компоненту; могут быть реализованы лишь локальные базы, и тогда суррогатный процесс выполняется от имени незарегистрированного пользователя или по некоторым правилам получает идентификатор кого-либо из локальных пользователей.

Идентификация групп пользователей может строиться на основе сетевых адресов хостов или (под)сетей. В то же время регистрационный журнал должен содержать достаточно информации для ассоциирования действий с конкретным пользователем. Сетевой адрес может являться частью глобального идентификатора пользователя.

В принципе возможен централизованный контроль доступа, когда решения принимает специальный сервер авторизации. Возможен и смешанный вариант, когда сервер авторизации разрешает соединение двух хостов, а дальше в дело вступают локальные механизмы хоста, содержащего объект доступа.

Аналогично, идентификация и аутентификация пользователей может производиться как централизованно (соответствующим сервером), так и локально - той системой, с которой пользователь непосредственно взаимодействует. Возможна передача идентификационной и аутентификационной информации между хостами (чтобы избавить пользователя от многократной аутентификации). При передаче аутентификационная информация должна быть защищена не слабее, чем на каждом из компонентов сетевой конфигурации.

В идентификации и аутентификации могут нуждаться не только пользователи, но и компоненты сети, такие как хосты.

Регистрационная информация в сетевом случае может включать в себя записи новых видов, например, сведения об установлении и разрыве соединений, о потенциальном нарушении целостности данных (в частности, ввиду неправильной маршрутизации датаграмм), об изменениях в конфигурации сети. "Адресное пространство пользователей" становится распределенным, а в число регистрируемых событий попадают действия с удаленными объектами (открытие, переименование и т.п.).

При ведении регистрационного журнала могут использоваться локальные или глобальные синхронизированные часы.

Регистрационные журналы разных компонентов сети должны быть согласованы между собой; должны предоставляться средства для комплексного анализа совокупности регистрационных журналов с целью глобального отслеживания деятельности пользователей.

Возможно выделение в сети одного или нескольких серверов протоколирования и аудита, обслуживающих другие компоненты, не имеющие ресурсов или по иным причинам не желающие вести протоколирование самостоятельно.

Переходя к рассмотрению вопросов гарантированности, отметим, что каждая часть надежной сетевой вычислительной базы, расположенная на отдельном компоненте, должна поддерживать отдельную область для собственного выполнения, защищенную от внешних воздействий.

Интерпретация отличается от самих "Критериев" учетом динамичности сетевых конфигураций.

Предусматривается наличие средств проверки подлинности и корректности функционирования компонентов перед их включением в сеть, наличие протокола взаимной проверки компонентами живучести и корректности функционирования друг друга, доступность средств оповещения администратора о неполадках в сети. Сетевая конфигурация должна быть устойчива к отказам отдельных компонентов или коммуникационных путей.

Динамичность, согласно Интерпретации, должна найти отражение в Руководстве администратора по средствам безопасности. Помимо прочих, это Руководство обязано освещать такие темы, как аппаратное конфигурирование сети, учет последствий подключения новых компонентов или отключения старых.

В качестве еще одного отличительного момента Интерпретации отметим повышенное внимание к целостности информации вообще и меток безопасности в частности (мы переходим к рассмотрению некоторых аспектов принудительного управления доступом, характерного для уровня безопасности B). Для контроля целостности меток и для их защиты от нелегального изменения в Интерпретации рекомендуется широкое использование криптографических методов. Далее, чтобы принудительное управление доступом в распределенной конфигурации имело смысл, совокупность уровней секретности и категорий должна поддерживаться централизованно. В этом одно из принципиальных отличий от произвольного управления доступом.

В целом следует отметить довольно очевидный характер первой части Интерпретации, что, впрочем, является прямым следствием выбранного методологического подхода. Описание существенно новых сервисов и механизмов вынесено во вторую часть документа.

Если первая часть посвящена в основном управлению доступом к информации, то во второй нашли отражение все основные аспекты безопасности - конфиденциальность, целостность и доступность.

2.4.2.2. Новые сервисы безопасности и защитные механизмы

Рассматриваемый документ создавался примерно в то же время, что и рекомендации X.800. Естественно, что две рабочие группы обменивались информацией, поэтому во многих отношениях их подходы схожи. Имеются, однако, и важные различия. Интерпретация не замыкается на эталонной семиуровневой модели, ее цель – оценка безопасности всей распределенной конфигурации, а не только чисто сетевых аспектов. Рекомендации X.800 в основном имеют дело с функциональностью (с сервисами безопасности) и в меньшей степени с защитными механизмами. В части 2 Интерпретации анализируется еще одна важнейшая характеристика - гарантированность.

Основой функционирования сетей вообще и коммуникационной безопасности в частности являются сетевые протоколы. Многие защитные механизмы встраиваются в протоколы. От протоколов зависит защита системы от тупиков и иных обстоятельств, способных повлиять на доступность сервисов, а также наличие средств обнаружения ситуаций недоступности. Протоколы влияют и на возможность поддержания целостности данных.

Среди защитных механизмов в сетевых конфигурациях на первом месте стоит криптография, помогающая поддерживать как конфиденциальность, так и целостность. Следствием использования криптографических методов является необходимость реализации механизмов управления ключами.

Для поддержания целостности (в аспектах, относящихся к коммуникациям) используются аутентификация, контроль целостности полей и механизмы обеспечения неотказуемости. Мы не останавливаемся подробно на этом и предыдущем сервисах, поскольку они подробно рассматривались в связи с рекомендациями X.800.

Новым по сравнению с X.800 является рассмотрение вопросов доступности. Сетевой сервис перестает быть доступным, когда пропускная способность коммуникационных каналов падает ниже минимально допустимого уровня или сервис не в состоянии обслуживать запросы. Удаленный ресурс может стать недоступным и вследствие нарушения равноправия в обслуживании пользователей. Надежная система должна быть в состоянии обнаруживать ситуации недоступности, уметь возвращаться к нормальной работе и противостоять атакам на доступность.

Для обеспечения непрерывности функционирования могут применяться следующие защитные меры:

- Внесение в конфигурацию той или иной формы избыточности (резервное оборудование, запасные каналы связи и т.п.).
- Наличие средств реконфигурирования для изоляции и/или замены узлов или коммуникационных каналов, отказавших или подвергшихся атаке на доступность.
- Рассредоточенность сетевого управления, отсутствие единой точки отказа.
- Наличие средств нейтрализации отказов (обнаружение отказавших компонентов, оценка последствий, восстановление после отказов).
- Выделение подсетей и изоляция групп пользователей друг от друга.

С точки зрения оценки надежности систем, критерии части 2 дополняют "Оранжевую книгу". Каждый сервис безопасности рассматривается независимо и может получить одну из трех положительных оценок. Таким образом, общая оценка сетевой конфигурации выглядит примерно так: класс безопасности С2, сервис_1 - удовлетворительно, сервис_2 - хорошо и т.д. Заказчик, зная свои потребности, в состоянии принять решение о пригодности той или иной конфигурации.

2.4.2.3. Оценка надежности сетевой конфигурации на основе оценки компонентов

В части 1 Интерпретации излагается подход к оценке надежности сетевой конфигурации как единого целого. В то же время имеет право на существование и другой взгляд, когда сеть составляется из предварительно проверенных компонентов, а общая оценка по определенным правилам выводится из их "рейтинга". Подобная точка зрения является предметом рассмотрения приложений к Интерпретации, где анализируются три главных вопроса:

- Как следует структурировать сеть, чтобы оценка компонентов помогала получить общую оценку?
- Какие критерии следует применять к компонентам?
- Как получать общую оценку?

Предварительным условием надежности сетевой конфигурации является наличие единой политики безопасности, с которой должны быть согласованы поведение каждого компонента и характер связей между ними. В Интерпретации рассматриваются следующие аспекты политики безопасности:

- произвольное управление доступом;
- принудительное управление доступом;
- идентификация и аутентификация;
- протоколирование и аудит.

Одним из важнейших в "Оранжевой книге" является понятие монитора обращений (см. раздел 2.1.1 "Основные понятия"). Применительно к структурированию сетевой конфигурации можно сформулировать следующее утверждение, дающее достаточное условие корректности фрагментирования монитора обращений.

Утверждение 1. Пусть каждый субъект (то есть процесс, действующий от имени какого-либо пользователя) заключен внутри одного компонента и может осуществлять непосредственный доступ к объектам только в пределах этого компонента. Пусть, далее, каждый компонент содержит свой монитор обращений, отслеживающий все локальные попытки доступа, и все мониторы проводят в жизнь согласованную политику безопасности. Пусть, наконец, коммуникационные каналы, связывающие компоненты, сохраняют конфиденциальность и целостность передаваемой информации. Тогда совокупность всех мониторов образует единый монитор обращений для всей сетевой конфигурации.

Истинность этого утверждения непосредственно следует из определения монитора обращений.

Отметим разумность структурирования на компоненты, содержащие собственные мониторы обращений. Обычно каждый такой компонент предоставляет законченный набор услуг, а, значит, его выделение естественно и целесообразно не только с точки зрения безопасности, но и с функциональной точки зрения.

Таким образом, сетевые конфигурации рекомендуется структурировать на компоненты, предоставляющие определенные виды сервиса и отслеживающие обращения к своим объектам, и на коммуникационные каналы, защищенные надежными сетевыми сервисами (использующими, как правило, криптографические механизмы).

Оценка компонентов производится по обычным критериям "Оранжевой книги" с одной важной оговоркой. Каждый компонент, вообще говоря, не обязан поддерживать все перечисленные выше аспекты политики безопасности. В этом случае к нему нужно применять соответствующее подмножество критериев. Компоненты, поддерживающие лишь часть аспектов политики безопасности, должны обладать программными и/или протокольными интерфейсами, чтобы получить недостающие им сервисы от других компонентов, предоставляющих такую возможность.

При оценке сетевой конфигурации принимается во внимание тип компонентов и присвоенный им класс безопасности. Комбинируя четыре аспекта политики безопасности, каждый из которых может независимо поддерживаться или не поддерживаться, получаем 15 типов компонентов и их комбинаций (случай, когда не поддерживается ни один аспект, не рассматривается). Как правило, условия корректности комбинаций и итоговый класс безопасности очевидным образом следуют из обычных критериев. Так, при объединении двух компонентов, поддерживающих произвольное управление доступом, необходимо, чтобы был определен протокол передачи идентификационной информации, на которой основываются решения о предоставлении запрашиваемого вида доступа. Политика безопасности каждого компонента и их объединения должна быть согласована с общей политикой. Итоговый класс безопасности объединения равен минимальному из присвоенных компонентам классов.

При объединении компонента с произвольным управлением доступом и компонента, поддерживающего идентификацию и аутентификацию, должны сохраниться возможности обоих компонентов и, кроме того, для классов C2 и выше, необходимо наличие интерфейса к компонентам протоколирования и аудита. Если компонент идентификации отнесен к классу безопасности C2, то итоговый класс объединения совпадает с классом компонента с добровольным управлением доступом.

Мы не будем останавливаться на всех возможных способах построения составных компонентов.

3. Обоснование выбранного направления разработки

В настоящее время важность обеспечения безопасности информации в вычислительных и информационных системах уже не вызывает сомнения. Вместе с тем, принятые стандарты не дают ответов на самый важный вопрос – как именно обеспечить безопасность в системе в течение всего ее жизненного цикла. Как это уже отмечалось ранее, стандарты и большинство работ в этом направлении лишь выдвигают требования к системам, не описывая процедур безопасности.

Существующие попытки описать процедуры безопасности, как правило, отличаются спонтанностью и более всего направлены на описание частных вопросов защиты определенных сетевых сервисов или протоколов. Такие работы, несомненно, полезны, однако они не способны дать целостной картины безопасности всей информационной системы.

Принимая во внимание вышеизложенное, можно сделать вывод о том, что наиболее важной целью настоящей работы является создание целостного, системного подхода к обеспечению безопасности информационной системы. Соответственно, важно не просто описать отдельные процедуры безопасности, но показать их совокупное влияние на безопасность всей информационной системы. Именно такой подход и предполагается заданием.

Какие же шаги необходимо предпринять, чтобы создать комплексную, максимально общую методику обеспечения безопасности? Для того чтобы можно было наметить план действий, необходимо сначала рассмотреть современные информационные системы. Что же мы увидим?

Современные информационные системы строятся на базе локальных и глобальных сетей, по технологии «клиент-сервер», с использованием самых разных сетевых служб и протоколов, и призваны решать самые разнообразные задачи. Типичная вычислительная сеть в наше время состоит из десятков серверов, сотен рабочих станций. В ней используются сложные СУБД, файловые сервисы, сервисы печати, электронная почта и т.д. Многие сети имеют подключения к другим сетям или Интернет. Часто сами по себе, внутри организации, сети так же делятся на уровни – например, сети отделов или департаментов. Многие организации содержат собственные www-серверы, ftp-архивы и прочие службы. Словом, типичная сеть предприятия в наше время – это весьма сложный, неоднородный и запутанный организм.

Прямой анализ безопасности таких сложных систем не просто затруднен, он практически невозможен. Следовательно, необходимо описать принципы, по которым станет возможно декомпозировать сложную сеть на ряд простых составляющих. После этого анализ составляющих по отдельности уже не будет столь сложен.

Для того чтобы была возможность произвести декомпозицию сети на составляющие, необходимо, прежде всего, абстрагироваться от конкретных задач, решаемых этой сетью и описать ее максимально общими, формальными терминами. Это и будет первым этапом в разработке методики обеспечения безопасности. Иными словами, необходимо создать модель информационной системы.

Разумеется, всякая модель – лишь приближенное описание реальной системы. Следовательно, имеются определенные границы применимости этой модели. В нашем

случае, границы применимости фактически уже были описаны ранее – локальная сеть с подключением к глобальной сети, использование технологии «клиент-сервер». Кроме этого, необходимо указать, что критическая информация не может находиться на рабочих станциях. То есть местами хранения информации являются только (!) серверы.

В модель обязательно должны войти не только составляющей самой информационной системы, но и ближайшее окружение - например, персонал, работающий с информацией.

Особое место необходимо уделить политике безопасности. Фактически, Политика безопасности – это важнейший из активных элементов защиты информации, дающий возможность определить, что именно и какими средствами необходимо защищать. Политика безопасности так же может (и должна) описывать разделение функций обеспечения безопасности между прочими элементами системы. К сожалению, Политика безопасности очень часто оказывается обделенной вниманием при рассмотрении вопросов обеспечения безопасности. Поскольку в нашем случае особенно важно рассмотреть системный подход к обеспечению безопасности, то Политика безопасности должна занимать особое место в рассмотрении.

После выделения основных элементов информационной системы (среди них, как уже говорилось, обязательно должны быть Политика безопасности и персонал), необходимо рассмотреть основные моменты, связанные с обеспечением безопасности информации на каждом уровне модели.

После того, как модель создана и рассмотрена, можно переходить к анализу и оценке безопасности всей информационной системы с практической точки зрения. Какие важнейшие шаги тут необходимо рассмотреть?

Прежде всего, отметим, что по сути своей, процесс обеспечения безопасности – это процесс управления рисками в информационной системе. Рисками, связанными с нарушением целостности, доступности или конфиденциальности информации. Поэтому важнейшим с практической точки зрения будет рассмотрение процесса анализа и управления рисками в информационной системе. Именно анализ возможных угроз и рисков позволяет в целом оценить безопасность информации, выделить слабые места и принять экономически обоснованные решения об устранении слабостей в защите.

Отдельного рассмотрения заслуживает вопрос физической защиты информационной системы. Действительно, для организации не важно, по каким причинам произойдет утеря информации – из-за кражи, хулиганства или пожара. Важно обеспечить минимальную опасность для информации. Здесь особо важно отметить физическую защиту как часть общей Политики безопасности предприятия.

Наконец, какую бы защиту мы не строили, всегда существует вероятность нарушения режима безопасности. К сожалению, абсолютной защиты не бывает и это надо понимать. Поэтому организация-владелец сети должна быть готова к тому, что инциденты нарушения Политики безопасности могут произойти. Соответственно, необходимо иметь инструкции и правила действий в случае таких инцидентов. При разработке рекомендаций по обеспечению режима безопасности этому вопросу должно быть уделено особое внимание, как, впрочем, и вопросу о мерах, предпринимаемых после нарушения. Каждый инцидент нарушения режима безопасности должен явиться уроком, из которого обязательно должны быть сделаны выводы. Меры, предпринимаемые после нарушения режима безопасности, призваны помочь

обнаружить слабости, из-за которых нарушение стало возможным, и наиболее эффективно устранить их.

Наконец, все рекомендации по обеспечению режима безопасности будут просто пусты звуком, если не привести конкретных примеров. Следовательно, в работе необходимо рассмотреть возможности, предоставляемые современными операционными системами для обеспечения режима безопасности.

Резюмируя, кратко описать план работы можно так:

- Модель информационной системы
- Возможности, предоставляемые каждым уровнем модели для обеспечения безопасности
- Процесс управления рисками в информационной системе
- Физическая защита как часть общей Политики безопасности
- Основные программно-технические меры обеспечения безопасности
- Реакция на нарушение режима безопасности
- Меры, предпринимаемые после инцидента нарушения
- Практические средства обеспечения безопасности в современной операционной системе.

4. Разработка архитектурной модели безопасности ИС и сетей на базе UNIX

В этом разделе мы постараемся создать некую модель информационной системы. Зачем нужна такая модель? Как уже упоминалось выше, современные информационные системы чрезвычайно сложны и многообразны. Типичная информационная система состоит из десятков серверов, сотен клиентских рабочих станций. Она включает в себя многочисленные сетевые службы и сервисы: системы печати, электронной почты, разнообразные базы данных и т.д. Все это делает такую систему крайне сложной для анализа. С тем, чтобы была возможность произвести анализ такой системы, ее необходимо декомпозировать на менее сложные компоненты. Предъявляя определенные требования к компонентам и связям между ними, мы можем составить общую оценку безопасности системы.

Создание модели информационной системы позволяет в чистом виде выделить основные компоненты информационной системы. При этом существует возможность абстрагироваться от конкретных задач, решаемых тем или иным компонентом, и тем самым, добиться необходимого уровня общности модели. В дальнейшем, используя базовые понятия модели можно любую реальную систему привести к общему виду, и уже затем произвести ее анализ. Кроме перечисленных выгод, представление модели позволит систематизировать понятия и подходы к информационной безопасности реальной системы. Как уже упоминалось выше, принятые в мире стандарты информационной безопасности не дают практических рекомендаций о путях построения безопасных информационных систем. Таким образом, модель послужит еще одной цели – создать некий «мостик» от рекомендаций к практическим шагам по обеспечению режима безопасности.

4.1. Общие положения

Ниже мы приведем предлагаемую модель информационной системы для оценки безопасности. Для каждого обозначенного уровня далее по тексту будет дано подробное описание. Как несложно заметить, модель состоит из 7-ми уровней. Отчасти это делалось для того, чтобы получить некое сходство с 7-ми уровневой сетевой моделью ISO/OSI, с которой хорошо знакомы администраторы UNIX. Уровни выстраивались по мере уменьшения их общности и широты. В таблице 4.1 приведены уровни модели, их название и краткое описание.

В целом, приведенная модель ориентирована на протокол IP, используемый в Интернет, но ее можно с минимальными модификациями так же использовать и для любой другой сетевой технологии. Следует отметить, что данная модель, вообще говоря, применима не только к UNIX-системам, но так же и к любым другим сетевым системам. Специфичны для конкретной ОС лишь реализация этой модели и конкретный набор программных средств.

Таблица 4.1

№ уровня	Название	Функциональное описание
7	Политика безопасности	Общие правила обеспечения безопасности предприятия
6	Персонал	Люди, использующие оборудование и данные (как правило, работники компании)
5	Локальная сеть	Компьютерное оборудование и внутренние линии связи: этажные коммутационные щиты, коммутационные комнаты, концентраторы и/или коммутаторы, маршрутизаторы т.д.
4	Сетевые сервисы	Сервисы баз данных, электронной почты, сетевых имен, файловые, печати, аутентификации, журналирования и т.д.
3	Брандмауэр	Firewall, прокси-серверы и т.д. – устройства, обеспечивающие защиту на уровнях 7,6,5,4 модели OSI
2	Пакетный фильтр	Маршрутизатор или иное устройство, работающее на уровнях 3,2,1 модели OSI
1	Внешние каналы связи	Каналы связи и оборудование, используемое для подключения к глобальным или иным сетям (модемы, выделенные или телефонные линии и т.д.)

В составе модели имеется 3 уровня, отвечающие за соединения с другими сетями. Заметим, что под другими сетями вовсе не обязательно понимать Интернет. Данная модель может быть применена для описания сети, например, отдела или подразделения. В этом случае под понятием «другие сети» следует понимать сети других отделов. Они могут быть соединены как локальной сетью в случае нахождения в одном здании, так и глобальной сетью, в случае территориально распределенной сети крупной организации. Наконец, это может быть Интернет, что с точки зрения модели несущественно. В терминах предложенной модели достаточно того, что «внешняя сеть» и каналы связи с ней находится вне юрисдикции отдела или организации, следовательно, потенциально являются враждебными. Однако эти сети могут предлагать какой-то сервис или некое множество сервисов, пользование которыми является желательным для персонала отдела/организации.

Схема взаимодействия уровней модели между собой приведена на рис 4.1.

Несколько слов о серверах и клиентах. Как можно заметить, в данной модели отсутствует понятие “сервер”, несмотря на то, что выше было указано, что для анализа принимается модель клиент-сервер. Это действительно так. Поясним, почему в модели не фигурирует прямо понятие “сервер” и понятие “клиент”.

Понятие “сервер” было заменено понятием “сервис”. Сделано это по двум причинам. Во-первых, для понимания порой сложно разграничить понятие “сервер” как физическая машина, например, с большой памятью и дисковыми ресурсами, и понятие “сервер” как программное обеспечение, обеспечивающее, например, обработку запросов SQL. Это создает определенные сложности для понимания, интерпретации и практического применения модели безопасности. Введение же понятия «сервис» устраняет эту неоднозначность. Действительно, «сервис» в данном контексте – это служба, обеспечивающая клиентам определенный уровень функциональности. Это может быть сервис СУБД, обрабатывающий запросы SQL, может быть сервис DNS, выполняющий

разрешение имен и сетевых адресов, может быть сервис NIS+, решающий задачи сетевого управления, наконец, это может быть просто файловый сервис или сервис печати.

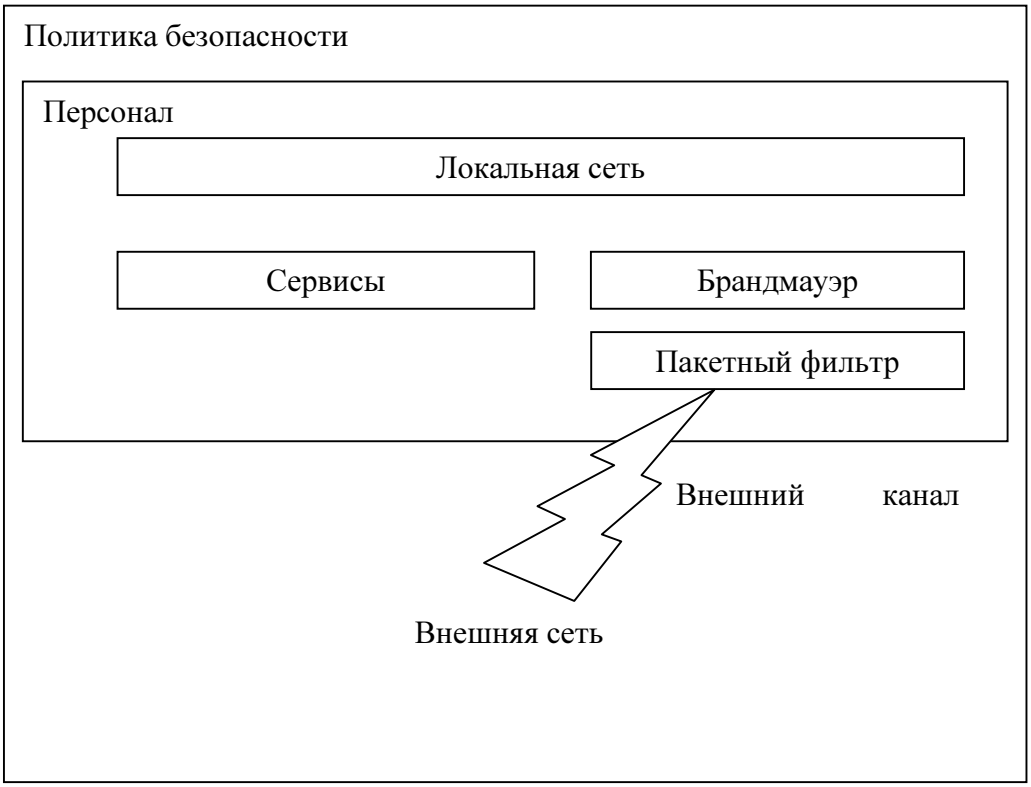


Рис 4.1 Схема взаимодействия уровней модели безопасности

В связи с вышесказанным, понятие «сервер» можно объяснить так: это физическая машина, предоставляющая некий сервис в общее пользование. При этом сервер может быть выделенным, если он предоставляет только один сервис, или невыделенным, если он предоставляет несколько сервисов. Данное определение выделенного и невыделенного сервера противоречит общепринятому. Однако в целях оценки безопасности сети такое определение является предпочтительным.

Заметим, что само понятие «сервер» не фигурирует в модели. Это легко объяснить, поскольку физически сервер практически неинтересен с точки зрения программно-технических и даже организационных аспектов обеспечения безопасности. Тем не менее, при рассмотрении практических вопросов обеспечения безопасности мы будем использовать понятие «сервер» именно в том смысле, какой мы ввели здесь. Потому что машины, предоставляющие сервисы в сеть все-таки нуждаются в особом подходе хотя бы в смысле обеспечения физической безопасности.

Наконец, в модели нет понятия «клиент». Это тоже легко объяснимо. Использование архитектуры клиент-сервер подразумевает хранение всей важной информации с помощью сервисов. То есть мы вправе предположить, что никакая критическая информация не будет храниться на локальных дисках пользователей. При таком допущении, кстати, вполне правомерном и даже необходимом, все рабочие станции и персональные компьютеры вкупе с программным обеспечением попадают в раздел «локальная сеть». В предложенной модели локальная сеть – это своего рода интерфейс между пользователем и первостепенными сервисами.

И последнее замечание. Оно необходимо, чтобы пояснить смысл значения “первостепенный сервис”, приведенного в предыдущем абзаце. Итак, что же такое “первостепенный сервис”? Именно так можно назвать сервисы, обеспечивающие необходимый функционал информационной системы. Это может быть, например, сервис СУБД. Под “второстепенными” или “служебными” сервисами мы будем понимать сервисы, обеспечивающие работу первостепенных сервисов или самой сети. К ним можно отнести, например, сервис bootp, NIS+, DNS и т.д. Они не принципиальны для функционала информационной системы, однако, необходимы.

Теперь, прояснив основные принципы построения модели, рассмотрим ее составные части подробнее.

4.2. Политика безопасности

В этом документе будет еще много ссылок на политику или правила безопасности. Часто эти ссылки будут так же включать рекомендации для определенных правил. Вместо повторения рекомендаций по созданию правил и политики безопасности и следованию им, читателю необходимо применить взгляды, выраженные в этом разделе при создании правил, рекомендованных далее в этой работе.

4.2.1. Что такое Политика безопасности и зачем она нужна?

Все решения по поводу безопасности, которые вы реализуете или пытаетесь реализовать как администратор, в огромной степени зависят от того, насколько защищена или незащищена ваша сеть, какую функциональность она предоставляет пользователям и насколько легка в использовании [10]. Однако вы не сможете принять грамотные решения в области безопасности, до тех пор, пока не определите, какие именно цели преследуете. До тех пор, пока вы не определите свои цели, вы так же не сможете хоть сколько-нибудь эффективно использовать любой набор инструментов и программ для обеспечения безопасности, поскольку вы просто не будете знать, что именно и как следует проверять, и какие ограничения использовать.

Например, ваши цели и цели производителя того или иного программного продукта могут отличаться. Производитель пытается сделать конфигурирование и работу своего продукта максимально легкой и простой, из чего следует, что начальная конфигурация продукта после установки может часто быть открытой, то есть – незащищенной. Насколько такая конфигурация упрощает установку нового программного продукта, настолько же она упрощает доступ к системе, а через эту систему – и прочим системам организации.

Ваши цели в большой степени должны определяться следующими моментами:

1. Предоставляемые сервисы – против обеспечиваемой безопасности. Каждый сервис, предоставляемый пользователю, несет в себе потенциальный риск. Для некоторых сервисов риск перевешивает пользу от сервиса. В этом случае, администратор может принять решение об исключении сервиса, вместо того, чтобы пытаться защитить его.
2. Легкость использования - против обеспечиваемой безопасности. Системы, которые проще всего использовать, должны предоставлять доступ с минимальными ограничениями для пользователей, возможно, даже не требовать пароля, однако, при этом невозможно обеспечить никакой безопасности. Требование пароля увеличивает

безопасность системы, но при этом уменьшается легкость в использовании. Требование же одноразового, аппаратно генерируемого пароля делает систему не слишком-то легкой в использовании, но зато гораздо более безопасной.

3. Стоимость мер безопасности – против стоимости возможных потерь. Существует много аспектов стоимости использования мер безопасности: чисто денежные (стоимость приобретения: оборудование для обеспечения безопасности, программное обеспечение – брандмауэры, генераторы паролей и т.д.), стоимость использования (например, шифрование и дешифрование требуют затрат времени), стоимость уменьшения легкости использования системы (как было показано ранее). С другой стороны, существует много возможных ущербов от нарушения политики безопасности: потеря приватности информации (например, прочтение информации посторонними лицами), потери данных (например, повреждение и уничтожение данных), утрата определенного сетевого сервиса (например, заполнение дискового пространства файлового сервера, использование процессорного времени посторонними людьми, или отказ в обслуживании сети). Стоимость каждого типа защиты должна соотносится с возможными потерями.

Ваши цели в области безопасности должны быть доведены до каждого работника, обслуживающего персонала и управленческого персонала с помощью набора формальных правил, называемых «политика безопасности». Здесь и в дальнейшем мы будем использовать этот термин вместо «политика безопасности компьютерных систем», так как на самом деле, политика безопасности касается всех видов информационных технологий в организации и все видов информации так или иначе касающейся деятельности организации и поддерживаемой информационными технологиями.

4.2.2. Определение Политики безопасности

Политика безопасности – это формальное выражение правил, согласно которым обязан действовать персонал, имеющий доступ к информационным ресурсам организации.

4.2.3. Цели политики безопасности.

Основная цель Политики безопасности – проинформировать пользователей, персонал и управляющих о требованиях, которые предъявляются к ним для достижения целей защиты оборудования и информации. Кроме того, Политика должна описать механизмы, с помощью которых эти требования могут быть выполнены. Другая цель Политики безопасности – сформулировать основную линию, согласно которой должна проводиться установка, настройка и аудит компьютерных систем и сетей. Следовательно, использование любого набора инструментов для обеспечения безопасности, при отсутствии четко сформулированной и высказанной политики безопасности не представляется возможным.

Правила допустимого использования (ПДИ) сетевых ресурсов так же могут являться частью общей политики безопасности. Следует четко определить, что пользователи могут, а чего не могут делать, используя сетевые ресурсы, включая так же и отдельные виды сетевого трафика. Эти Правила должны быть сформулированы предельно четко и точно, чтобы исключить возможность непонимания или неверного толкования их пользователями. Например, при использовании групп новостей USENET, Правила допустимого использования могут включать в себя полный список тех групп, пользование которыми разрешено.

4.2.4. Кто должен принимать участие в формировании Политики безопасности?

Для того, чтобы Политика безопасности была разумной и эффективной, необходимо участие в ее разработке на всех уровнях в организации и повсеместная ее поддержка. Особенно важной является поддержка политики безопасности на уровне руководства, поскольку в противном случае Политика безопасности, вероятнее всего, окажется просто документом, не имеющим никаких шансов на воплощение. В процессе работы над Политикой безопасности целесообразно привлечь к работе следующих людей:

1. Администратора безопасности организации и/или сети
2. Специалистов по информационным технологиям (то есть специалистов отдела АСУ и/или информатизации)
3. Административных работников, в подчинении которых находится большое число пользователей (например, начальники отделов)
4. Группу реагирования на случаи нарушения режима безопасности
5. Представителей групп пользователей, на которых в дальнейшем будет распространяться действие Политики безопасности
6. Руководителей предприятия или организации
7. Юридических консультантов

Представленный список, конечно же, не является жестким требованием. С учетом специфики того или иного предприятия он может изменяться. В целом же, он должен включать представителей всех сторон, которых будет так или иначе касаться политика безопасности, а так же руководителей, которые будут выделять средства на реализацию политики. Такой подход представляется разумным и, вероятно, единственно верным, чтобы получить в дальнейшем поддержку на всех уровнях предприятия.

4.2.5. Что такое хорошая политика безопасности?

Вот несколько ключевых моментов, отличающих правильно сформированную политику безопасности:

1. Она должна быть реализуема с помощью процедур и средств системного администрирования, путем публикации правил допустимого использования сетевых ресурсов, или другими методами, приемлемыми для данной организации.
2. Она должна поддерживаться в максимальной степени, где это только возможно, автоматизированными средствами обеспечения безопасности, или санкциями к нарушителям, там, где применение автоматизированных средств защиты невозможно.
3. Она должна четко определять области и степени ответственности пользователей, администраторов сетей, серверов и сервисов, а так же управленческого персонала.

Как документ, типичная Политика безопасности может содержать:

1. Руководство по закупке компьютерного и информационного оборудования, которое определяет требуемые или желательные средства обеспечения безопасности, которыми должно обладать закупаемое оборудование или программное обеспечение.
2. Правила безопасности, которые определяют разумное ограничение прав пользователей на тайну. Например, регистрация и анализ команд, подаваемых пользователем, учет электронной почты, учет доступа к файлам и т.д.

3. Правила доступа, которые определяют ограничение права доступа и привилегий пользователей, для защиты данных и оборудования от потерь, повреждения или уничтожения. А так же определение допустимого использования сетевых ресурсов и сервисов для пользователей и обслуживающего персонала. Так же в этом разделе должны содержаться правила, описывающие внешние соединения и их использование, обмен данными, подключение новых устройств и ресурсов к сети, а так же установку нового программного обеспечения. Здесь же желательно специфицировать уведомления (например, сообщение при соединении с удаленной машиной должно предупреждать о необходимости авторизованного использования и возможности наблюдения за линией, а не просто содержать строчку “Welcome to”).
4. Правила Учета – определяют ответственность пользователей, администраторов и руководящего персонала. В данном документе необходимо определить возможности учета и аудита, и обеспечивать руководство по предотвращению случаев нарушения режима безопасности (например, с кем следует связаться и взаимодействовать в случае подозрений на попытку нарушения режима безопасности)
5. Правила аутентификации. Эти правила устанавливают степень доверия методам аутентификации, путем, например, предъявления требований к процедуре установления пароля и самому паролю. Кроме того, здесь же необходимо описать правила доступа и использования устройств аутентификации (например, генераторов одноразовых паролей).
6. Условия доступности. Определяют тот уровень доступности сетевых ресурсов, который пользователи вправе ожидать. Здесь необходимо определить такие моменты, как резервирование, восстановление после возможных сбоев, а так же часы гарантированной работы оборудования и возможные перерывы на тех. обслуживание – их расписание и продолжительность.
7. Правила обслуживания сети и информационной системы в целом. Описывают, в какой степени собственный и внешний обслуживающий персонал имеет доступ к данным и технологиям, обеспечивающим работу предприятия. Особенно важно четко определить, допустимо ли удаленное управление сетевыми ресурсами внешним персоналом, и каким образом отслеживается тот уровень доступа, который имеют люди, не состоящие в организации, но выполняющие те или иные работы по обслуживанию.
8. Правила сообщения о нарушениях. Эти правила описывают, о каких именно типах нарушений правил безопасности следует сообщать и кто должен это делать. Отметим, что в условиях невысокого риска и возможности анонимного сообщения о замеченных нарушениях работниками предприятия, существует высокая вероятность того, что о замеченных нарушениях будет сообщено сразу же.
9. Информация о поддержке. В этом разделе политики безопасности указывается, с кем следует связаться в случае нарушения политики безопасности, каким образом определить такую попытку, какую информацию следует считать конфиденциальной или частной, а так же приводятся ссылки на соответствующие процедуры безопасности, пункты правил безопасности предприятия и, быть может, законы, относящиеся к этой области.

Возможно, что в рамках политики безопасности потребуются некоторые регуляторы (например, наблюдение за коммуникационными линиями). Создатели политики безопасности должны постоянно иметь в виду законность тех или иных методов обеспечения

безопасности. Как минимум, с политикой безопасности должен ознакомится юрист предприятия или консультант.

После того, как политика безопасности будет создана, проверена юристом и одобрена руководством предприятия, ее необходимо довести до сведения и понимания каждого работника. Будет разумно, если каждый работник распишется в том, что он прочитал, понял и обязуется соблюдать правила политики безопасности в своей повседневной работе. Наконец, политика безопасности должна периодически пересматриваться на предмет соответствия меняющимся целям и задачам предприятия, а так же новым технологиям, которые, возможно, появятся.

4.2.6. Поддержание гибкости политики

Для того чтобы политика безопасности и правила безопасности не устарели и могли с успехом использоваться в течение долгого времени, необходимо обеспечить гибкость уже в самой архитектуре безопасности. Что это означает? Политика безопасности должна быть максимально независимой от конкретного оборудования и программного обеспечения (насколько то или иное оборудование может быть заменено каким-то иным). Механизмы поддержания актуальности и обновления политики и правил безопасности должны быть сформулированы просто и четко. В частности, должны быть назначены люди, работающие над поддержанием актуальности политики безопасности и люди, в чьи полномочия входит утверждать те или иные обновления политики безопасности.

Важно так же понимать, что для каждого положения и правила безопасности могут быть и исключения. Более того, желательно, чтобы политика безопасности описывала, какого рода исключения допустимы. Например, при каких условиях системному администратору допустимо иметь доступ к файлам пользователей. Так же, возможны условия, при которых пользователи могут иметь доступ к данным под одним и тем же регистрационным именем. Например, на машине с пользователем-администратором “root” несколько администраторов могут знать пароль и использовать регистрационную запись “root” для управления системой или сервисами.

Другая опасность называется «синдром грузовика с мусором». Что это такое? В том случае, если в организации или на предприятии какая-либо ключевая фигура внезапно окажется не способной исполнять свои должностные обязанности (например, по болезни), а в организации принята очень жесткая политика безопасности, возникает риск потери критически важных данных, поскольку информация не может быть доступна другим пользователям. Так происходит до тех пор, пока ограничение распространения и использования информации требуется по соображениям безопасности. При формировании политики безопасности очень важно определить правильный баланс между этими требованиями и опасностями.

4.2.7.Пример политики безопасности

Чтобы сделать изложение более конкретным, рассмотрим вслед за [11] гипотетическую локальную сеть, которой владеет организация XYZ, и приведем пример политики безопасности среднего уровня. Дабы пример не выглядел вырванным из контекста, в него добавлены некоторые положения политики верхнего уровня.

Описание аспекта. Информация, циркулирующая в рамках локальной сети, является критически важной. Локальная сеть позволяет пользователям разделять программы и данные; это увеличивает риск. Следовательно, каждый из компьютеров, входящих в сеть, нуждается в более сильной защите, чем отдельная машина. Эти повышенные меры безопасности и являются предметом данного документа.

Документ преследует две главные цели - продемонстрировать сотрудникам XYZ важность защиты сетевой среды и описать их роль в обеспечении безопасности, а также распределить конкретные обязанности по защите информации, циркулирующей в сети, равно как и самой сети.

Область применения. В сферу действия данной политики попадают все аппаратные, программные и информационные ресурсы, входящие в локальную сеть предприятия. Политика ориентирована также на людей, работающих с сетью, в том числе на пользователей, субподрядчиков и поставщиков.

Позиция организации. Целью организации XYZ является обеспечение целостности, доступности и конфиденциальности данных, а также их полноты и актуальности. Более частными целями являются:

- Обеспечение уровня безопасности, соответствующего нормативным документам.
- Следование экономической целесообразности в выборе защитных мер (расходы на защиту не должны превосходить предполагаемый ущерб от нарушения информационной безопасности).
- Обеспечение безопасности в каждой функциональной области локальной сети.
- Обеспечение подотчетности всех действий пользователей с информацией и ресурсами.
- Обеспечение анализа регистрационной информации.
- Предоставление пользователям достаточной информации для сознательного поддержания режима безопасности.
- Выработка планов восстановления после аварий и иных критических ситуаций для всех функциональных областей с целью обеспечения непрерывности работы сети.
- Обеспечение соответствия с имеющимися законами и общеорганизационной политикой безопасности.

Роли и обязанности (общие положения). Следующие группы людей отвечают за реализацию сформулированных выше целей. Детально их обязанности будут описаны ниже.

- Руководители подразделений. Они отвечают за доведение положений политики безопасности до пользователей и за контакты с пользователями.
- Администраторы локальной сети. Они обеспечивают непрерывное функционирование сети и отвечают за реализацию технических мер, необходимых для проведения в жизнь политики безопасности.
- Администраторы сервисов. Они отвечают за конкретные сервисы и, в частности, за то, что их защита построена в соответствии с общей политикой безопасности.
- Пользователи. Они обязаны использовать локальную сеть в соответствии с политикой безопасности, подчиняться распоряжениям лиц, отвечающих за отдельные аспекты безопасности, ставить в известность руководство обо всех подозрительных ситуациях.

Законопослушность. Нарушение политики безопасности может подвергнуть локальную сеть и циркулирующую в ней информацию недопустимому риску. Случаи нарушения со стороны персонала будут рассматриваться руководством для принятия мер вплоть до увольнения.

Роли и обязанности (детальное изложение).

Руководители подразделений обязаны:

- Постоянно держать в поле зрения вопросы безопасности. Следить за тем, чтобы то же делали их подчиненные.
- Проводить анализ рисков, выявляя активы, требующие защиты, и уязвимые места систем, оценивая размер возможного ущерба от нарушения режима безопасности и выбирая эффективные средства защиты.
- Организовать обучение персонала мерам безопасности. Обратить особое внимание на вопросы, связанные с антивирусным контролем.
- Информировать администраторов локальной сети и администраторов сервисов об изменении статуса каждого из подчиненных (переход на другую работу, увольнение и т.п.).
- Обеспечить, чтобы каждый компьютер в их подразделениях имел хозяина или системного администратора, отвечающего за его безопасность и имеющего достаточную квалификацию для выполнения этой роли.

Администраторы локальной сети обязаны:

- Информировать руководство об эффективности существующей политики безопасности и о технических мерах, которые могут улучшить защиту.
- Обеспечить защиту оборудования локальной сети, в том числе интерфейсов с другими сетями.
- Оперативно и эффективно реагировать на события, таящие угрозу. Информировать администраторов сервисов о попытках нарушения защиты. Оказывать помощь в отражении угрозы, выявлении нарушителей и предоставлении информации для их наказания.
- Использовать проверенные средства аудита и обнаружения подозрительных ситуаций.
- Ежедневно анализировать регистрационную информацию, относящуюся к сети в целом и к файловым серверам в особенности.
- Следить за новинками в области информационной безопасности, информировать о них пользователей и руководство.
- Не злоупотреблять своими полномочиями. Пользователи имеют право на тайну.
- Разработать процедуры и подготовить инструкции для защиты локальной сети от зловредного программного обеспечения. Оказывать помощь в обнаружении и ликвидации зловредного кода.
- Регулярно выполнять резервное копирование информации, хранящейся на файловых серверах.
- Выполнять все изменения сетевой аппаратно-программной конфигурации.

- Гарантировать обязательность процедуры идентификации и аутентификации для доступа к сетевым ресурсам.
- Выделять пользователям входные имена и начальные пароли только после заполнения регистрационных форм.
- Периодически производить проверку надежности защиты локальной сети. Не допускать получения привилегий неавторизованными пользователями.

Администраторы сервисов обязаны:

- Управлять правами доступа пользователей к обслуживаемым объектам.
- Оперативно и эффективно реагировать на события, таящие угрозу. Информировать администраторов локальной сети о попытках нарушения защиты. Оказывать помощь в отражении угрозы, выявлении нарушителей и предоставлении информации для их наказания.
- Регулярно выполнять резервное копирование информации, обрабатываемой сервисом.
- Выделять пользователям входные имена и начальные пароли только после заполнения регистрационных форм.
- Ежедневно анализировать регистрационную информацию, относящуюся к сервису.
- Регулярно контролировать сервис на предмет зловредного программного обеспечения.
- Периодически производить проверку надежности защиты сервиса. Не допускать получения привилегий неавторизованными пользователями.

Пользователи обязаны:

- Знать и соблюдать законы, правила, принятые в XYZ, политику безопасности, процедуры безопасности.
- Использовать доступные защитные механизмы для обеспечения конфиденциальности и целостности своей информации.
- Использовать механизм защиты файлов и должным образом задавать права доступа.
- Выбирать хорошие пароли, регулярно менять их. Не записывать пароли на бумаге, не сообщать их другим лицам.
- Помогать другим пользователям соблюдать меры безопасности. Указывать им на замеченные упущения с их стороны.
- Информировать администраторов или руководство о нарушениях безопасности и иных подозрительных ситуациях.
- Не использовать слабости в защите сервисов и локальной сети в целом.
- Не совершать неавторизованной работы с данными, не создавать помех другим пользователям.
- Всегда сообщать корректную идентификационную и аутентификационную информацию, не пытаться работать от имени других пользователей.
- Обеспечивать резервное копирование информации с жесткого диска своего компьютера.
- Знать принципы работы зловредного программного обеспечения, пути его проникновения и распространения, слабости, которые при этом могут использоваться.

- Знать и соблюдать процедуры для предупреждения проникновения зловредного кода, для его обнаружения и уничтожения.
- Знать слабости, которые используются для неавторизованного доступа.
- Знать способы выявления ненормального поведения конкретных систем, последовательность дальнейших действий, точки контакта с ответственными лицами.
- Знать и соблюдать правила поведения в экстренных ситуациях, последовательность действий при ликвидации последствий аварий.

4.3. Управление персоналом

Управление персоналом начинается с приема нового сотрудника на работу и даже раньше - с составления описания должности. Уже на этом этапе желательно привлечение специалиста по информационной безопасности для определения компьютерных привилегий, ассоциируемых с должностью. Существует два общих принципа, которые следует иметь в виду:

- разделение обязанностей,
- минимизация привилегий.

Принцип разделения обязанностей предписывает так распределять роли и ответственность, чтобы один человек не мог нарушить критически важный для организации процесс. Например, нежелательна ситуация, когда платежи от имени организации выполняет один человек. Надежнее поручить одному сотруднику оформлять заявки на платежи, а другому - заверять эти заявки.

Принцип минимизации привилегий предписывает выделять пользователям только те права доступа, которые необходимы им для выполнения служебных обязанностей. Назначение этого принципа очевидно – уменьшить ущерб от случайных или умышленных некорректных действий пользователей.

Предварительное составление описания должности позволяет оценить ее критичность и спланировать процедуру проверки и отбора кандидатов. Чем критичнее должность, тем тщательнее нужно проверять кандидатов: навести о них справки, быть может, побеседовать с бывшими сослуживцами и т.д. Подобная процедура может быть длительной и дорогой, поэтому нет смысла усложнять ее сверх необходимого. В то же время неразумно и совсем отказываться от предварительной проверки, рискуя принять на работу человека с уголовным прошлым или с душевными болезнями.

Когда кандидат отобран, он, вероятно, должен пройти обучение; по крайней мере, его следует подробно ознакомить со служебными обязанностями, а также с нормами и процедурами информационной безопасности. Желательно, чтобы меры безопасности были им усвоены до вступления в должность и до заведения его учетной записи с входным именем, паролем и привилегиями.

С момента заведения учетной записи начинается ее администрирование, а также протоколирование и анализ действий пользователя. Постепенно изменяется окружение, в котором работает пользователь, его служебные обязанности и т.п. Все это требует соответствующего изменения привилегий. Техническую сложность составляют временные перемещения сотрудника, выполнение им обязанностей взамен лица, ушедшего в отпуск, и иные обстоятельства, когда полномочия нужно сначала дать, а через некоторое время взять

обратно. В такие периоды профиль активности пользователя резко меняется, что создает трудности при выявлении подозрительных ситуаций. Определенную аккуратность следует соблюдать и при выдаче новых постоянных полномочий, не забывая изымать старые права доступа.

Ликвидация учетной записи пользователя, особенно в случае конфликта между сотрудником и организацией, должна производиться максимально оперативно (в идеале - одновременно с извещением о наказании или увольнении). Возможно и физическое ограничение доступа к рабочему месту. Разумеется, если сотрудник увольняется, у него нужно принять все его компьютерное хозяйство и, в частности, криптографические ключи, если использовались средства шифрования.

К управлению сотрудниками примыкает администрирование лиц, работающих по контракту (например, специалистов фирмы-поставщика, помогающих запустить новую систему). В соответствии с принципом минимизации привилегий, им нужно выделить ровно столько прав, сколько необходимо, и изъять эти права сразу по окончании контракта. Проблема, однако, в том, что на начальном этапе внедрения "внешние" сотрудники будут администрировать "местных", а не наоборот. Здесь на первый план выходит квалификация персонала организации, его способность быстро обучаться, а также оперативное проведение учебных курсов. Важны и принципы выбора деловых партнеров.

Иногда внешние организации принимают на обслуживание и администрирование ответственные компоненты компьютерной системы, например, сетевое оборудование. Нередко администрирование выполняется в удаленном режиме. Вообще говоря, это создает в системе дополнительные слабости, которые необходимо компенсировать усиленным контролем средств удаленного доступа или, опять-таки, обучением собственных сотрудников.

Мы видим, что проблема обучения - одна из центральных с точки зрения информационной безопасности. Если сотрудник не знаком с политикой безопасности своей организации, он не может стремиться к достижению сформулированных в ней целей. Если он не знает мер безопасности, он не сможет их соблюдать. Напротив, если сотрудник знает, что его действия протоколируются, он, возможно, воздержится от нарушений.

Из педагогической психологии известно: чтобы обучение было эффективным, ему должен предшествовать этап мотивации. Сотрудникам необходимо объяснить, зачем нужна учеба, зачем нужны меры безопасности. Обычно и то, и другое вызывает раздражение, поскольку мешает основной деятельности. Важно, чтобы сотрудники смотрели на вещи шире, имея в виду долговременные интересы организации и свои собственные.

Обучение должно проводиться регулярно и в то же время каждый раз по-новому, иначе оно превратится в формальность и потеряет эффективность. К сожалению, здесь не существует общих рецептов, все зависит от изобретательности организаторов.

4.4. Защита локальной сети

Существует несколько проблем, из-за которых сети могут стать слабым местом в защите информационной системы. Классическая проблема – атака класса «отказ в обслуживании». Этот тип атак заключается в приведении сети в состояние, неспособности обмениваться данными с авторизованными, законными пользователями. Существует 2 основных пути, каким образом это может быть сделано: атакой на маршрутизаторы, или

«забиванием» канала связи бесполезным, но интенсивным трафиком. Отметим, что понятие «маршрутизатор» в данном контексте обозначает очень широкий спектр оборудования, который может включать такие компоненты как брандмауэры или, например, сервисы-посредники.

Целью атаки на маршрутизатор является приведение его в такое состояние, в котором он перестает пересылать пакеты между интерфейсами, или делает это неправильно. Такое может произойти по нескольким причинам: неверная конфигурация, посылка пакетов с неверными обновлениями маршрутной информации, или бомбардировка маршрутизатора пакетами, пути для которых он не способен определить. «Забивание» полосы пропускания сети аналогично бомбардировке маршрутизатора. Идеальная атака такого типа – генерация такого пакета, который будет циркулировать в сети по кругу, между несколькими узлами. Хорошо построенная атака такого типа вызывает экспоненциальный рост числа передаваемых пакетов в сети, и способна полностью блокировать полезную работу.

Другая классическая проблема, связанная с сетями, известна как spoofing. В этом случае маршрутизаторам передается неверное обновление маршрутной информации, в результате чего они составляют неверные маршруты для пакетов. Отличие этой атаки от «отказа в обслуживании» только в целях нарушения маршрутов. «Отказ в обслуживании» полностью блокирует работу маршрутизатора, и это состояние может быть легко определено обслуживающим персоналом. В случае же спуфинга, маршрутизатор может направлять пакеты на тот адрес, где атакующий может произвести анализ содержимого пакета, после чего пакеты передаются настоящему адресату. Однако атакующий при этом имеет возможность изменять содержимое пакета по своему усмотрению.

Решение многих из этих проблем в том, чтобы защитить пакеты с маршрутной информацией каким-либо образом: паролем, электронной подписью, шифрованием. Пароль дает самый минимум защиты – только от тех атакующих, кто не может получить доступ к локальной сети. Кроме того, пароли являются защитой и от ошибок конфигурирования. Преимуществом парольной защиты является очень низкая дополнительная загрузка процессоров маршрутизаторов и полосы пропускания сети. Электронная подпись защищает от посылки пакетов с неверными обновлениями маршрутной информации, даже если атакующий имеет физический доступ к локальной сети. В комбинации с номером последовательности или другим уникальным идентификатором электронная подпись может защитить от повтора пакета атакующим, но уже с другими обновлениями маршрутной информации. Наконец, наибольшая защита обеспечивается полным шифрованием обновлений маршрутной информации. Это не только защищает от подмены пакетов атакующим, но так же позволяет скрыть внутреннюю топологию сети. Недостатком этого метода является то, что он требует серьезных затрат процессорного времени на шифрование и дешифрование информации.

Протоколы маршрутизации RIP-2 и OSPF стандартно поддерживают простые текстовые пароли. Дополнительно есть расширения для поддержки шифрования по алгоритму MD5.

К сожалению, в настоящее время нет достаточной защиты от атак путем «забивания» полосы пропускания сети. Однако этот тип атак легко обнаруживается и ликвидируется администраторами.

Защита сети необходима и от возможности «сниффинга». Что это такое? Это возможность перехвата пакетов локальной сети с целью анализа трафика. Sniffing (в переводе с английского «вынюхивание») может быть крайне опасен для сети и сетевых сервисов. В простейшем случае, возможен сбор паролей, передаваемых в незашифрованном виде, например, сервису telnet. Для защиты от сниффинга необходимо обеспечить защиту всех точек подключения к сети, всего оборудования аппаратных и этажных распределительных шкафов. Хорошую помощь в этом может оказать современное сетевое оборудование. Например, система безопасности BaySecure, встроенная в маршрутизаторы и коммутаторы BayNetworks, позволяет определить с каких сетевых карт коммутатор должен обрабатывать пакеты, а с каких – нет.

Даже просто применение коммутатора для передачи сетевых пакетов резко снижает или исключает возможности сниффинга. При использовании коммутаторов трафик передается только между источником и получателем. Таким образом, произвольная машина не может осуществить прослушивание трафика. Этот достаточно простой метод позволяет избежать многих опасностей сниффинга.

Наконец, существует еще одна опасность, связанная с локальными сетями предприятия. Это опасность перехвата побочных излучений в сетях. Поскольку реально перехват излучения, например, мониторов, не дает доступа к конфиденциальной информации, а перехват излучений процессорных плат или плат ввода-вывода практически не подлежит анализу, то перехват и анализ излучения таких компонент локальной сети, как, например, кабели, достаточно несложен и эффективен. Более того, располагая информацией о строении сигнала (а это описано в соответствующих стандартах), можно без труда и с небольшими затратами осуществить перехват и анализ всего сетевого трафика.

Конечно, чтобы кто-то был заинтересован в привлечении дополнительного оборудования, в сети должны циркулировать очень важные данные. Тем не менее, опасность остается. Избежать ее можно либо тотальным шифрованием всего трафика (что, очевидно, нереально), либо использованием средств физической защиты кабелей – металлические кабельные желоба с заземлением и экранированные сетевые кабели, так же с заземлением. При этом все оборудование и кабельное хозяйство должно иметь общие точки заземления. Заземляются так же аппаратные шкафы, стены распределительных комнат или шкафов, распределительные панели и корпуса компьютеров или терминалов.

И еще несколько слов о физической защите. Как это уже упоминалось выше, все комнаты и оборудование, обслуживающие локальную сеть (серверные фермы, коммутационные комнаты, аппаратные шкафы и т.д.), являются жизненно важными для обеспечения безопасности сети. Поэтому доступ к ним должен быть физически ограничен – замки на комнатах, замки на шкафах, возможно использование пломб. При этом доступ разрешается только очень ограниченному числу администраторов сети. Администраторы сервисов вообще не должны иметь доступа к таким помещениям и аппаратуре. Все подключения и коммутация кабельного хозяйства должны быть строго задокументированы. Документированию подлежит так же подключения и настройка портов активного сетевого оборудования. Этот несложный комплекс мер позволит обеспечить надежное функционирование сети не только в плане безопасности.

4.5. Защита сервисов

Существует очень большое количество разнообразных сервисов, и каждый имеет свои собственные требования по обеспечению безопасности. Во многом эти требования обусловлены тем, как будет использоваться тот или иной сервис. Сервисы, которые используются только внутри локальной сети предприятия (например, NFS), возможно, потребуют иных механизмов защиты, нежели сервисы, предназначенные для использования так же вне локальной сети. Возможно, что для обеспечения безопасности внутренние сервисы вообще необходимо будет ограничить от какого-либо доступа извне. И это может быть сделано внешними защитными средствами. Напротив, www-сервер, хранящий домашние странички пользователей и предназначенный обслуживать всех пользователей Интернет, должен иметь хорошую встроенную защиту, так как возможности «прикрыть» его какими-то внешними средствами ограничены.

Внутренние сервисы (то есть сервисы, используемые исключительно внутри сети предприятия) и внешние сервисы (то есть сервисы, предназначенные для обслуживания пользователей извне), в общем, будут иметь примерно такие отличия в требованиях к защите, как это было описано выше. Следовательно, логично и разумно стремиться разделить внутренние и внешние сервисы на различных серверах. Это означает, что внутренние и внешние сервисы никогда не должны предоставляться одними и теми же серверами. На деле, в большинстве случаев желательно (если это возможно) не просто разделить сервисы, но разделить подсети – подсеть с возможностью доступа извне, содержащую внешние сервисы, и подсеть без доступа извне, содержащую только внутренние сервисы. Обычно для соединения этих подсетей используются брандмауэры. Особое внимание в такой конфигурации требует настройка и тестирование брандмауэра, поскольку в огромной степени защита внутренних сервисов зависит от его правильного функционирования.

В настоящее время все больше и больше интереса проявляется в области использования интрасетей для объединения различных частей предприятий или организаций (например, отделов). В данной работе в основном проводится деление сервисов на 2 группы – внутренние и внешние. Однако в сетях с использованием технологий интранет необходимо иметь три уровня деления. Это обусловлено тем, что сервисы, обслуживающие интранет, с одной стороны, не являются внешними, поскольку доступ к ним ограничивается пределами организации. С другой стороны, эти сервисы не являются и внутренними, поскольку их использование не ограничено пределами одной административной единицы. Следовательно, для поддержки этих сервисов потребуются отдельные системы.

Некоторые внешние сервисы используют особые условия доступа – анонимный или гостевой доступ. Это может быть анонимный вход на FTP или гостевой вход в систему (без пароля). Для общей безопасности системы крайне важно, чтобы такие анонимные пользователи были максимально изолированы в системе. Необходимо, чтобы файловые системы, используемые ими, были отделены от остальных файловых систем. Особого внимания требует предоставление таким пользователям права на запись, поскольку владелец сервиса может нести ответственность за информацию, хранимую на его серверах, даже если она была там размещена анонимным пользователем. Следовательно, необходимо тщательно следить за тем, какого рода информацию записывают анонимные пользователи.

Теперь рассмотрим некоторые наиболее популярные сервисы: сервис имен, сервис аутентификации, сервис-посредник, электронную почту, WWW, FTP и, наконец, сетевую файловую систему NFS. Поскольку именно эти сервисы являются наиболее часто используемыми, именно они могут в первую очередь подвергнуться атаке.

4.5.1. Сервис имен (DNS, NIS, NIS+)

Сеть Интернет использует в своей работе DNS для выполнения разрешения адресов по сетевым именам машин. Сетевая служба имен NIS или NIS+ не используется в Интернет, но подвержена таким же рискам, как и DNS. Преобразование имен в физические адреса крайне важно для обеспечения безопасности сети. Атакующий, получив контроль над сервисом DNS, может перенаправить трафик сети, и таким образом обойти ограничения, накладываемые защитой сети. Например, сетевой трафик может быть перенаправлен на скомпрометированную машину, где подвергнется анализу.

Исторически так сложилось, что сервисы DNS не имели никаких средств защиты безопасности. В частности, информация не проверялась на предмет обнаружения модификаций, кроме того, не существовало проверок, что запрос пришел именно от того сервера, который запрашивался, а не откуда-то еще. Была продела большая работа по внедрению механизма электронной подписи в протокол DNS, и теперь существует возможность проверки целостности и правильности информации с использованием этого механизма.

4.5.2. Сервис проверки ключей/паролей (NIS, NIS+)

Как правило, серверы ключей и/или паролей осуществляют защиту жизненно важной информации (ключей и паролей) алгоритмами шифрования. Однако даже алгоритмы шифрования без обратного хода могут быть подвержены атакам с использованием словарей. Для уменьшения риска атаки необходимо убедиться, что сервисы проверки ключей и паролей никаким образом недостижимы с тех машин сети, которые не будут пользоваться этими сервисами. Кроме того, необходимо, чтобы серверы, на которых размещены сервисы проверки паролей и ключей не поддерживали никаких прочих сервисов (например, telnet или FTP).

4.5.3. Сервисы аутентификации и сервисы-посредники

Большинство сервисов-посредников имеют достаточно большие средства обеспечения безопасности. Главное их предназначение – сконцентрировать весь трафик, идущий к внешним сервисам таким образом, чтобы он проходил через одну точку сети (например, через один сервер). Это позволяет достаточно легко осуществлять слежение за трафиком, ограничивать доступ, а так же скрывать внутреннюю структуру сети. Однако такой сервис-посредник в первую очередь находится под угрозой внешнего вторжения. Именно он более всего подвержен атакам. Какой тип сервера-посредника выбрать и какой вид защиты можно с его помощью обеспечить, зависит от тех протоколов, которые используются в сети и от тех сервисов, какие необходимо защитить. Обычные правила ограничения – разрешить доступ к сервису только тем адресам, которым он необходим, а так же разрешить доступ с того или иного адреса только к тем сервисам, которые ему необходимы.

4.5.4. Сервис электронной почты

Сервисы электронной почты в течение долгого времени служили основными объектами атак. Происходило это оттого, что сервис электронной почты – один из старейших и широко распространенных в мире. Кроме того, совершенно естественно, что серверу электронной почты необходим доступ не только внутри сети, но и вне ее. Большинство сервисов электронной почты настроены таким образом, что могут принимать сообщения с любого адреса сети. Типичное строение сервиса электронной почты включает в себя две части – приемник/отправитель почты и обработчик почты. Поскольку электронная почта доставляется всем пользователям системы (как правило) и, кроме того, является конфиденциальной, то обработчику почты необходимо иметь высший приоритет в системе (root – в UNIX-системах), чтобы он мог осуществлять доставку почты. Многие реализации сервиса электронной почты построены таким образом, что часть, отвечающая за прием-передачу почты, так же требует высших привилегий. Это ведет к появлению еще нескольких слабых мест в защите, детальное рассмотрение которых не входит в задачи этого документа. Некоторые реализации сервиса электронной почты позволяют разделить части доставки и обработки почты. С точки зрения безопасности, они являются более предпочтительными, однако все равно, при их установке и настройке следует соблюдать крайнюю осторожность.

4.5.5. WWW-сервисы

Популярность «всемирной паутины» (WWW) растет в последнее время по экспоненте, поскольку именно этот тип сервиса сочетает в себе удобство и легкость использования с возможностью представления различных видов информации. Большинство реализаций сервисов WWW позволяют лицам, имеющим к ним доступ, выполнять какие-то действия. Наиболее общим примером являются так называемые cgi-скрипты, которые позволяют удаленному пользователю запустить на сервере, содержащем WWW-сервис, какую-то программу, передав ей определенные аргументы в запросе. Многие, если не большинство, таких программ создаются без учета или тщательной проверки безопасности их использования и, следовательно, могут быть причиной нарушения безопасности. Если сервис WWW должен быть доступен всему сообществу Интернет, необходимо особо следить за тем, чтобы ни конфиденциальная, ни частная информация не находилась на сервере, на котором действует WWW-сервис. Более того, в случае необходимости предоставления какой-либо информации в WWW, необходимо иметь отдельный, выделенный сервер для обслуживания сервиса WWW. Причем, этот сервер не должен являться доверенным адресом для других сервисов, расположенных во внутренней сети.

Во многих случаях может казаться удобным содержать на одном сервере сервисы WWW и FTP. Это возможно, но только в том случае, если сервис FTP обеспечивает только чтение файлов, но не запись. Сочетание анонимного доступа к FTP с возможностью записи и сервиса WWW на одном сервере может привести к нарушению защиты системы (например, результатом может быть модификация информации, представляемой сервисом WWW)

4.5.6. Сервисы передачи файлов

Сервисы FTP и TFTP позволяют пользователям принимать и передавать файлы в режиме точка-точка. Однако если FTP требует авторизации пользователей, то TFTP – нет. По этой причине, TFTP необходимо запретить во всех случаях, где это возможно.

Неверно установленный и/или настроенный сервис может позволить атакующему копировать, заменять или удалять файлы на том сервере, который поддерживает неверно установленный сервис. Следовательно, важно обратить особое внимание на настройку этого сервиса. Доступ через FTP к файлам, содержащим пароли или приватные данные, возможность внедрения «троянских коней» - все это гораздо более серьезные угрозы безопасности, которые могут возникнуть вследствие неверной конфигурации FTP. В любом случае, FTP-сервисы следует размещать на собственных серверах. Администраторы некоторых сетей предпочитают размещать публичные WWW и FTP сервисы на одном и том же сервере, объясняя это тем, что оба сервиса, в общем, имеют одинаковые требования относительно безопасности. В реальной же практике такая стратегия применима, только если публичный FTP-сервис не предоставляет возможности записи. Если же предполагается анонимный доступ к FTP -сервису с правом на запись, необходимо размещать сервисы FTP и WWW на разных серверах. Еще раз отметим, что так же нельзя размещать сервисы, предназначенные для внутреннего и внешнего использования, на одном и том же сервере.

Протокол TFTP не поддерживает все то многообразие функций, которое обеспечивает FTP, вместе с тем, TFTP не поддерживает и каких-либо мер обеспечения безопасности. Если же использование TFTP по каким-то причинам необходимо, тогда использование этого сервиса должно строго ограничиваться только рамками локальной сети предприятия. Соответственно, сервис TFTP необходимо настроить таким образом, чтобы он мог получить доступ только к заранее определенному набору файлов. Вероятно, одно из немногих оправданных применений TFTP состоит в загрузке на маршрутизаторы конфигурационных файлов. Наконец, TFTP-сервисы должны располагаться только на внутренних серверах, то есть на этих серверах не может быть сервисов FTP или WWW с общим доступом.

4.5.7. Сетевая файловая система (NFS)

Сетевая файловая система позволяет различным системам разделять дисковые ресурсы. Очень часто NFS применяется на бездисковых системах, использующих файловые серверы для хранения всех необходимых файлов. К сожалению, столь широко используемый сервис не имеет встроенных средств обеспечения безопасности. Следовательно, необходимо ограничить использование сервиса NFS только теми системами, которые реально нуждаются в нем. Это обеспечивается путем указания какие именно машины и с какими правами могут пользоваться экспортируемыми по NFS файловыми системами. Файловые системы не могут быть экспортированы для систем и машин, расположенных вне сети предприятия, поскольку это означает возможность доступа к сервису NFS снаружи. В идеале, внешний доступ к NFS должен ограничиваться с помощью брандмауэра.

4.5.8. Защита защиты

Сложно сказать, насколько часто администраторы оставляют без внимания наиболее важную проблему обеспечения безопасности, оставляя сервисы обеспечения безопасности без должной защиты. Основываясь на соображениях, изложенных выше, совершенно ясно что: сервис обеспечения безопасности не может и не должен быть доступен извне сети предприятия. Сервер, на котором располагается сервис безопасности, должен предоставлять минимальный набор прочих сервисов. В идеале в сети должен быть выделенный сервер

безопасности. Наконец, любая попытка доступа к серверу, на котором работает сервис обеспечения безопасности, включая доступ к самому этому сервису, должна регистрироваться в системных журналах.

4.6. Брандмауэр

Один из наиболее широко распространенных и известных сервисов обеспечения безопасности в сети Интернет – это брандмауэры. Брандмауэры получили репутацию панацеи от большинства, если не от всех проблем, связанных с безопасностью в Интернет. На самом деле это, конечно, не так. Брандмауэр – просто еще один инструмент в обеспечении безопасности системы в целом. Он обеспечивает некоторый уровень защиты и, в общем, предоставляет возможность реализации политики безопасности на сетевом уровне. Уровень безопасности, предоставляемый брандмауэром, может варьироваться так же, как уровень безопасности на отдельно взятой машине. Существует традиционное противоречие между безопасностью, легкостью в использовании, стоимостью, и т.д.

Брандмауэр – это один из механизмов, используемых для управления и наблюдения за доступом в сеть и из нее с целью защиты сети. Брандмауэр является шлюзом, через который проходит весь трафик по направлению из внешнего мира к защищаемой сети и наоборот. Брандмауэр позволяет установить ограничения на количество и тип связей, имеющих место между приватной сетью и остальными сетями.

Брандмауэр – это способ установить «стену» между одной частью сети – внутренней сетью предприятия, и другой частью сети – например, Интернет. Уникальная особенность этой «стены» состоит в том, что она может предоставлять трафику с определенными характеристиками проход через узкие, тщательно охраняемые и наблюдаемые «двери». Сложность состоит лишь в установлении критериев, по которым тому или иному пакету разрешается или не разрешается пройти через «двери». Многочисленные книги, посвященные брандмауэрам, используют различную терминологию для того, чтобы описать различные типы брандмауэров. Это может затруднить понимание вопроса администраторами, кто не слишком хорошо знаком с брандмауэрами. Единственное, что мы может отметить здесь – это то, что не существует единой терминологии для описания брандмауэров.

Брандмауэр – не всегда, хотя и достаточно часто, единая машина. Брандмауэром может достаточно часто быть комбинация маршрутизаторов, сегментов сети и компьютеров. Как правило, брандмауэры строят на базе двух различных компонент – пакетных фильтров и серверов-посредников (прокси). Однако, в контексте настоящего документа, термин «брандмауэр» хотя и может обозначать более чем одно физическое устройство, но все-таки пакетных фильтр мы рассмотрим отдельно, как другой уровень модели безопасности.

Сервер-посредник – это способ сконцентрировать весь обмен информацией с внешними сетями на одной машине. Более того, как правило, одна машина является сервером-посредником для большого количества протоколов (telnet, ftp, http, smtp и т.д.). Но возможно так же выделение отдельных машин для работы серверов-посредников различных приложений и протоколов. Вместо подключения напрямую к внешнему серверу, клиенты соединяются с сервером-посредником, а он, в свою очередь, соединяется с внешним сервером. В зависимости от того, какой именно сервер-посредник используется, возможна настройка клиентов таким образом, чтобы они выполняли это перенаправление

автоматически, без участия пользователя. Другие требуют от пользователя дополнительных знаний, чтобы установить соединение с сервером-посредником, а уж потом – с требуемым внешним адресом.

Существует несколько важных преимуществ, касающихся безопасности, обеспечиваемых сервером-посредником. Например, есть возможность создать списки управления доступом, для протоколов, пользователей или систем, чтобы обеспечить некий уровень аутентификации, прежде чем предоставить запрашиваемый доступ. Наиболее развитые серверы-посредники, иногда называемые «шлюзами прикладного уровня», могут быть написаны таким образом, что способны различать, что же именно передается с использованием протокола, который они обслуживают, и блокировать некоторые возможности протокола. Например, такой шлюз может быть сконфигурирован так, что позволяет использовать команду get протокола ftp, но блокирует использование команды put. Например, пакетный фильтр способен только полностью запретить или разрешить доступ к сервису ftp, но не может ограничить какую-то его часть.

Серверы-посредники так же могут быть настроены таким образом, чтобы шифровать данные в зависимости от некоторого набора параметров. Организация может использовать это свойство для установления шифрованного приватного канала между локальными сетями, принадлежащими этой организации, и соединенными между собой посредством Интернет.

Брандмауэры обычно используют для защиты сети от внешних атак. Но их так же можно с успехом использовать для предоставления доступа в локальную сеть пользователям, по каким-либо причинам не находящимся внутри сети. Существует много примеров, когда работнику организации необходимо иметь доступ к ресурсам сети, когда он сам не находится в офисе. Например, если этот пользователь находится в командировке. В этом случае у него может быть доступ в Интернет, однако доступ этот будет осуществляться с «ненадежной» машины. Брандмауэры позволяют предоставить таким пользователям доступ в сети организации, но при этом ограничивают доступ прочих пользователей.

Наиболее полная реализация технологии брандмауэра использует некоторое количество серверов-посредников в сети, расположенной между двумя пакетными фильтрами. При такой установке внешний маршрутизатор фильтрует все попытки использовать атаки с использованием неверных IP-пакетов (IP-спуфинг, маршрутизацию от источника, фрагментирование пакетов), но позволяет серверам-посредникам предупреждать потенциальные проблемы с безопасностью на более высоких уровнях протокола. Внутренний же маршрутизатор должен блокировать любой трафик, кроме трафика между серверами-посредниками и клиентами. Если такую конфигурацию скрупулезно реализовать, то возможно достижение очень высокого уровня безопасности.

Большинство брандмауэров обеспечивают возможности ведения системных журналов. Информация, заносимая в них, может изменяться путем настройки таким образом, чтобы облегчить управление безопасностью сети. Ведение системных журналов может быть централизовано, а система может быть настроена таким образом, чтобы при совпадении некоторых условий, свидетельствующих о попытке атаки, предпринимались какие-то действия. В простейшем случае, это может быть уведомление администратору по электронной почте или пейджеру. Очень важно периодически следить за содержимым системных журналов, чтобы обнаружить попытки атак и предпринять какие-то действия для их отражения. Поскольку многие атакующие будут стремиться прикрыть последствия своего

вторжения путем редактирования журналов, необходимо позаботиться о защите этих журналов. Существует несколько методов такой защиты: например, использование WORM-накопителей; дубликация журналов не бумаге; централизованное ведение журналов с использованием возможностей, предоставляемых программной syslog. Другой метод состоит в использовании фиктивного принтера, когда все журналы выводятся на принтер через последовательный порт, а вместо принтера установлен другой компьютер, занимающийся хранением журналов.

На рынке существует большое количество различных продуктов типа брандмауэра, разной цены, качества, степени защиты. Коммерческие продукты могут стоит от нескольких тысяч до четверти миллиона долларов. Есть и более дешевые варианты, есть бесплатные утилиты. Однако какой бы брандмауэр вы не использовали, необходимы большие знания и опыт в области протоколов TCP/IP, сетевых технологий и сервисов. Любой брандмауэр требует регулярной поддержки, обслуживания, установки различных программных обновлений, а так же постоянного наблюдения. Когда планируется бюджет для установки брандмауэра, необходимо иметь в виду эти соображения, поскольку это ведет к увеличению стоимости продукта. Кроме того, вероятно, потребуется некоторое время, чтобы определить оптимальную конфигурацию брандмауэра для конкретной сети. При этом необходимо постоянно следить и принимать во внимание возможные угрозы, стоимость защищаемых активов, риски в случае нарушения режима безопасности.

Вообще говоря, защита сети от внешних угроз – трудная задача, поскольку защищать приходится от большого числа разнообразных угроз. Однако необходимо так же понимать, что защита брандмауэром – это только часть решения проблемы. Защитить сеть от всех видов атак возможно, только, разве что, отключив ее.

4.7. Пакетный фильтр

Пакетные фильтры, как уже упоминалось выше, могут быть составной частью брандмауэра. Однако, в некоторых случаях желательно использование внешнего фильтра на базе маршрутизатора. Маршрутизатор, как правило, перемещает пакеты данных между двумя или более различными сетями. «Обычный» маршрутизатор принимает пакет из сети А и передает адресату в сети Б. Маршрутизатор с пакетным фильтром решает не только куда направлять пакет, но и следует ли его передавать куда-то вообще. Это происходит с помощью набора фильтров, установленных на маршрутизаторе, с помощью которых он решает, передавать ли тот или иной пакет.

Обсуждение преимуществ или недостатков различных маршрутизаторов, работающих под управлением различных версий программного обеспечения не является целью этого документа. Однако, проверка маршрутизатора, который вы планируете использовать для фильтрации пакетов, способен ли он осуществлять фильтрацию по следующим критериям, может быть важной. Что это за критерии?

1. адрес источника и адрес приемника пакета
2. номер порта TCP источника и приемника пакета
3. статус бита “ACK” TCP
4. номер порта источника и приемника UDP и направление передачи

Другая информация, необходимая для создания схемы фильтрации это: может ли маршрутизатор изменять порядок приложения фильтров (для оптимизации быстродействия,

однако это может в некоторых случаях повлечь возможность неавторизованного доступа), и возможно ли применение фильтров (возможно, различных) для входящих и исходящих пакетов на каждом интерфейсе. Если маршрутизатор способен фильтровать только исходящие пакеты, он сам как бы оказывается «снаружи» и может быть атакован. Кроме того, что такой маршрутизатор сам может стать объектом атаки, особое значение возможность использования разных фильтров на разных интерфейсах приобретает на маршрутизаторах, имеющих больше, чем 2 интерфейса. Другое важное свойство – возможность создания фильтров, основанных на опциях заголовка IP, в частности, статусе фрагмента пакета. Построение хороших фильтров может быть очень сложным делом, требующим больших знаний и хорошего понимания типов сервисов и протоколов, подлежащих фильтрации.

Для лучшей безопасности, фильтры обычно ограничивают доступ между сетями каким-то одним адресом. Так сказать «бастионом». Доступ ко всем ресурсам внешней сети возможен только через этот адрес. Соответственно, только этот адрес, а не все адреса внутренней сети, может быть атакован. Администраторам намного проще поддерживать определенный уровень безопасности на этой машине, тщательно его защищать и внимательно следить за попытками атак на него. Чтобы ресурсы, расположенные за брандмауэром, были доступны обычным пользователям, необходимо обеспечить пересылку обращений к ним, средствами сервера-«бастиона». Некоторые сервисы имеют встроенные возможности пересылки (например, DNS или SMTP), для других необходимо воспользоваться серверами-посредниками. Серверы-посредники могут обеспечить доступ к ресурсам, расположенным за брандмауэром, наиболее безопасным способом.

4.8. Внешние каналы связи

Этот уровень модели описывает в целом внешнее подключение локальной сети (приватной сети) к сетям общего пользования или глобальным сетям. Как правило, в состав этого уровня входит оборудование и телефонные линии, которые не находятся во владении организации или под властью системного администратора. Целью этого уровня является показать это подключение и оборудование в рамках общей модели безопасности. На этом уровне ваше собственное оборудование, соединяется с оборудованием, принадлежащим, например, телефонной компании. Ваша собственная политика безопасности и ее конкретная реализация должны постоянно отражать ваше взаимодействие с телефонной компанией или с другим владельцем оборудования (например, провайдером Internet). Кроме этого, в Политике безопасности необходимо разработать план взаимодействия с провайдером или телефонной компанией в случае возникновения нештатных ситуаций. В дальнейшем это может исключить или, по меньшей мере, сильно снизить ущерб при возникновении подобных ситуаций.

Говоря о подключении к внешним каналам связи, нельзя не упомянуть о том, что все устройства, работающие за пакетным фильтром (или до него – как посмотреть), находятся в потенциально опасном и даже враждебном окружении. Более того, в случае атаки на внешние каналы, администратору, скорее всего, ничего не удастся сделать, чтобы оказать противодействие. Действительно, если атака типа «отказ в обслуживании» путем «забивания» полосы пропускания внутри сети ликвидируется тривиально, то ликвидировать «забивание» внешнего канала может только поставщик услуг Интернет – провайдер.

Поэтому важной частью обеспечения безопасности на данном уровне является проработка процедур взаимодействия с провайдером.

Администраторам следует учитывать, что внешним каналом связи является и канал связи между двумя локальными сетями, расположенными на удалении. При этом поток информации между сетями проходит через оборудование, не принадлежащее организации-владельцу сети. Так же невозможно организовать и физическую охрану. Поэтому возможно прослушивание этого канала, обрывы т.д.

Это вынуждает использовать несколько путей подключения, либо создание резервного канала по требованию. Например, при обрыве выделенной линии может происходить установление связи по коммутируемым линиям.

Особого внимания требует информация, передаваемая по внешним каналам связи. Скорее всего, весь трафик придется шифровать каким-либо способом. Только такой подход позволит обеспечить безопасность передаваемых данных на приемлемом уровне. Возможно так же использование разных каналов для организации работы сети и для предоставления внешних сервисов. Например, если организация содержит WWW-сервер, то для его работы, вероятно, будет разумным установить отдельный канал. Как правило, серверы общего доступа, будь то FTP или WWW, привлекают к себе внимание злоумышленников. Следовательно, они значительно больше подвержены атакам извне. В частности, атакам класса «отказ в обслуживании». Выделение отдельного канала для собственных нужд и отдельного – для внешних сервисов, во-первых, существенно снизит риск атаки сети организации, а во-вторых, позволит избежать нарушений в работе сети при атаке на внешние сервисы и серверы.

4.9. Заключение

В заключение, приведем таблицу, аналогичную таблице 3, показывающую на каком уровне предложенной модели реализуются различные механизмы безопасности.

Таблица 4.2.

Функция безопасности	Уровень модели безопасности						
	1	2	3	4	5	6	7
Аутентификация			+	+			+
Управление доступом		+	+	+	+	+	+
Конфиденциальность соединения			+	+	+		+
Конфиденциальность вне соединения				+			+
Избирательная конфиденциальность			+	+			+
Конфиденциальность трафика		+	+	+	+		+
Целостность с восстановлением				+			+
Целостность без восстановления			+	+			+
Избирательная целостность			+	+			+
Целостность вне соединения				+			+
Неотказуемость			+	+			+

Обозначения:
“+” – данный уровень может предоставить функцию безопасности

Отметим особо, что уровень 4 – «сервисы» предоставляет все функции безопасности. Не менее важен и уровень брандмауэра (3), поскольку именно он обеспечивает реализацию функций безопасности для внешних сетевых соединений. Наконец, последний уровень – 7 –

«Политика безопасности». Знаки «+» во всех строках обозначают, что в политике безопасности могут и должны быть отражены все аспекты обеспечения безопасности сети.

Применение предложенной модели на практике, возможно, не сразу очевидно. Действительно, ведь современные сети состоят из большого количества рабочих станций, персональных компьютеров и серверов, а так же сетевого оборудования. Однако, как показала практика использования модели, любую сеть можно представить в предложенных терминах. Возможно, для этого ее придется разбить на несколько подсетей с разными уровнями доступа, соединенных между собой брандмауэрами. В этом случае, каждую такую подсеть следует рассмотреть отдельно. С точки зрения же внешних атак, можно рассматривать всю сеть целиком. Такой подход позволит упростить практическую оценку безопасности сетей и сервисов.

Позволим себе еще несколько замечаний в адрес защиты сервисов. Как можно заметить в реальной практике, не все сервисы являются целостными. Простейший пример представляет собой сервис NFS. При конфигурировании этого сервиса права доступа различных категорий пользователей к дисковым ресурсам, предоставляемым сервисом, определяются (в случае Solaris) правами пользователей и групп NIS+. В этом случае сервис NFS выступает клиентом сервиса NIS+. Таким образом, для защиты сервиса NFS недостаточно правильно его сконфигурировать и настроить. Недостаточно так же использовать инструменты для защиты сервиса NFS. Можно сказать, что сервис NFS в данном случае является составным сервисом, поскольку для его защиты необходимо так же обеспечить защиту сервиса NIS+. Вообще говоря, защита служебных сервисов, возможно, даже более важна, нежели защита основных сервисов, поскольку от них зависит безопасность прочих сервисов сети. Таким образом, нарушение работы служебных сервисов способно нарушить работу и вызвать нарушение защиты нескольких основных сервисов.

При рассмотрении модели невозможно выделить более или менее важные уровни. Все уровни важны одинаково для обеспечения безопасности. Вместе с тем, можно отметить особую роль Политики безопасности, как основополагающего документа.

Именно Политикой безопасности определяются шаги и меры, принимаемые на каждом уровне модели. Именно Политика безопасности является тем стержнем, вокруг которого строится безопасность системы в целом. Это важно донести до понимания как руководства организации, так и рядовых сотрудников. Хотя, конечно, особенно важна поддержка руководства. Как уже упоминалось выше, Политика безопасности, не имеющая активных сторонников в руководстве, обречена.

Но так же важно и понимание сотрудниками организации того, что общая безопасность информации определяется не только персоналом, отвечающим за безопасность непосредственно, но каждым сотрудником. Очень распространенная угроза – компьютерные вирусы. Неконтролируемое распространение программных продуктов в сети организации способно привести к серьезному нашествию вирусов, а, следовательно, и к возможным потерям информации.

Использование рабочих машин в качестве игровых автоматов так же подвергает опасности информацию в корпоративной сети. Особенно опасными могут быть сетевые

игры. Они создают очень большой и бесполезный трафик в сети, что может привести к заторам, нарушению работы маршрутизаторов, перегрузке внешних каналов связи и т.д. Кроме того, большинство игр в настоящее время, как и множество утилит не покупаются, а «скачиваются» из Интернет, что значительно увеличивает риск распространения вирусов. Все вышесказанное вынуждает серьезнее относиться к управлению персональными компьютерами и рабочими станциями пользователей, а так же к понятию «допустимое использование сетевых ресурсов». Очень часто в реальной практике этим пренебрегают.

Словом, как мы видим, безопасность информации в сети предприятия или организации – вопрос комплексный и довольно сложный. Обеспечение безопасности включает в себя множество аспектов как административных, так и программно-технических. Причем заранее сложно сказать в каком случае какие из них преобладают и играют большую роль в общей картине безопасности. Именно такому анализу и посвящена следующая глава.

5. Практические рекомендации по обеспечению безопасности ИС

5.1. Общие положения

В этой главе мы рассмотрим практические моменты, касающиеся обеспечения безопасности информационных систем. Но прежде необходимо пояснить само это понятие.

Под безопасностью информационной системы мы будем понимать защищенность информации и поддерживающей ее инфраструктуры от случайных или преднамеренных воздействий естественного или искусственного характера, чреватых нанесением ущерба владельцам или пользователям информации и поддерживающей инфраструктуры.

Как уже отмечалось выше, информация в корпоративной сети должна отвечать трем требованиям: целостность, доступность и конфиденциальность. Безопасность информационной системы на практике – это поддержание перечисленных требований в конкретных условиях. Ранее уже упоминался комплексный характер обеспечения безопасности. Дополним лишь, что обеспечение безопасности не является сугубо прерогативой и заботой самой организации. Обеспечение безопасности информации начинается с законодательной базы, отвечающей за сертификацию различных средств обеспечения безопасности (прежде всего – криптографических), предусматривающей ответственность за нарушение режима безопасности. До тех пор, пока не будет соответствующих законов, можно говорить лишь о частичном обеспечении безопасности.

До сих пор в России не проработан механизм доказательства нарушения режима безопасности организаций. Поэтому нередко организация, подвергшаяся атаке со стороны, через Интернет, в ответ предпринимает атаку на сеть организации, откуда, собственно, и была атакована. Все это никоим образом не способствует безопасности обоих. Более того, организация, с чьей стороны была произведена атака, на самом деле ведет себя крайне рискованно, так как сотрудник, использовавший сеть для атаки на стороннюю организацию, сам вряд ли заслуживает доверия.

Особого внимания заслуживает также тот факт, что подавляющее большинство вычислительной техники и программного обеспечения, используемого в России, зарубежного производства. И лишь очень немногие из них имеют российские сертификаты безопасности. Это обстоятельство не позволяет потребителям информационных систем получить готовую систему с сертификатом безопасности. А заказывать и сертифицировать собственное программное обеспечение практически ни у кого нет ресурсов. Таким образом, для потребителя в России в настоящее время практически нереально заказать и получить «под ключ» безопасную информационную систему со всеми необходимыми сертификатами.

Тем не менее, прогресс есть. Уже принят новый Уголовный кодекс, в котором предусмотрена ответственность за атаки на информационные системы. Более того, есть уже и первые прецеденты применения этой статьи на практике. Конечно, несовершенство законодательной базы и процедуры доказательства таких преступлений еще пока оставляет много подобных преступлений безнаказанными, однако движение есть, и это не может не радовать.

Другой положительный момент связан с тем, что все чаще коммерческие структуры проявляют интерес к безопасности собственной информации. Если раньше безопасность

заботила лишь режимные государственные органы и под безопасностью понималась конфиденциальность, то теперь ситуация изменилась. Надо признать, что первоначально вопросы безопасности информации волновали только финансовые, банковские структуры. Это можно понять, поскольку именно они зачастую являются «лакомым кусочком» для злоумышленников всех мастей, кроме того, они имеют средства на развитие и поддержание режима безопасности. Но это не значит, что информация, используемая в других местах менее ценна. Напротив, если оценить стоимость информации, хранимой в бухгалтерских или складских программах, может оказаться, что она очень велика. И тут как нельзя более важно более широкое понимание безопасности, нежели просто конфиденциальность. Доступность и целостность порой играют даже большую роль, чем конфиденциальность, поскольку потеря информации в бухгалтерской системе может очень дорого обойтись любому предприятию.

Каким же образом обеспечить безопасность информационной системы на практике? Разумеется, на этот вопрос не может быть единого и полного ответа. Связано это с тем, что каждая организация имеет свою специфику, свой набор сервисов и защита должна осуществляться от разных угроз. Вместе с тем, вполне реально описать подход к построению безопасной информационной системы. Этот же подход можно использовать для оценки уровня безопасности уже существующей системы, и для выработки рекомендаций по повышению и поддержанию уровня безопасности.

Базовый подход к достижению этой цели был предложен Fites в 1989 году [12]. Он включает в себя следующие шаги:

1. Определить, что именно мы хотим защищать.
2. Определить, от чего именно мы хоть защитить тот или ной ресурс.
3. Определить какие именно опасности существуют и насколько они велики.
4. Реализовать меры, защищающие требуемые ресурсы от угроз. Эти меры должны быть экономически целесообразны.
5. Пересмотреть оставшиеся риски и угрозы, начиная с шага 1.

Значительная часть настоящей работы посвящена шагу 4 в данной цепочке. Однако другие шаги не должны быть лишены внимания, если, конечно, мы хотим установить действительно реальную и эффективную политику безопасности. Традиционный подход к оценке эффективности защиты таков – стоимость мероприятий по защите не должна превышать возможного ущерба от возможного нарушения режима безопасности. Ущерб в данном контексте включает в себя не только реальные денежные потери, но и ущерб репутации, потерю доверия со стороны клиентов и партнеров, а так же иные виды ущерба. Без полного и разумного понимания, что именно и от каких угроз мы защищаем, следовать этому правилу будет крайне сложно.

Тут мы еще раз вернемся к важности четко сформулированной Политики безопасности. Одной из наиболее важных причин создания комплексной Политики безопасности, является уверенность, что средства, затрачиваемые на обеспечение безопасности, тратятся разумно и дают ощутимую отдачу в виде увеличения защищенности системы. Кроме того, без точной формулировки Политики безопасности, порой очень сложно оценить, в какой именно области требуется наибольшая защита, а значит, и капиталовложения. Например, общественность более всего опасается так называемых «хакеров» или «взломщиков» компьютерных систем. Однако исследования показывают, что ущерб, наносимый собственными работниками, значительно выше

Как мы видим, предложенный общий подход к обеспечению безопасности системы по существу является бесконечным циклом. Точнее, он продолжается в течение всего периода жизни информационной системы. Ранее уже упоминалось, что обеспечение безопасности – это не разовая работа, она выполняется все время, пока существует система. В противном случае, защита имеет тенденцию ослабевать со временем. Причем происходит это довольно быстро.

Вернемся к практическим подходам к обеспечению безопасности. Какие же шаги необходимо предпринять, чтобы обеспечить безопасность информационной системы? Для определения этих шагов используем модель безопасности, предложенную в предыдущей главе. Разумно будет рассматривать ее «сверху вниз», то есть, начиная с политики безопасности. Для нижних уровней (начиная с уровня локальной сети) возможно применение программно-технических мер. Большая часть этого раздела будет посвящена как раз программно-техническим мерам обеспечения безопасности.

В разделе 5.2 мы рассмотрим первый пункт приведенного выше плана – определили, что же именно мы хотим защищать. Это называется идентификацией активов. Раздел 5.3 будет, соответственно посвящен второму пункту – идентификация угроз, или от чего мы хотим защищать активы. В разделе 5.4 мы позволим себе несколько замечаний касательно физической защиты сети и информационной системы в целом. В разделе 5.5 будут приведены общие принципы конфигурации основных сетевых сервисов, а так же программные инструменты обеспечения или тестирования безопасности систем. В разделе 5.6 будут даны ответы на вопрос о том, каким образом реагировать на нарушение режима безопасности. Наконец, раздел 5.7 посвящен мерам, предпринимаемым после инцидента нарушения режима безопасности.

5.2. Идентификация активов

Идентификация активов совместно со следующим шагом – идентификацией угроз, составляет основу управления рисками в информационной системе. Управление рисками предназначено для оценки реальных рисков, имеющихся в конкретной системе, с тем, чтобы обеспечить эффективную защиту от наиболее существенных рисков. Без процедур управления рисками и учета рисков, невозможно определить от каких именно угроз следует защищаться в первую очередь. Следовательно, нет возможности рационально спланировать мероприятия по защите и определить необходимое финансирование. Управление рисками – это важнейший шаг в обеспечении безопасности не только информационных систем. Но в области информационных систем он имеет свои особенности.

Первым шагом анализа рисков является определение всех активов, над защитой которых предстоит работать. Некоторые вещи просты и очевидны: ценная частная информация, интеллектуальная собственность, какое-либо оборудование, некоторые можно только описать: например, люди, работающие с системой.

Промежуточная цель этой работы – составить список всего, что может быть значимо в контексте безопасности системы. Как пример, что же именно следует рассматривать в контексте безопасности можно привести [13]:

1. Оборудование: процессоры, платы, клавиатуры, терминалы, рабочие станции, персональные компьютеры, принтеры, дисковые накопители, линии связи, терминальные серверы, маршрутизаторы и т.д.

2. Программное обеспечение: утилиты, диагностические программы, операционные системы, прикладное программное обеспечение и т.д.
3. Данные: находящиеся в реальном использовании, хранимые в режиме on-line, данные в архивах (ленты и CD-ROM), системные журналы, базы данных, транзитные данные, проходящие по линиям коммуникации и т.д.
4. Персонал: пользователи, администраторы сетей, сервисов и серверов.
5. Документация: относящаяся к программам, оборудованию, системам, местные административные процедуры.
6. Носители информации: бумага, бланки и документы, магнитные носители и т.д.

Вообще говоря, уязвимым является каждый компонент информационной системы - от куска сетевого кабеля, который могут прогрызть мыши, до базы данных, которая может быть разрушена из-за неумелых действий администратора. Как правило, в сферу анализа невозможно включить каждый винтик и каждый байт. Приходится останавливаться на некотором уровне детализации, опять-таки отдавая себе отчет в приближенности оценки. Для новых систем предпочтителен детальный анализ; старая система, подвергшаяся небольшим модификациям, может быть проанализирована менее глубоко.

Выбор анализируемых объектов и степени детальности их рассмотрения - первый шаг в оценке рисков. Для небольшой организации допустимо рассматривать всю информационную инфраструктуру; однако, если организации крупная, всеобъемлющая оценка может потребовать неприемлемых затрат времени и сил. В таком случае следует сосредоточиться на наиболее важных сервисах, заранее соглашаясь с приближенностью итоговой оценки. Если важных сервисов все еще много, выбираются те из них, риски для которых заведомо велики или неизвестны.

Одним из главных результатов процесса идентификации активов можно считать получение детальной информационной структуры организации и способов использования такой структуры. Если обратиться к модели безопасности информационной системы, приведенной в предыдущей главе, то результат этапа идентификации активов – это не просто перечень оборудования и программного обеспечения, это описание того, что именно входит в каждый уровень модели, какое оборудование и/или программное обеспечение составляет тот или иной уровень. Подобное следование модели безопасности позволяет систематизировать и структурировать весь процесс оценки безопасности системы.

Процесс управления рисками – процесс нелинейный. В какой-то момент при проведении анализа может оказаться, что выбранные первоначально границы анализа не отвечают поставленным задачам. В этом случае, необходимо вернуться снова к определению степени детальности рассмотрение и совершить еще одну итерацию. Вполне возможно, таких итераций потребуется несколько. Процесс идентификации активов – это подготовительный процесс в анализе рисков, поскольку здесь еще не рассматриваются угрозы. Собственно риски появляются там, где возникают угрозы. Поэтому следующий шаг – идентификация угроз.

5.3. Идентификация угроз и управление рисками

Как уже говорилось ранее, безопасность информации состоит из трех компонент: целостность, доступность и конфиденциальность. Соответственно, и все угрозы можно разделить на три класса:

- неавторизованное ознакомление с информацией – угроза конфиденциальности
- неавторизованное изменение информации – угроза целостности
- отказ в обслуживании – угроза доступности

На практике, конечно, существует множество угроз, порой даже не связанных с неправомерными действиями. Их тоже надо учитывать. Примерами таких угроз могут быть, например, наличие мышей или тараканов в помещении, опасность пожаров, отказ аппаратных компонент системы. Мыши могут прогрызть кабель, следовательно, сеть перестанет правильно функционировать – информация окажется недоступной. Тараканы способны вызвать короткое замыкание в электронных компонентах. Пожар может вывести из строя большое количество аппаратуры. Наконец, сама аппаратура, серверы и сетевое оборудование тоже не вечны. А выход из строя, например, жестких дисков или блоков питания – не редкость. Все это необходимо учитывать при анализе возможных угроз. Результатом этапа идентификации угроз может служить перечень возможных угроз как для информационной системы в целом, так и для отдельных ее компонент.

Отметим, что само понятие "угроза" в разных ситуациях зачастую трактуется по-разному. Например, для подчеркнуто открытой организации может просто не существовать угроз конфиденциальности - вся информация считается общедоступной; однако в большинстве случаев нелегальный доступ считается серьезной опасностью.

Самыми частыми и самыми опасными (с точки зрения размера ущерба) являются непреднамеренные ошибки пользователей, операторов, системных администраторов и других лиц, обслуживающих информационные системы. Иногда такие ошибки являются угрозами (неправильно введенные данные, ошибка в программе, вызвавшая крах системы), иногда они создают слабости, которыми могут воспользоваться злоумышленники (таковы обычно ошибки администрирования). Согласно [14], 65% потерь - следствие непреднамеренных ошибок. Пожары и наводнения можно считать пустяками по сравнению с безграмотностью и расхлябанностью. Очевидно, самый радикальный способ борьбы с непреднамеренными ошибками - максимальная автоматизация и строгий контроль правильности совершаемых действий.

На втором месте по размерам ущерба располагаются кражи и подлоги. По данным газеты USA Today [14], в 1992 году в результате подобных противоправных действий с использованием персональных компьютеров американским организациям был нанесен суммарный ущерб в размере 882 миллионов долларов. Можно предположить, что подлинный ущерб намного больше, поскольку многие организации по понятным причинам скрывают такие инциденты. В большинстве расследованных случаев виновниками оказывались штатные сотрудники организаций, отлично знакомые с режимом работы и защитными мерами. Еще раз мы убеждаемся в том, что внутренняя угроза гораздо опаснее внешней.

Весьма опасны так называемые обиженные сотрудники - нынешние и бывшие. Как правило, их действиями руководит желание нанести вред организации-обидчику, например:

- повредить оборудование;
- встроить логическую бомбу, которая со временем разрушит программы и/или данные;
- ввести неверные данные;
- удалить данные;

- изменить данные и т.д.

Обиженные сотрудники, даже бывшие, знакомы с порядками в организации и способны вредить весьма эффективно. Необходимо следить за тем, чтобы при увольнении сотрудника его права доступа к информационным ресурсам аннулировались.

Угрозы, исходящие от окружающей среды, к сожалению, отличаются большим разнообразием. В первую очередь следует выделить нарушения инфраструктуры - аварии электропитания, временное отсутствие связи, перебои с водоснабжением, гражданские беспорядки и т.п. Разумеется, опасны стихийные бедствия и события, воспринимаемые как стихийные бедствия - пожары, наводнения, землетрясения, ураганы. По данным [14], на долю огня, воды и аналогичных "врагов" (среди которых самый опасный - низкое качество электропитания и его перебои) приходится 13% потерь, нанесенных информационным системам.

Много говорят и пишут о хакерах, но исходящая от них угроза зачастую преувеличивается. Верно, что почти каждый Internet-сервер по несколько раз в день подвергается попыткам проникновения; верно, что иногда такие попытки оказываются удачными; верно, что изредка подобные действия связаны со шпионажем. Однако в целом ущерб от деятельности хакеров (в сравнении с другими угрозами) представляется не столь уж значительным. Вероятно, больше всего пугает непредсказуемость действий людей такого сорта. Представьте себе, что в любой момент к Вам в квартиру могут забраться посторонние люди. Даже если они не имеют злого умысла, а зашли просто так, посмотреть, нет ли чего интересного, приятного в этом мало.

Много говорят и пишут и о программных вирусах. В этой связи обратим внимание на следующий факт. Как показало проведенное в 1993 году исследование [14], несмотря на экспоненциальный рост числа известных вирусов, аналогичного роста количества инцидентов, вызванных вирусами, не зарегистрировано. Соблюдение несложных правил компьютерной гигиены сводит риск заражения практически к нулю. Там где работают, а не играют, число зараженных компьютеров составляет лишь доли процента.

Мы не будем останавливаться на уточнении понятий "зловредный код", "вирус", "червь", "Троянский конь" (см., например, [15, 16, 17]). Справедливости ради отметим лишь, что зловредный код поражает не только персональные компьютеры, но и системы других типов.

Однако просто перечислить угрозы недостаточно. Действительно, перечисление угроз может быть сколь угодно полным – вплоть до угрозы землетрясения. Необходимо еще оценить вероятность возникновения той или иной угрозы, а так же – возможный ущерб. Тут будет уместно сказать несколько слов о методике оценки вероятности и возможного ущерба. В общем, методика таких оценок составляет собой отдельную проблему. Во-первых, довольно сложно точно оценить вероятность, например, пожара. Во-вторых, оценка возможного ущерба тоже затруднительна.

Несомненно, вероятность, например, пожара, можно рассчитать в процентах, исходя из статистики исторической и современной. Вероятность отказа жесткого диска можно рассчитать, исходя из заявленной наработки на отказ. Однако, для любой мало-мальски крупной организации подобная методика оценки вероятности того или иного события неприемлема. Связано это прежде всего с тем, что она займет слишком много времени, а значит, окажется слишком дорогой.

Оценка ущерба тоже имеет свои особенности. Начать хотя бы с того, в каких единицах измерять ущерб? Рублями, долларами? Насколько точной окажется эта оценка? Учитывая, что оценить, например, потерю репутации в денежном выражении вообще вряд ли возможно, можно прийти к выводу, что денежная оценка не может быть всеобъемлющей в реальной практике.

Исходя из этих соображений, можно предложить пользоваться приближенными методами оценки [18]. Например, оценка вероятности той или иной угрозы вполне может производиться с использованием 5-ти балльной шкалы. Ущерб, конечно, желательно рассчитывать в денежном выражении, но с учетом приближенности оценки. Выражение возможного ущерба в некоторой сумме полезно с точки зрения оценки эффективности тех или иных защитных мер. Кроме того, такая оценка позволить более четко вести речь о финансировании защитных мероприятий. В некоторых источниках, посвященных проблеме безопасности информационных систем можно встретить рекомендации пользоваться приближенной 3-х или 5-ти балльной шкалой для оценки ущерба. Возможен и такой подход.

После получения перечня угроз для каждого из активов или для системы в целом, а так же соответствующих вероятностей и ущербов от этих угроз, можно довольно легко составить перечень рисков. Критерием оценки риска можно считать произведение вероятности на возможный ущерб. После этого все возможные риски можно ранжировать по степени опасности и, следовательно, важности.

Если какие-либо риски оказались недопустимо большими, необходимо использование дополнительных защитных мер. Как правило, существует несколько возможных способов снижения того или иного риска. Каждый из способов так же характеризуется стоимостью. Стоимость защитного механизма может состоять из стоимости закупки оборудования, стоимости программного обеспечения, обучения персонала, а так же стоимости уменьшения удобства работы с системой. Последнее, впрочем, весьма сложно оценить. Соотношение стоимости внедрения защитных мер и возможного ущерба, а так же соотношение между вероятностью той или иной угрозы до и после внедрения защитных мер может служить оценкой эффективности выбранных защитных механизмов.

Важным обстоятельством является совместимость нового средства со сложившейся операционной и аппаратно-программной инфраструктурой. Меры безопасности, как правило, носят недружественный характер, что может отрицательно сказаться на энтузиазме работников. Порой сохранение духа открытости может оказаться важнее минимизации материальных потерь. Впрочем, такого рода ориентиры должны быть расставлены в Политике безопасности.

Можно представить себе ситуацию, когда для уменьшения риска не существует эффективных и приемлемых по цене мер. Например, компания, базирующаяся в сейсмически опасной зоне, не всегда может позволить себе строительство защищенной штаб-квартиры. В таком случае приходится поднимать планку приемлемого риска и переносить центр тяжести на смягчение последствий и выработку планов восстановления после аварий, стихийных бедствий и иных происшествий. Продолжая пример с сейсмоопасностью, можно рекомендовать регулярное тиражирование данных в другой город и овладение средствами восстановления первичной базы данных.

Как и всякую иную деятельность, реализацию и проверку новых регуляторов безопасности следует предварительно распланировать. В плане необходимо учесть наличие

финансовых средств, сроки обучения персонала. Нужно составить план тестирования (автономного и комплексного), если речь идет о программно-техническом механизме защиты.

Когда намеченные меры приняты, необходимо проверить их действенность, то есть убедиться, что остаточные риски стали приемлемыми. Если это на самом деле так, значит, все в порядке и можно спокойно намечать дату ближайшей переоценки. В противном случае придется проанализировать допущенные ошибки и провести повторный сеанс управления рисками немедленно.

5.4. Физическая защита

Безопасность компьютерной системы зависит от окружения, в котором она работает. Следовательно, необходимо принять меры для защиты зданий и прилегающей территории, поддерживающей инфраструктуры и самих компьютеров.

Мы кратко рассмотрим следующие направления физической защиты:

- физическое управление доступом,
- противопожарные меры,
- защита поддерживающей инфраструктуры,
- защита от перехвата данных,
- защита мобильных систем.

Меры физического управления доступом позволяют контролировать и при необходимости ограничивать вход и выход сотрудников и посетителей. Контролироваться может все здание организации и, кроме того, отдельные помещения, например, те, где расположены серверы, коммуникационная аппаратура и т.п. Средства физического управления доступом известны давно - это охрана, двери с замками, перегородки, телекамеры, датчики движения и многое другое. Важно в максимальной степени разграничить компьютеры и поток посетителей или, в крайнем случае, позаботиться о том, чтобы от окон и дверей не просматривались экраны мониторов и принтеры. Необходимо, чтобы посетители отличались от штатных сотрудников. Если отличие состоит в том, что посетителям выдаются идентификационные карточки, а сотрудники ходят "без опознавательных знаков", злоумышленнику достаточно снять карточку, чтобы его считали "своим". Очевидно, карточки разных видов нужны всем.

Профессия пожарника - одна из древнейших, но пожары по-прежнему случаются и наносят большой ущерб. Мы не собираемся цитировать параграфы противопожарных инструкций или изобретать новые методы борьбы с огнем - для этого есть профессионалы. Отметим лишь крайнюю желательность установки противопожарной сигнализации и автоматических средств пожаротушения. Обратим также внимание на то, как защитные меры могут создавать новые слабости. Если на работу взят новый охранник, это, вероятно, улучшает физическое управление доступом. Если же он по ночам курит и пьет, то повышенная пожарная опасность делает его скорее врагом, чем другом организации.

К поддерживающей инфраструктуре можно отнести системы электро-, водо- и теплоснабжения, кондиционеры, средства коммуникаций. В принципе к ним применимы те же требования целостности и доступности, что и к информационным системам. Для обеспечения целостности нужно защищать оборудование от краж и повреждений. Для

поддержания доступности целесообразно выбирать оборудование с максимальным временем наработки на отказ, дублировать ответственные узлы, всегда иметь под рукой запчасти.

Отдельную проблему составляют аварии водопровода. Они происходят нечасто, но чреваты серьезными материальными потерями. При размещении компьютеров разумно принять во внимание расположение водопроводных и канализационных труб и постараться держаться от них подальше. Сотрудники должны знать, куда следует обращаться при обнаружении протечек.

Перехват данных может осуществляться самыми разными способами: подсматриванием за экраном монитора, чтением пакетов, передаваемых по локальной сети, улавливанием стука иголок матричного принтера или кнопок на клавиатуре, анализом побочных электромагнитных излучений и наводок (ПЭМИН). К сожалению, некоторые способы перехвата данных, такие как анализ ПЭМИН [19], относительно доступны и дешевы, а бороться с ними трудно и дорого. Остается уповать на то, что для коммерческих систем обеспечение конфиденциальности не является главной задачей, пытаться держать под контролем линии связи (например, заключать их в надувную оболочку с обнаружением прокалывания) и разместиться в тихом особняке, поодаль от других домов.

Мобильные и портативные компьютеры - заманчивый объект кражи. Их довольно часто оставляют без присмотра, в автомобиле или на работе, и унести и спрятать такой компьютер весьма несложно. Следует настоятельно рекомендовать шифрование данных на жестких дисках ноутбуков и лэптопов.

Вообще говоря, при выборе средств физической защиты следует производить анализ рисков. Так, принимая решение о закупке источника бесперебойного питания, необходимо учесть качество электропитания в доме, занимаемом организацией (впрочем, почти наверняка оно окажется плохим), характер и длительность нарушений питания, стоимость доступных источников и возможные потери от аварий (выход из строя техники, приостановка работы организации и т.п.). В то же время, во многих случаях решения очевидны. Меры противопожарной безопасности обязательны для всех организаций. Стоимость реализации многих мер (например, установка обычного замка на дверь серверной комнаты) пренебрежимо мала, другие имеют хоть и заметную стоимость, но все же явно меньшую, чем возможный ущерб. К числу последних можно отнести регулярное копирование больших баз данных. Физическая защита, как и другие области информационной безопасности, должна базироваться на здравом смысле, который подскажет большинство решений.

5.5. Основные программно-технические меры обеспечения безопасности

5.5.1. Определение Плана безопасности

Все предприятия и организации должны определить полный комплексный план безопасности. Этот план должен быть полнее, чем те правила, которые были описаны в предыдущем разделе. И он должен определять общее направление, детали которого будут раскрыты в конкретных правилах [18].

Очень важно спроектировать такой план безопасности достаточно широко, но вместе с тем – просто и недвусмысленно, поскольку все остальные документы и правила должны

соответствовать общей архитектуре безопасности. Например, жесткие правила относительно подключения в Интернет одновременно с очень слабыми ограничениями относительно доступа по модему к сети организации несовместимы с общей философией жесткого ограничения доступа к сети извне.

План безопасности должен определять: список сетевых сервисов, которые предоставляются пользователям; какие именно отделы организации будут предоставлять тот или иной сервис; кто может иметь доступ к предоставляемому сервису; каким именно образом этот доступ будет обеспечиваться; кто будет осуществлять управление сервисом; и т.д.

План должен также отвечать на вопрос, каким именно образом будет происходить реагирование на попытку нарушения режима безопасности? В разделе 5 настоящего документа будет дано более серьезное и глубокое обсуждение этого вопроса, однако наиболее важно для каждой организации определить классы опасности нарушений и степень реакции для каждого класса. Простой пример – организации, использующие брандмауэры, должны определить, сколько попыток проникновения через брандмауэр должны произойти, прежде чем потребуется предпринимать какие-то меры? Ранжирование необходимо как для возможных атак, так и для соответствующих им реакций. Организации, не имеющие брандмауэров, должны, например, определить, является ли попытка соединения с машиной внутри сети извне атакой? А попытки систематического сканирования адресов, принадлежащих организации?

Для сетей, подключенных к Интернет, раздуваемые средствами массовой информации опасности внешнего вторжения могут «затенять» значительно более важные и потенциально опасные внутренние проблемы с безопасностью. Аналогично, организации, не использующие внешнего подключения и имеющие строгие и четкие правила внутренней безопасности, могут оказаться неспособными правильно отреагировать на возможные внешние угрозы.

5.5.2. Разделение сервисов

Существует большое количество разнообразных сервисов, которые организация может предоставлять для своих пользователей. Некоторые из пользователей могут быть так же вне сети организации. Существует большое количество доводов, относящихся к вопросам безопасности, в пользу выделения каждого сервиса на отдельный, выделенный сервер. Кроме безопасности, в пользу такого решения говорят, например, соображения производительности. Но их обсуждение не входит в задачи данного документа.

Сервисы, которые организация может предоставлять, как правило, имеют различные уровни важности и потребности, а так же – модели доверия и надежности. Жизненно важные для безопасности или работы сети сервисы вообще лучше всего расположить на выделенных серверах с крайне ограниченным доступом (модель «запретить все» – см. ниже), вместо того, чтобы располагать их на серверах, предоставляющих менее безопасные публичные сервисы.

Кроме того, важно проводить различие между серверами, работающими в рамках разных моделей доверия и надежности (например, серверами, находящимися под защитой брандмауэра, и серверами в общедоступной сети).

Некоторые сервисы, которые желательно вынести на отдельные серверы кратко будут рассмотрены ниже. Необходимо помнить, что общая защита системы определяется защитой

наиболее слабого компонента. Большинство наиболее известных проблем с безопасностью в последние годы связано с системой электронной почты. Однако, если сервисы, предоставляемые в сети, не разделены, атакующий может получить доступ к другим ресурсам, используя слабости почтовой системы.

Обобщая, можно сказать, что каждый сервис в сети желательно располагать на отдельном, выделенном сервере, настроенном для выполнения именно этой задачи. Это позволит не только точнее и скорее обнаружить и изолировать нарушителя, но так же снизить возможный ущерб безопасности всей системы.

5.5.3. Все разрешить или все запретить?

Эти две диаметрально противоположные философии могут и должны приниматься во внимание при разработке плана безопасности. Обе имеют право на существование и могут стать основой при разработке плана безопасности. Какую из них принять за основу зависит от того, какую сеть необходимо защищать и насколько жесткой должен быть режим безопасности.

Первый подход состоит в том, чтобы отключить все сервисы, а затем выборочно включать какие-то из них по мере необходимости. Это может быть сделано как на уровне управления программным обеспечением, так и на уровне фильтрации пакетов. Реализация зависит от конкретного случая. Эта модель защиты здесь и далее будет называться моделью «тотального запрещения». Как правило, она уже по сути своей гораздо более безопасна, нежели противоположная модель, описанная в следующем абзаце. Для реализации модели тотального запрещения требуется значительно больше работы, кроме того, требуется более полное понимание сервисов и их значения в работе системы и сети в целом. Разрешение только известных сервисов обеспечивает более полный анализ частных и вспомогательных сервисов и общего построения и реализации механизмов безопасности в данной сети.

Другая модель, приводимая здесь – модель «полного разрешения». Ее реализация много проще, но, как правило, значительно менее безопасна. Достаточно включить все сервисы, как это обычно сделано при установке систем по умолчанию, разрешить всему трафику спокойно перемещаться по сети, как это обычно включено на маршрутизаторах по умолчанию. Если в такой конфигурации обнаруживаются «дыры», то они закрываются на уровне модификации программного обеспечения или путем ограничения трафика.

Каждая из изложенных идеологий может быть применима при построении защиты сети, в зависимости от функциональных требований, административного управления, правил безопасности и т.д. Например, можно использовать модель полного разрешения при установке рабочих станций общего назначения, но модель «тотального запрещения» при установке информационных серверов. Аналогично, модель полного разрешения применима для трафика локальной сети, в то время как модель тотального запрещения более разумно использовать для внешних подключений и Интернет.

Необходимо соблюдать осторожности при таком смешивании двух противоположных подходов. Многие сети и организации на практике предпочитают теорию «жесткой» оболочки и «мягкой» сердцевины. Они готовы платить за мощную защиту своей сети от вторжений извне, за ограничения внешнего трафика и жесткие меры безопасности, в то же время, они не желают позаботиться об элементарной защите внутри собственной сети. Такой подход работает, но только до тех пор, пока защита способна сдерживать атаки извне или

пока собственному персоналу можно полностью доверять. Однако, когда внешняя оболочка (например, брандмауэр) будет «расколота», получить доступ ко всем ресурсам и информации внутри сети не составить ни малейшего труда.

5.5.4. Определение разумной достаточности в сервисах

Существует огромное количество разнообразных сервисов, которые локальная и/или глобальная сеть может предоставлять пользователям. Управление безопасностью, во многих случаях, это управление доступом к ресурсам, расположенным в данной сети и управление доступом к ресурсам, распложенным вне данной сети.

Распространение различных сервисов в сети Интернет подобно волне. В течение многих лет на серверы устанавливались ftp-, gopher-, wais-, www- сервисы только потому, что они становились популярными, но не необходимыми. Проверка всех новых сервисов, которые вы бы хотели установить, с точки зрения их реальной необходимости – это сложная, но крайне важная задача.

Помните, что сложность поддержания безопасности в сети растет экспоненциально с ростом числа поддерживаемых сервисов. Необходимо обновлять программное обеспечение на маршрутизаторах, для поддержки новых протоколов. Многие протоколы по своей внутренней сути сложны для фильтрования и обеспечения безопасности (например, RPC или UDP). Следовательно, обеспечивая больше сервисов, мы все больше открываем внутреннюю сеть для вторжения. Кроме того, сервисы, располагаемые на одной и той же машине, могут порой взаимодействовать с результатами, опасными для сети. Например, если на одной и той же машине предоставляется доступ к anonymous ftp и на этой же машине работает WWW-сервер, возможны ошибки в конфигурации, когда атакующий может поместить произвольные файлы с помощью ftp на ваш сервер, а с помощью www-сервера – запустить их. Какие это может иметь последствия – можно только гадать.

5.5.5. Сервисы и процедуры обеспечения безопасности

В этой главе будут рассмотрены различные моменты, которые необходимо иметь в виду при обеспечении безопасности сети. В каждом параграфе будет рассмотрен определенный сервис безопасности или некоторые свойства сервисов, которые могут использоваться для обеспечения безопасности. В начале каждого параграфа приводится общий обзор рассматриваемой темы.

В течение всей главы довольно часто будет упоминаться криптография или шифрование. В задачи этого документа не входит подробное рассмотрение вопросов криптографии, однако, интересующиеся читатели могут получить более подробную информацию из соответствующих книг или статей.

5.5.5.1 Аутентификация

В течение многих лет основным методом аутентификации было использование обычных текстовых паролей. Первоначально они использовались для аутентификации пользователей, получающих доступ к центральному компьютеру с терминала. В то время, когда сети еще не были столь распространены, риск компрометации такого пароля был минимален. В наше время, когда большинство систем объединены в локальные сети, эти локальные сети связаны между собой в глобальные сети, простые текстовые пароли уже

нельзя считать достаточно безопасными. Пользователи используют ресурсы серверов, расположенных порой даже не в их локальной сети, а за многие сотни и тысячи километров. Их пароли часто передаются по многим сетям, и они крайне уязвимы для перехвата на этом долгом пути. Существует большое количество специальных программ, предназначенных для перехвата простых текстовых паролей.

В настоящее время существую технологии, позволяющие избежать опасности перехвата пароля. Это такие системы как:

- одноразовые пароли (S/Key)
- PGP
- системы аутентификации, построенные на токенах
- и т.д.

Однако, если, например, токены неверно выбраны или защищены, существует угроза безопасности сети.

Одноразовые пароли

Как было сказано выше, в современном сетевом окружении использование простых текстовых паролей нежелательно. В тех сетях, где предъявляются требования к безопасности, необходимо применять более современные технологии. Зафиксировано много случаев атаки сервисов с использованием «троянских коней» (например, измененные программы telnet или rlogin) и сетевых снифферов (программ перехвата и анализа сетевого трафика). Эти программы способны сохранить в особых файлах информацию о пользователях и их паролях. Атакующие способны использовать эту информацию для последующего доступа в систему. Это становится возможным по двум причинам: во-первых, пользователь использует свой пароль многократно, во-вторых, пароль, передаваемый по сети, никоим образом не шифруется.

Существует несколько способов решения этой проблемы. Некоторые из них используют технологию вызова-ответа и обеспечивают аутентификацию паролями, которые используются только единожды – в этой сессии. Существует большое количество систем аутентификации использующих такой подход. Решение использовать тот или иной из них зависит от условий в организации. К сожалению, одноразовые пароли сильно снижают удобство пользователей, поэтому принимать решение об использовании такой схемы аутентификации должно принимать в организации после тщательного тестирования конкретных продуктов.

Kerberos

Kerberos – это распределенная сетевая система аутентификации, которая обеспечивает надежную аутентификацию пользователей и сервисов в небезопасной сетевой среде. Кроме того, если это необходимо для приложения, имеется поддержка контроля за целостностью информации и шифрование. Первоначально система Kerberos была разработана в Массачусетском технологическом институте в середине 80-х годов. На сегодняшний день существует две версии Kerberos – версии 4 и 5, к сожалению, несовместимые между собой.

Kerberos базируется на базе данных симметричных ключей шифрования, расположенной в центре распределения ключей, известном так же как сервер Kerberos. Сервер Kerberos является доверенной третьей стороной имеющей достаточно информации для аутентификации. Для взаимодействия, пользователю и сервису выдаются

криптографические «билеты» после того, как их (сервиса и клиента) легитимность была проверена сервером Kerberos. Эти «билеты» используются для аутентификации клиента и сервера между собой. Каждый «билет» ограничен по времени, то есть он очень быстро устаревает. Следовательно, для правильной работы и клиент и сервис, использующие для аутентификации Kerberos, должны иметь безопасный источник точного времени и иметь возможность точного отсчета времени в системе.

С практической точки зрения Kerberos – это его интеграция с сервисами. Необходимо, чтобы каждый сервис, требующий аутентификации пользователей, имел поддержку Kerberos. В настоящее время существует огромное количество реализаций различных программ, использующих для аутентификации Kerberos. Это сервисы ftp, telnet, pop, NFS и т.д.

Выбор и защита токенов

Если вы решили использовать для аутентификации токены, необходимо подойти к их выбору очень тщательно. Аналогично паролям, токены должны быть способны противостоять прямым методам взлома. Например, ключевая фраза токена не должна состоять из одного слова, особенно, слова из словаря какого-либо языка, не может быть общепринятым или специфическим сокращением. В идеале, ключевая фраза должна быть скорее длинной, нежели короткой, и состоять из символов разных регистров, цифр и прочих символов.

После выбора ключевой фразы, необходимо обеспечить сохранность токена. Аппаратные токены не должны быть подписаны или располагаться рядом с устройством для чтения. Другие, такие как ключи PGP, должны быть защищены от неавторизованного доступа.

Несколько слов о ключах шифрования. При использовании продуктов для шифрования (например, PGP), внимательно отнеситесь к выбору длины ключа и убедитесь, что ваши пользователи понимают важность длины ключа. По мере развития технологии, минимальная безопасная длина ключа все время увеличивается. Старайтесь быть в курсе дел и поддерживать свои знания в области криптографии на современном уровне, чтобы можно было быть уверенным, что тот уровень защиты, на который вы рассчитываете, используя шифрование, действительно обеспечивается.

Обычные пароли

Несмотря на то, что в целях обеспечения безопасности необходимо избавляться от использования обычных многократных паролей, вполне понятно, что многие организации продолжают их использовать. Хоть мы и можем рекомендовать таким организациям применять в своей работе более передовые технологии, мы приведем несколько соображений, призванных помочь в выборе и поддержании традиционных паролей. Однако необходимо помнить, что ни одна из этих мер не обеспечивает защиты от перехвата программами-снифферами.

1. Важность сложных паролей. Во многих (если не во всех) случаях, при атаке системы атакующему необходимо получить доступ к системе, используя какую-либо учетную запись. Одним из путей достижения этой цели является подбор пароля кого-либо их реальных, законных пользователей. Часто для этого применяются программы автоматического подбора пароля, которые используют очень большие словари.

Единственным способом защититься от взлома пароля, таким образом, является тщательный выбор пароля, чтобы исключить возможность подбора. Пароли так же должны быть длинными насколько это позволяет система или может запомнить пользователь.

2. Изменяйте пароли по умолчанию. Многие системы и программное обеспечение устанавливаются с некими учетными записями и паролями по умолчанию. Необходимо немедленно изменить эти пароли и, если возможно, имена учетных записей, чтобы атакующие не могли ими воспользоваться.
3. Ограничьте доступ к файлу с паролями. В частности, администраторы многих систем желают ограничить доступ к файлу, хранящему зашифрованные пароли, таким образом, чтобы возможные атакующие не могли получить к нему доступ. Одной из общепринятых технологий ограничения доступа является использование «затенения» паролей. В этом случае в стандартном для Unix файле /etc/passwd пароли не хранятся, а для их хранения используется файл /etc/shadow. Большинство систем Unix уже сразу после установки используют «затенение» паролей.
4. Устаревание паролей. В течение какого времени и каким образом должны устаревать пароли, до сих пор остается темой обсуждения среди специалистов по информационной безопасности. Общепризнанно, что пароль не должен поддерживаться, если учетная запись более не используется. Но идут горячие дебаты на тему того, должен ли пользователь периодически менять пароль, если он удовлетворяет всем требованиям: длинный, не входящий ни в один словарь и т.д. Аргументом в пользу смены пароля является предупреждение использования учетной записи, если пароль каким-то образом уже был однажды раскрыт или подобран. Оппоненты возражают, что при достаточно частой смене паролей пользователи не запоминают их, что вынуждает их записывать новые пароли, или выбирать простые пароли, которые проще подобрать. Кроме того, если уж атакующий подобрал пароль или перехватил его при передаче по сети, то он, вероятно, воспользуется им немедленно, а не станет откладывать надолго. По этим причинам, смена пароля дает очень небольшую степень защиты, если вообще что-то дает. Поскольку не существует определенного ответа на вопрос о необходимости смены пароля, правила, относящиеся к смене пароля должны прямо давать указания пользователям, насколько часто им следует менять пароль. Обычно периодическая смена паролей не представляет сложностей для пользователей, поэтому имеет смысл все-таки рекомендовать иногда менять пароли. Пароли обязательно надо менять в случае, если учетная запись администратора была скомпрометирована, если произошли серьезные кадровые изменения, особенно – если сменился администратор системы. Необходимо сменить пароль, если учетная запись была скомпрометирована. Если же скомпрометирована была учетная запись администратора, необходимо менять все пароли в системе.
5. Блокирование пароля/ учетной записи. Во многих системах возможно и желательно использование блокирования учетной записи после нескольких неудачных попыток аутентификации. Если вы решите применить такой механизм, помните: он не должен «рекламировать» себя. После того, как учетная запись была заблокирована, даже если будет введен правильный пароль, сообщение должно сообщать, что был введен неверный пароль. Реализация этого механизма требует, чтобы все пользователи, учетные записи

которых оказались заблокированными, обращались напрямую к администратору системы с тем, чтобы разблокировать их.

6. Несколько слов о службе finger. По умолчанию, сервис finger выводит информацию о пользователях, работающих в системе. Например, он может выдавать содержимое файла .plan пользователя. Вся эта информация может использоваться атакующими для определения имени пользователя и подбора пароля. Можно рекомендовать либо запретить использование сервиса finger вообще, либо ограничить информацию, выдаваемую им.

5.5.5.2. Конфиденциальность

В любой сети существует некий набор информации, которую необходимо защитить от возможности неавторизованного доступа. Многие операционные системы имеют встроенные механизмы защиты, которые позволяют администраторам управлять, кто в системе может иметь права на чтение или запись в данный файл. Более жесткий и надежный механизм обеспечения конфиденциальности – шифрование. Шифрование делает доступ к содержимому файла для кого бы то ни было, кроме авторизованных пользователей или владельцев файла, сложным и требующим большого количества времени (например, на расшифровку). Авторизованные пользователи и владельцы файла имеют соответствующие ключи шифрования и могут легко расшифровать файл и получить доступ к его содержимому. Учитывая достижения современной техники шифрования, можно рекомендовать использование шифрования для ограничения доступа к конфиденциальным данным.

Использование алгоритмов шифрования во многих случаях регулируется государством или положениями внутри организации. Соответственно, администраторы должны быть в курсе возможных законодательных ограничений на шифрование, прежде чем использовать его. В задачи данного документа не входит обсуждение программ и алгоритмов шифрования, однако отметим, что стандартная утилита UNIX – crypt – использует очень простой алгоритм, поэтому информация, зашифрованная с ее помощью, может быть легко расшифрована. Кроме того, можно рекомендовать перед тем, как принять решение об использовании того или иного алгоритма или программы, потратить некоторое время на изучение того, насколько хорошую защиту предоставляет этот алгоритм. Большинство широко известных алгоритмов хорошо описаны в литературе, так что это будет несложная работа.

5.5.5.3. Целостность

Как администратор, вы, вероятно, захотите быть уверенным, что ваша информация (например, системные файлы, данные пользователей и т.д.) не была изменена путем неавторизованного доступа. Это означает, что вы хотите обеспечить уверенность в целостности информации в системе. Одним из способов убедиться в этом является вычисление контрольных сумм файлов, хранение их в отдельном, защищенном месте и периодическая сверка с контрольными суммами файлов работающей системы. При этом если файл изменялся, контрольные суммы, естественно, будут различаться.

В некоторых операционных системах есть утилиты, позволяющие вычислять контрольные суммы, например sum в Unix. Однако эти утилиты не обеспечивают защиты как

таковой. Возможно таким образом модифицировать файл, что его контрольная сумма совпадет с неизмененным вариантом. Следовательно, необходимо использовать криптографически стойкую программу, такую как MD5, для подсчета контрольных сумм файлов, если вы хотите использовать их как гарантию целостности системы.

Существуют так же иные приложения, для которых необходим контроль целостности, например, пересылка электронной почты. Существует множество продуктов, обеспечивающих такую возможность. Для выбора необходимо сначала четко определить, какие именно свойства могут быть важны в данном случае, а уж потом производить отбор продуктов по сформулированным критериям.

5.5.5.4. Авторизация

Авторизацией называется процесс предоставления прав доступа и привилегий пользователям и их процессам. Отличие авторизации от аутентификации состоит в том, что аутентификация – это процесс подтверждения того, что пользователь действительно является тем, за кого он себя выдает. После надежной аутентификации, права и привилегии пользователя определяются авторизацией.

Составить полный список авторизованных действий пользователей над системными объектами, пожалуй, невозможно ни в одной реальной системе. Для упрощения управления системными объектами возможно применение нескольких технологий.

Один подход, традиционный для UNIX-систем, состоит в том, что с каждым объектом связываются три класса доступа – владелец, группа и прочие. Владельцем файла является либо создатель файла, либо назначенный администратором пользователь. Права доступа владельца (чтение, запись и исполнение) применяются только к владельцу файла. Группа – это набор из нескольких пользователей, разделяющих равные права на доступ к объекту. Права доступа для группы действуют только на членов группы (исключая владельца файла). Наконец права для прочих пользователей применяются для всех пользователей, исключая владельца и тех, кто входит в группу-владельца.

Другой подход состоит в создании для каждого файла списка, в котором полностью перечислены пользователи и их права доступа к данному файлу. Такой список называется списком управления доступом. Преимущество списка управления доступом в том, что им довольно легко управлять и поддерживать, кроме того, очень легко проверить, кто какие права доступа имеет по отношению к данному файлу. Недостаток состоит в том, что для хранения списков управления доступом необходимо дополнительное пространство, тем более, что в некоторых случаях эти списки могут быть достаточно большими.

5.5.5.5. Доступ

Физический доступ

Ограничьте физический доступ к машинам, разрешив доступ только тем людям, которые должны работать на конкретной машине. Термин «машина» в данном контексте подразумевает все терминалы, подключенные к серверу, если в организации используются таковые, а так же – персональные компьютеры и рабочие станции, подключенные к вашей сети. Особенно следует позаботиться о защите консолей серверов, поскольку некоторые действия возможно выполнить с консоли без авторизации. Убедитесь, что рабочие места

пользователей имеют ограниченный доступ. В противном случае они могут стать причиной и местом нарушения режима безопасности.

Следует обратить внимание на аккуратное и безопасное хранение оригиналов и архивных копий программ и данных. Кроме того, что их следует хранить в требуемых условиях для обеспечения сохранности со временем, они еще должны быть защищены от кражи. Важно хранить резервные копии отдельно от оригиналов не только в целях сохранности в случае какой-либо опасности или происшествия, но так же на случай возможной кражи.

Переносные компьютеры и ноутбуки представляют собой отдельную проблему. Убедитесь, что утрата какого-либо переносного компьютера не создаст проблем с безопасностью информации. Необходимо четко определить, какого типа данные могут размещаться на дисках переносных компьютеров, и каким образом эти данные будут защищены. Вероятно, и тут хорошим решением окажется использование шифрования.

Так же зонами со строго ограниченным доступом должны являться коммутационные комнаты и жизненно важные элементы сети – серверы, маршрутизаторы, коммутаторы и концентраторы.

Точки подключения мобильных пользователей

Под точками подключения мобильных пользователей мы понимаем те места в сети, где возможно подключение к сети переносных компьютеров и ноутбуков.

Помните, что если вам необходимо обеспечить возможность подключения к сети переносных компьютеров, то это дает возможность пользователям подключать к сети организации неавторизованное и непроверенное оборудование. Это увеличивает риск подвергнуться такого вида атакам, как IP-спуфинг (подмена адресов), сниффинг и т.д. И пользователям и администраторам сети необходимо четко определись степень допустимого риска. Если все-таки будет принято решение о предоставлении возможности подключения переносных компьютеров, необходимо тщательно спланировать и четко определить в каких именно точках данный тип подключения возможен. Кроме того, необходимо позаботиться о физическом ограничении доступа к таким местам.

Переносной компьютер должен быть аутентифицирован, прежде чем получить какой-либо доступ к ресурсам сети. Как альтернатива, возможно просто жесткое ограничение доступа к точкам такого подключения. Например, если необходимо обеспечить подключение переносных компьютеров студентов, разъемы для подключения должны располагаться только в студенческих лабораториях.

Если подключение к сети переносных компьютеров используется для обеспечения подключения компьютеров гостей вашей организации (например, чтобы они могли прочесть свою электронную почту и т.д.), используйте для этих целей выделенную подсеть, по возможности не связанную с сетью вашей организации.

Постоянно держите в поле зрения области, в которых возможно подключение к сети вашей организации. Такие, например, как пустующие офисы. Возможно, будет разумным, отключить их от сети в коммутационной комнате.

5.5.5.6. Модемы

Модемные линии всегда должны управляться

Поскольку модемные линии обеспечивают пользователям обычный доступ в сеть, они так же могут обеспечивать эффективный путь обхода брандмауэра организации. Одного этого уже достаточно, чтобы обратить особое внимание на управление модемными линиями.

Не позволяйте пользователям самостоятельно устанавливать модемы, поскольку, как правило, при этом не будет настроено строгой аутентификации и авторизации. Это правило касается и временных установок, таких как, например, подключение модема на ночь.

Постоянно следите и регистрируйте все имеющиеся и появляющиеся модемные линии. Желательна так же регулярная проверка на сети на предмет наличия самостоятельно установленных модемов.

Все пользователи, получающие доступ по модему, должны аутентифицироваться

Прежде чем пользователь получит доступ к ресурсам сети, необходимо чтобы он прошел аутентификацию с помощью ввода имени и пароля. При этом условия, описанные ранее для обычных текстовых паролей, здесь приобретают особое значение.

Помните, что телефонные линии могут прослушиваться и что перехватить сообщения или сигналы, передаваемые по сотовому телефону, очень просто. Современные высокоскоростные модемы используют довольно сложные алгоритмы модуляции, которые трудно расшифровать, даже если иметь возможность прослушивания линии. Однако не стоит надеяться, что атакующий не в состоянии этого сделать. Наиболее оптимальным способом будет, пожалуй, использование одноразовых паролей.

Возможно, существенно легче использовать один большой модемный пул для всех входящих звонков, поскольку в этом случае проще осуществлять слежение за аутентификацией и активностью пользователей. Кроме того, проще поддерживать единый механизм аутентификации для всех.

Иногда пользователи могут допускать ошибки при вводе пароля. Установите небольшую задержку, скажем, в 2 секунды, между первой и второй неудачными попытками ввода пароля, а после третьей – разрыв связи. Это сильно усложнит и замедлит использование автоматизированных средств подбора пароля. Не стоит так же сообщать пользователю, что же именно – имя или пароль, было введено неверно.

Возможность обратного дозвона

Некоторые серверы удаленного доступа предоставляют возможность обратного дозвона. Это означает, что после того, как пользователь был аутентифицирован, сервер разрывает связь и дозванивается сам по некоторому номеру. Возможность обратного дозвона хороша тем, что при компрометации пароля пользователя происходит разрыв связи и сервер дозванивается к тому пользователю, чей пароль был подобран. Наилучшим же выбором, пожалуй, является возможность случайного дозвона. В этом случае, в некоторых случаях обратный дозвон происходит, а в некоторых – нет. Это определяется случайным образом. Такая настройка сервера удаленного доступа подразумевает, что пользователь может пользоваться возможностью удаленного доступа к сети только из одного места – с номера, зарегистрированного на сервере удаленного доступа.

Возможностью обратного дозвона следует пользоваться с осторожностью, поскольку есть возможность обойти эту меру безопасности. Как минимум, убедитесь, что обратный

звонок никогда не делается с того же самого телефона, на который попал входящий звонок. В целом, возможность обратного дозвона значительно увеличивает безопасность соединений по модему, но не стоит полагаться только на обратный дозвон.

Использование модемных линий должно полностью журналироваться

Все подключения по модемным линиям, успешные или неуспешные, должны журналироваться. Однако это следует отметить отдельно, в журналы никогда не должны попадать пароли. Необходимы просто отметки об успешной аутентификации и соединении. Так как большинство неверных паролей – это просто опечатки пользователей, отличающиеся от настоящих паролей одним-двумя символами, запись их в журналы может помочь атакующему подобрать пароль. Поскольку полностью гарантировать безопасность журнала от просмотра невозможно, лучше не записывать неверные пароли вообще.

Если есть возможность определения номера звонящего, лучше всего использовать ее для записи в журнал каждого номера, с которого происходила попытка соединения – успешная или нет. Следует обратить особое внимание на вопросы частной жизни, которые, быть может, окажутся затронутыми процедурой определения номера. Кроме того, помните, что информация, выдаваемая определителем номера, не всегда является точной и поэтому не может считаться «надежной». Использование этих данных возможно только для информации, но никаким образом не для аутентификации.

Обратите особое внимание на заставку

Во многих системах в качестве заголовка перед приглашением к вводу имени пользователя используется стандартный системный заголовок. К сожалению, во многих системах он содержит информацию об оборудовании, на котором функционирует система, типе системы и т.д. Это может снабдить атакующего некоей начальной информацией. Желательно в каждой сети для каждой системы создать свой собственный заголовок, обращая особое внимание на его содержимое. Заголовок должен содержать только необходимую информацию.

Создайте короткий заголовок, однако не включайте в него информацию о том, где данная система установлена – название организации, отдела и т.д. Вместо этого выдайте имя компьютера, короткое предупреждение о том, что сессия связи может быть под наблюдением и приглашение к вводу имени и пароля.

Аутентификация исходящих звонков

Пользователи, делающие исходящие звонки, так же должны быть аутентифицированы, хотя бы потому, что вашей организации придется оплачивать их телефонные счета.

Никогда не позволяйте неаутентифицированным пользователям производить исходящие звонки. И обратите внимание на то, позволить ли это делать пользователям вообще. Целью этих мер является предотвратить использование вашего модемного пула в цепочке подключений взломщиков. Факт такого использования довольно сложно обнаружить, особенно, если уже в вашей сети используется цепь соединений. Как минимум, не позволяйте использовать одни и те же телефонные линии для входящих и исходящих соединений. Это несложно реализовать, если использовать различные модемные пулы для входящих и исходящих звонков.

Проверяйте конфигурацию модемов

Убедитесь, что ваши модемы не были перепрограммированы каким-либо образом, пока находились в ремонте или обслуживании, а так же при покупке. Как минимум, проверьте, что последовательность «+++» не переводит модемы в пуле входящих звонков в командный режим.

Желательно запрограммировать модемы и программное обеспечение таким образом, чтобы они сбрасывали установки модема на установки «по умолчанию» при обработке каждого нового звонка. Если это по каким-либо причинам не удается сделать, пусть такой сброс происходит после обработки звонка. Это защитит модемы от возможного перепрограммирования. Сброс модема до и после обработки очередного звонка позволит защититься от возможности перехвата сессии одного пользователя другим после того, как предыдущий пользователь по каким-либо причинам отключился от линии.

Убедитесь, что ваши модемы корректно обрабатывают прекращение связи. После того, как пользователь отключился от сервера удаленного доступа, сервер должен немедленно освободить телефонную линию. В равной степени важно, чтобы сервер удаленного доступа закрывал все соединения пользователя, если последний неожиданно отключился от линии.

5.5.5.7. Аудит

Этот раздел описывает процедуры сбора данных об активности в сети, которые могут использоваться для анализа безопасности сети и обнаружения и анализа инцидентов нарушения режима безопасности.

Что имеет смысл регистрировать?

Данные аудита должны содержать все попытки получить определенный уровень доступа лицами, процессами или любыми другими элементами сети. Сюда входят: регистрация входа и выхода пользователей, получение привилегий суперпользователя, генерация ключей аутентификации (например, для сервиса Kerberos), а так же любые иные изменения доступа или статуса пользователей. Особенно важно иметь информацию о так называемых «гостевых» или анонимных подключениях.

Реальные потребности в сборе информации могут сильно различаться для различных сетей и ситуаций. В общем случае следует обеспечить сбор и хранение следующей информации: имя пользователя и имя машины - для регистрации входов и выходов; начальные и новые права доступа - при изменении прав доступа; а так же время изменений или входа/выхода. Разумеется, возможна регистрация значительно большего количества информации, в зависимости от того, что именно позволяет конкретная система и каков объем доступного пространства для хранения данных аудита.

Важное замечание: никогда не следует сохранять в журналах данные о паролях! Это создает очень серьезную брешь в защите в случае, если информация системных журналов кем-то будет прочитана. Не следует сохранять и информацию о неверных паролях. Как правило, неверные пароли, вводимые пользователями – это просто ошибки легальных пользователей, отличающиеся от настоящих паролей одним-двумя знаками.

Процесс сбора информации

Процесс сбора информации в общем случае зависит от того, о доступе к какому сервису или на какой адрес должна быть собрана информация. В зависимости от важности этих данных и от потребности хранить их в том же месте, где располагается сам сервис,

данные могут либо храниться непосредственно на сервере, предоставляющем сервис, либо на удаленной машине. Хранить журналы удаленно или нет, зависит так же от важности журнальной информации.

Традиционно существуют три способа хранения файлов журналов. Во-первых, обычный файл, во-вторых, файл на устройстве типа WORM (write once – read many), либо на устройстве «только для записи», например, принтере. Каждый метод имеет свои преимущества и недостатки.

Хранение журналов в виде обычного файла – это наиболее часто используемый на практике способ, к тому же, наиболее простой в настройке. Как правило, операционные системы и сервисы по умолчанию используют этот способ хранения журналов. Он позволяет легко получать доступ к журнальной информации и производить ее анализ. Это может оказаться важным при обнаружении атак. Вместе с тем, это наименее надежный способ. Если машина, занимающаяся хранением журнальной информации будет скомпрометирована, атакующий может легко исправить информацию системных журналов, и тем самым, скрыть следы взлома.

Хранение журнальной информации на устройствах с однократной записью (WORM) ненамного сложнее в настройке, но существенно более надежно, поскольку атакующий не сможет изменить информацию системных журналов, свидетельствующую о взломе. Недостатком этого метода является необходимость замены дисков или лент, и слежения за их заполнением. Кроме того, этот метод дороже, так как накопители и носители WORM стоят каких-то денег. Наконец, возможно, доступ к данным для их анализа будет несколько затруднен.

Использование принтера для хранения журнальной информации применимо в том случае, если необходимо постоянное и немедленное слежение за журнальной информацией. Примером таких систем могут быть системы реального времени, когда необходимо точно зафиксировать атаку. Использование для ведения журналов лазерных принтеров в этом случае крайне нежелательно, так как лазерные принтеры имеют очень большой буфер и не фиксируют журнальную информацию немедленно при поступлении. Лучшим выбором тут, очевидно, являются матричные принтеры. Существенным недостатком это способа хранения журнальной информации являются горы бумаги, ежедневно выдаваемые принтером и необходимость просмотра и анализа этой информации вручную.

Для каждого из перечисленных способов хранения журнальной информации существует несколько моментов, важных с точки зрения обеспечения безопасности пути доставки журнальной информации до конечного устройства хранения (принтера или привода CD-ROM). Если путь доставки журнальной информации будет скомпрометирован, возможно нарушение целостности журнальной информации. В идеальном случае, устройство хранения журнальной информации должно быть подключено одним цельным кабелем типа «точка-точка». На деле же, такая ситуация вряд ли достижима. Поэтому необходимо позаботиться о минимизации пути проходимого журнальной информацией до устройства хранения. Особенно это касается серверов хранения журнальной информации. Путь от клиента до сервера, на котором он хранит журнальную информацию, должен проходить через минимальное количество подсетей и маршрутизаторов. Кроме того, желательно использовать механизм электронной подписи журнальной информации, чтобы избежать нарушения целостности. Использование же шифрования журнальной информации при

передаче, пожалуй, излишне, поскольку журналы не хранят жизненно важной и секретной информации.

Позаботьтесь о достаточном месте для хранения журналов

Системные журналы имеют свойство довольно быстро увеличиваться в объеме. Поэтому, чтобы предотвратить нарушения работы системы журналирования, необходимо позаботиться о достаточном количестве доступного пространства для хранения журнальной информации. Существует несколько способов уменьшить количество требуемого места. Во-первых, данные системных журналов могут сжиматься с использованием различных алгоритмов. Во-вторых, возможно минимизировать требуемое пространство путем хранения полных журналов за короткий период времени, например, сутки. По окончании этого периода, происходит обработка данных журналов и далее хранится уже обработанная информация, а не журнал целиком. Существенным недостатком такого метода является затруднение расследования инцидента нарушения режима безопасности. Часто проходит некоторое время после случая нарушения режима безопасности, прежде чем администраторы или пользователи заметят его и начнут реагировать и расследовать. Если же нет полной информации в системных журналах, то проведение расследования может быть сильно затруднено.

Хранение и защита данных системных журналов

Данные аудита должны тщательно храниться и защищаться. Необходимо регулярное резервное копирование данных системных журналов. Если атакующий получает доступ к данным журналов, то под угрозой оказываются не только данные, хранимые в информационной системе, но сама система в целом.

Содержимое системных журналов является ключевым для анализа инцидента нарушения безопасности, причин, по которым это стало возможно, и для выработки мер по недопущению таких инцидентов в будущем. По этим причинам будет предусмотрительно более тщательно проанализировать угрозы данным такого рода и принять меры для снижения соответствующих рисков. Разумеется, такая работа должна быть проделана до того, как произойдет инцидент нарушения безопасности.

Если план защиты данных аудита не будет разработан и реализован до инцидента нарушения безопасности, тогда, вероятно, не будет способов восстановить картину происшедшего уже после такого события, а значит, не будет возможности оценить причины происшедшего и сделать какие-то выводы.

Принимайте во внимание действующее законодательство

Если в дальнейшем, при расследовании инцидента нарушения режима безопасности вы планируете использовать действующие законы, вероятно, вам следует проконсультироваться с юристами организации о содержимом системных журналов и порядке их ведения, чтобы их данные могли дать какую-то информацию следственным органам. Если вы ведете системные журналы, вы должны быть готовы для последующего анализа их содержимого.

Кроме того, в некоторых случаях, данные системных журналов могут нарушать права граждан (работников) на тайну, поскольку будут содержать некоторые частные данные (например, номера телефонов, с которых были произведены входы в вашу сеть, если у вас действует определение номера). Процедура анализа или поиска по системным журналам,

таким образом, может вмешаться в личную жизнь работников. Этот аспект так же нуждается в рассмотрении юристов с точки зрения законодательства.

Другим примером, когда необходимо принимать во внимание законодательство, является анализ атак, возможно, происходящих из сети вашей организации. Если организация ведет системные журналы, должна ли она проводить их анализ на предмет выявления таких случаев? Если какой-либо адрес из вашей сети использовался для действий против другой организации, может ли пострадавшая сторона использовать данные ваших системных журналов для нахождения злоумышленника? Вот только несколько вопросов имеющих отношение к данным системных журналов, которые необходимо проработать совместно с юристами вашей организации.

5.5.5.8. Безопасность резервных копий

Процедура создания резервных копий – это классическая часть поддержания компьютерных систем и данных. В контексте настоящего документа, резервные копии рассматриваются как часть общего плана безопасности. Есть несколько моментов, связанных с резервными копиями, относящихся к вопросам безопасности:

- Убедитесь, что в вашей сети регулярно выполняется резервное копирование данных.
- Используйте стороннее хранилище для хранения резервных копий. Убедитесь, что оно является надежным и доступным в случае необходимости.
- Используйте шифрование для обеспечения дополнительной безопасности резервных копий, хранимых сторонней организацией. Однако при этом имейте в виду, что потребуется хорошая схема хранения и учета ключей шифрования, чтобы в дальнейшем была возможность восстановить данные с резервных копий. Так же убедитесь, что у вас есть резервные копии программ шифрования, чтобы они так же были доступны при необходимости.
- Никогда не полагайтесь целиком на надежность резервных копий. Существует довольно много случаев, когда неправомерный доступ к информации продолжался довольно долго, прежде чем был обнаружен. В таком случае, резервные копии так же могут содержать программы и данные с нарушениями целостности.
- Периодически проверяйте возможность и процедуру восстановления с резервных копий.

5.6. Реакция на нарушение режима безопасности

5.6.1. Обзор

В данной главе излагаются соображения, применимые к ситуациям, когда происходит нарушение информационной безопасности отдельного компьютера, сети, организации или корпоративной среды. Основное положение состоит в том, что враждебные действия, будь то атака внешних злоумышленников или месть обиженного сотрудника, необходимо предусмотреть заранее. Ничто не может заменить предварительно составленного плана восстановительных работ. Традиционная информационная безопасность, хоть и имеющая весьма большое значение для общеорганизационных защитных планов, как правило, концентрируется вокруг защиты от атак и, до некоторой степени, вокруг их обнаружения.

Обычно почти не уделяют внимания мерам, предпринимаемым, когда атака уже идет. В результате поспешных, непродуманных действий могут быть затруднены выявление причины инцидента, сбор улик для расследования, подготовка к восстановлению системы и защита ценной информации.

5.6.1.1. Имейте план, которому будете следовать во время инцидента

Частью реакции на нарушения безопасности является предварительная подготовка ответных мер. Под этим понимается поддержание должного уровня защиты, так что ущерб даже от серьезного инцидента будет ограниченным. Подготовка включает в себя составление руководства по мерам реагирования на инциденты и плана восстановительных работ. Наличие отпечатанных планов способно устранить многие двусмысленности, возникающие во время инцидента, и ведет к серии более точных и основательных ответов. Далее, частью защиты является выработка процедуры извещения об инциденте, чтобы каждый знал, кто кому звонит и по каким номерам. Целесообразно устраивать "учебные тревоги", когда сотрудники службы безопасности, системные администраторы и руководители отрабатывают реакцию на инциденты.

Отработка эффективных ответов на инциденты важна по многим причинам. Главнейшая из них – чисто человеческая: предотвращение угрозы жизням людей. Некоторые компьютерные системы критически важны для сохранения жизней (например, системы жизнеобеспечения в больницах или комплексы, участвующие в управлении движением воздушных судов).

Еще одно существенное достоинство предварительной подготовки, о котором часто забывают, носит экономический характер. Содержание технического и управленческого персонала, ответственного за реакцию на инциденты, требует значительных ресурсов, которые с выгодой можно было бы употребить на другие нужды. Если персонал обучен эффективным приемам реагирования, обслуживание инцидентов будет отнимать меньше времени.

Третье достоинство - обеспечение защиты секретной, критически важной или частной информации. Весьма опасно то, что компьютерный инцидент может разрушить невосстановимую информацию. Эффективная реакция на инциденты минимизирует эту опасность. Когда речь идет о секретной информации, следует учесть и включить в план соответствующие правительственные постановления. Четвертое достоинство касается связей с прессой. Сведения о компьютерном инциденте могут повредить репутации организации среди нынешних или потенциальных клиентов. Эффективная реакция на инцидент уменьшает вероятность нежелательной огласки.

Наконец, упомянем правовой аспект. Можно представить себе ситуацию, когда организация подвергается судебному преследованию, поскольку один из принадлежащих ей узлов был использован для атаки на сеть. С похожими проблемами могут столкнуться люди, реализующие заплаты или надстройки, если те оказались неэффективными и не смогли предотвратить ущерб или сами стали причиной ущерба. Знание уязвимых мест операционных систем и типичных приемов атаки, а также принятие превентивных мер поможет избежать конфликтов с законом.

5.6.1.2. Порядок изложения в данной главе можно использовать в качестве плана

Данная глава организована таким образом, что ее содержание может послужить отправной точкой при написании политики безопасности, касающейся реакции на инциденты. В политике должны быть освещены следующие темы:

1. Обзор (цели, преследуемые политикой безопасности в плане реакции на инциденты).
2. Оценка (насколько серьезен инцидент).
3. Извещение (кого следует известить об инциденте).
4. Ответные меры (что следует предпринять в ответ на инцидент).
5. Правовой аспект (каковы правовые последствия инцидента).
6. Регистрационная документация (что следует фиксировать до, во время и после инцидента).

Каждая из перечисленных тем важна при общем планировании реакции на инциденты. Оставшаяся часть главы посвящена их подробному изложению. Будут сформулированы рекомендации по формированию политики безопасности, касающейся реакции на инциденты.

5.6.1.3. Возможные цели и побудительные мотивы эффективной реакции на инциденты

Как и во всякой деятельности по планированию, в первую очередь необходимо уяснить преследуемые цели. Эти цели следует упорядочить в порядке убывания важности. Итоговый список, конечно, будет разным для разных организаций. Ниже приведен один из возможных вариантов:

- Гарантировать целостность критических важных (для сохранения человеческих жизней) систем.
- Сохранить и восстановить данные.
- Сохранить и восстановить сервисы.
- Выяснить, почему инцидент стал возможен.
- Предотвратить развитие вторжения и будущие инциденты.
- Избежать нежелательной огласки.
- Найти виновников.
- Наказать нарушителей.

Важно заранее определить приоритеты действий, совершаемых во время инцидента. Бывают столь сложные случаи, когда невозможно одновременно принять все необходимые ответные меры; без учета приоритетов тут не обойтись. Хотя, как всегда, шкала приоритетов зависит от организации, следующий список может послужить отправной точкой при выработке иерархии ответных мер.

Первый приоритет - защитить жизнь и здоровье людей; при всех обстоятельствах защита человеческих жизней должна стоять на первом месте.

Второй приоритет - защитить секретные и/или критически важные данные (в соответствии с правительственными или организационными нормами).

Третий приоритет - защитить прочие данные, включая частную, научную и управленческую информацию, поскольку потеря данных дорога с точки зрения ресурсов, затраченных на их накопление.

Четвертый приоритет - предотвратить повреждение систем (потерю и изменение системных файлов, повреждение дисководов и т.п.) чтобы избежать дорогостоящих простоев и восстановлений.

Пятый приоритет - минимизировать урон, нанесенный вычислительным ресурсам; во многих случаях лучше выключить систему или отсоединить ее от сети, чем подвергать риску информацию, программное обеспечение или аппаратуру.

Важным следствием определения приоритетов является то, что, после человеческих жизней и интересов государственной безопасности, наиболее ценным активом обычно являются данные, а не программное или аппаратное обеспечение. Хотя нежелательны любые потери, системы можно заменить; в то же время потерю или компрометацию данных (особенно секретных), как правило, нельзя допускать ни при каких обстоятельствах.

Как уже отмечалось, частью реакции на инциденты является предварительная подготовка ответных мер. Для каждой машины и системы должна существовать и выполняться процедура резервного копирования. Наличие копий в значительной степени устраняет потери даже после серьезных инцидентов, поскольку исключаются массовые потери данных. Далее, Ваши системы должны иметь безопасную конфигурацию. Под этим понимается устранение слабостей, проведение эффективной политики управления паролями, а также использование других процедур, разъясняемых далее.

5.6.1.4. Руководство по местной политике безопасности и юридическим положениям

Любой план реагирования на инциденты должен составляться на основе политики безопасности и юридических положений. Правительственные и частные организации, имеющие дело с секретной информацией, должны следовать дополнительным правилам.

Политика, разработанная Вашей организацией применительно к реакции на нарушения режима безопасности, позволит оформить ответные меры. Например, нет особого смысла создавать механизмы для отслеживания нарушителей, если Ваша организация не собирается после поимки предпринимать против них какие-либо действия. На Ваши планы может влиять политика других организаций. Например, телефонные компании обычно сообщают информацию для прослеживания звонков только правоохранительным органам.

Если Вы собираетесь предпринимать правовые акции, необходимо следовать особым рекомендациям, чтобы собранная Вами информация могла быть использована в качестве свидетельских показаний.

5.6.2. Оценка

5.6.2.1. А что на самом деле?

На этой фазе точно выясняется характер проблем. Конечно, многие, если не большинство, проявлений, часто приписываемых вирусным инфекциям или вторжениям злоумышленников, являются следствием обычных отклонений, таких как аппаратные сбои. Чтобы понимать, действительно ли имеет место нарушение режима безопасности, полезно приобрести и использовать специальное программное обеспечение. Например, широко доступные программные пакеты могут оказать существенную помощь в выявлении вируса, проникшего в Macintosh. Весьма полезна и регистрационная информация, особенно применительно к сетевым атакам. При подозрениях на вторжение чрезвычайно важно

сделать моментальный снимок системы. Многие инциденты порождают целую цепь событий, и снимок системы, сделанный на начальной стадии, может оказаться полезнее других мер для установления сути проблемы и источника опасности. Наконец, важно завести регистрационную книгу. Запись системных событий, телефонных разговоров, временных меток и т.д. способна ускорить и систематизировать процесс идентификации проблемы, послужить основой последующих действий по нейтрализации инцидента.

Имеется ряд отчетливых признаков, или "симптомов" инцидента, заслуживающих особого внимания:

- Крахи системы.
- Появление новых пользовательских счетов (например, необъяснимым образом создался счет RUMPLESTILTSKIN) или необычайная активность со стороны пользователя (счета), практически не подававшего признаков жизни в течение нескольких месяцев.
- Новые файлы (обычно со странными именами, такими как data.xx или k).
- Рассогласования в учетной информации (например, на UNIX-системах это может проявляться как сокращение файла /usr/admin/lastlog, что вызывает сильные подозрения в присутствии нарушителя).
- Изменения в размерах и датах файлов (например, пользователя MS-DOS должно насторожить внезапное удлинение .EXE-файла более чем на 1800 байт).
- Попытки записи в системные файлы (например, системный администратор замечает, что привилегированный пользователь VMS пытается изменить RIGHTSLIST.DAT).
- Модификация или удаление данных (например, начали исчезать файлы).
- Отказ в обслуживании (например, системные администратор и все остальные пользователи оказались выброшенными из UNIX-системы, которая перешла в однопользовательский режим).
- Необъяснимо низкая производительность системы (например, необычно плохое время отклика системы).
- Аномалии (например, на экране терминала вдруг появляется слово GOTCHA, или раздаются частые и необъяснимые звуковые сигналы).
- Подозрительные пробы (например, многочисленные неудачные попытки входа с другого узла сети).
- Подозрительное рысканье (например, некто стал пользователем root UNIX-системы и просматривает файл за файлом).

Ни один из этих признаков не может служить бесспорным доказательством нарушения режима безопасности, точно так же, как реальный инцидент обычно не сопровождается всем набором симптомов. Если, однако, Вы заметили какой-либо из перечисленных признаков, следует подозревать нарушение и действовать соответственно. Не существует формулы, позволяющей с абсолютной достоверностью обнаруживать инциденты.

Пожалуй, единственным исключением являются антивирусные пакеты. Если они говорят, что вирус есть, им можно верить. В такой ситуации лучше всего воспользоваться помощью других технических специалистов и сотрудников службы информационной безопасности и сообща решить, действительно ли инцидент имеет место.

5.6.2.2. Масштабы инцидента

Идентификации инцидента сопутствует выяснение его масштабов и возможных последствий. Для эффективного противодействия важно правильно определить границы инцидента, Кроме того, оценка возможных последствий позволит установить приоритеты при выделении ресурсов для принятия ответных мер. Без выяснения масштабов и возможных последствий события трудно определить, как именно нужно действовать. Для определения масштабов и возможных последствий, следует воспользоваться набором критериев, подходящих для конкретной организации и имеющихся связей с внешним миром. Вот некоторые из них:

- Затрагивает ли инцидент несколько организаций?
- Затрагивает ли инцидент многие компьютеры Вашей организации?
- Находится ли под угрозой критически важная информация?
- Какова стартовая точка инцидента (сеть, телефонная линия, локальный терминал и т.д.)?
- Знает ли об инциденте пресса?
- Каков потенциальный ущерб от инцидента?
- Каково предполагаемое время ликвидации инцидента?
- Какие ресурсы требуются для ликвидации инцидента?

5.6.3. Возможные типы извещений

Когда Вы убедились, что нарушение режима безопасности действительно имеет место, следует известить соответствующий персонал. Чтобы удержать события под контролем как с технической, так и с эмоциональной точки зрения, очень важно, кто и как будет извещен.

5.6.3.1. Внятность

Прежде всего, любое извещение, направленное своему или стороннему сотруднику, должно быть внятным. Это значит, что любая фраза об инциденте (идет ли речь об электронном сообщении, телефонном звонке или факсе) обязана быть ясной, точной и полной. Всякий "туман" в извещении, направленном человеку, от которого Вы ждете помощи, отвлечет его внимание и может повести к недоразумениям. Если предлагается разделение труда, полезно снабдить каждого участника информацией о том, что делают другие. Это не только уменьшит дублирование, но и позволит человеку, занятому определенной работой, знать, где получить дополнительные сведения, чтобы справиться со своей частью проблемы.

5.6.3.2. Правдивость

Другой важный аспект извещений об инциденте - правдивость. Попытки скрыть отдельные моменты, сообщая ложную или неполную информацию, способны не только помешать принятию эффективных ответных мер; они могут повести даже к ухудшению ситуации. Это тем более верно в случае, когда об инциденте узнали журналисты. Если имеет место достаточно серьезный инцидент, привлекший внимание прессы, то, скорее всего, любая сообщенная Вами ложная информация не получит подтверждения из других источников. Это бросит тень на организацию и испортит отношения с журналистами, а, значит, и с общественностью.

5.6.3.3. Выбор языка

Язык, которым написано извещение, существенным образом влияет на восприятие информации об инциденте. Если Вы используете эмоциональные обороты, Вы увеличиваете ощущение опасности и ожидание неблагоприятного завершения инцидента. Важно сохранять спокойствие и в письменных, и в устных извещениях.

Другим моментом, связанным с выбором языка, является извещение нетехнического и внешнего персонала. Важно точно описать инцидент, без лишней тревоги и непонятных фраз. Хотя неспециалистам объяснить суть дела труднее, зачастую это более важно. Нетехническое описание может понадобиться для высшего руководства, прессы или сотрудников правоохранительных органов. Важность подобных извещений нельзя недооценивать. От этого зависит, получит ли инцидент адекватное решение или приведет к еще более серьезным последствиям.

5.6.3.4. Извещение конкретных лиц

Кого извещать во время и после инцидента? На этот предмет можно рассмотреть несколько категорий лиц.

Персонал в точках контакта (техническая и административная группы, группа реагирования, органы дознания, другие правоохранительные органы, производители, поставщики услуг). Необходимо определить, кто отвечает за извещения в адрес каждой из перечисленных контактных групп.

Более широкое сообщество (пользователи).

Другие организации, вовлеченные в инцидент.

Следует заранее установить, кого извещать из центральной точки контакта организации (см. также п. 5.3.6). Список лиц в каждой из выбранных категорий поможет сэкономить массу времени в случае нарушения режима безопасности. В суете инцидента, когда срочные дела накладываются друг на друга, очень трудно выяснять, где и кого можно отыскать.

Кроме лиц, отвечающих за определенные аспекты реакции на инциденты, в извещении нуждаются другие организации, которых нарушение затронуло или может затронуть. Пользователям зачастую также полезно знать об инциденте. Им разумно направить отчет о нарушении (если этот отчет решено сделать открытым).

5.6.3.5. Связи с общественностью - пресс-релизы

Один из самых важных вопросов - когда, кто и насколько много должен сообщить общественности через прессу. При этом следует учитывать несколько моментов. Во-первых, если в организации существует пресс-центр, важно задействовать именно его. Сотрудники пресс-центра имеют опыт общения с журналистами, и это поможет сохранить лицо организации во время и после инцидента. С сотрудниками пресс-центра можно говорить откровенно, они сами буферизуют предназначенную для прессы информацию, а Вы в это время сможете заниматься инцидентом.

Если пресс-центра нет, следует тщательно взвешивать сообщаемые прессе сведения. Если информация конфиденциальна, разумно ограничиться минимумом данных обзорного характера. Весьма возможно, что все, сообщенное прессе, быстро дойдет до виновника инцидента. С другой стороны, как отмечалось выше, введение прессы в заблуждение может

оказаться бумерангом, наносящим больший вред, чем разглашение конфиденциальной информации.

Хотя заранее сложно определить, насколько детальные сведения стоит сообщать прессе, разумно учесть следующие соображения.

Избегайте технических деталей. Детальная информация об инциденте может повести к повторению подобных нарушений или даже помешать организации расследовать текущий случай.

Избегайте предположений. Предположения о виновнике инцидента и его побудительных мотивах могут оказаться ошибочными, что способно усугубить ситуацию.

Работайте с профессионалами из правоохранительных органов, чтобы обеспечить защиту улик. Если в деле участвуют следственные органы, убедитесь, что собранные улики не стали достоянием прессы.

Избегайте интервью, если Вы не готовы к ним. Помните, что журналисты попытаются вытянуть из Вас максимум информации, в том числе конфиденциальной.

Не позволяйте прессе отвлекать Ваше внимание от реакции на инцидент. Постоянно помните, что успешная борьба с нарушением - дело первостепенной важности.

5.6.3.6. Чьей помощью воспользоваться?

В мире существует довольно много групп реагирования на нарушения информационной безопасности (например, CERT, CIAC). Аналогичные группы имеются во многих важных правительственных агентствах и больших корпорациях. Если у Вашей организации есть контакты с подобной группой, с ней необходимо связаться в первую очередь и как можно раньше. Такие группы отвечают за координацию реакции на инциденты нескольких организаций или более крупных сообществ. Даже если кажется, что нарушение затрагивает только одну организацию, информация, доступная через группу реагирования, способна помочь успешной борьбе с нарушением.

При выработке политики, касающейся реакции на инциденты, может быть принято решение о создании собственной группы реагирования по типу существующих, отвечающей перед организацией за борьбу с нарушениями информационной безопасности. Если группа создана, ей необходимо наладить взаимодействие с аналогичными структурами - во время инцидента налаживать доверительные отношения гораздо труднее.

5.6.4. Ответные меры

Важная тема, которой мы пока не касались, - это реальные меры, предпринимаемые для борьбы с нарушением. Их можно подразделить на следующие основные категории: сдерживание, ликвидация, восстановление, "разбор полетов".

- Сдерживание. Цель сдерживания - ограничить атакуемую область. Например, важно как можно быстрее приостановить распространение червя в сети. Обязательной частью сдерживания является принятие решений (останавливать ли систему, отсоединять ли ее от сети, отслеживать ли ее работу и события в сети, устанавливать ли ловушки, отключать ли некоторые сервисы, такие как удаленная пересылка файлов в ОС UNIX и т.д.). Иногда подобные решения очевидны. Если риску подвергается секретная, конфиденциальная или частная информация, систему нужно остановить. В некоторых случаях стоит пойти на риск, связанный с нанесением системе определенного ущерба,

если поддержание ее работы способно помочь в идентификации злоумышленника. Сдерживание должно выполняться с использованием предварительно выработанных процедур. Ваша организация должна определить приемлемые границы рисков при борьбе с нарушениями и предложить соответствующие стратегические и тактические решения. Наконец, на стадии сдерживания должны извещаться заранее выбранные инстанции.

- Ликвидация. После обнаружения инцидента необходимо в первую очередь позаботиться о его сдерживании. Когда эта задача решена, можно приступать к ликвидации. В этом Вам может помочь программное обеспечение. Например, существуют программы, ликвидирующие вирусы в небольших системах. Если нарушитель создал какие-либо файлы, самое время их удалить. В случае вирусной инфекции важно вычистить все диски, содержащие зараженные файлы. Убедитесь в чистоте резервных копий. Многие системы, подвергавшиеся вирусным атакам, время от времени заражаются повторно только потому, что не производится систематическая чистка резервных носителей.
- Восстановление. Когда инцидент ликвидирован, наступает время восстановления, то есть приведения системы в нормальное состояние. В случае сетевых атак важно установить заплаты, ликвидирующие использованные системные слабости.
- "Разбор полетов". Одну из самых важных стадий реакции на инциденты, о которой, тем не менее, почти всегда забывают, можно назвать "разбором полетов". Данная стадия важна потому, что она позволяет всем причастным лицам извлечь уроки из инцидента (см. раздел 6.3), чтобы в будущем в аналогичных ситуациях действовать эффективнее. В процессе "разбора полетов" служба информационной безопасности объясняется перед руководством и систематизирует информацию, необходимую для юридических акций.

Самый важный элемент данной стадии - анализ случившегося. Что именно и когда произошло? Насколько хорошо сработал персонал? Какая срочная информация понадобилась в первую очередь и как ее быстрее всего можно было получить? Что в следующий раз нужно делать по-другому? Постинцидентный отчет ценен как руководство к действию в аналогичных случаях. Составление хронологии событий (с указанием точного времени) важно и с юридической точки зрения. Необходимо также в кратчайшие сроки получить денежную оценку ущерба, нанесенного инцидентом: утраченных программ и файлов, повреждений аппаратуры, потерь времени на восстановление измененных файлов, реконфигурацию атакованных систем и т.п. Эта оценка может послужить основанием для последующего официального расследования.

5.6.4.1. Единая точка контакта

Когда инцидент в разгаре, важно решить, кто координирует действия множества специалистов. Принципиальной ошибкой была бы организация нескольких точек контакта, которые не в состоянии наладить согласованное управление событиями, а лишь увеличивают общую неразбериху, вызывая своими указаниями напрасную или неэффективную затрату усилий.

Человек, находящийся в единой точке контакта, может быть, а может и не быть руководителем работ по борьбе с нарушением. В принципе речь идет о двух разных ролях, для которых нужно подобрать "исполнителей". Руководитель работ принимает решения (например, он интерпретирует политику безопасности применительно к происходящим

событиям). На него возлагается ответственность за реакцию на инцидент. Напротив, непосредственная функция точки контакта состоит в координации усилий всех сторон, вовлеченных в ликвидацию инцидента.

В точке контакта должен находиться специалист, техническая подготовка которого позволяет ему успешно координировать действия системных администраторов и пользователей. Нередко управленческая структура организации такова, что администратор множества ресурсов не имеет достаточной технической подготовки и не знает деталей функционирования компьютеров, но, тем не менее, отвечает за их использование. Другая важная функция точки контакта - поддержание связей с правоохранительными органами и другими внешними организациями, когда возникает нужда в согласованных действиях нескольких инстанций.

Наконец, если предусматриваются правовые действия, такие как расследование, сотрудник, обслуживающий точку контакта, может представлять организацию в суде. Если свидетелей несколько, их показания трудно координировать, а это ослабляет позиции обвинения и затрудняет наказание нарушителя. Сотрудник точки контакта может представить суду собранные улики, минимизируя тем самым число прочих свидетелей. Как показывает опыт, чем больше свидетелей рассказывает об одном и том же, тем меньше вероятность, что суд им поверит.

5.6.5. Регистрационная документация

Целесообразно документировать все детали, связанные с инцидентом. В результате Вы получите информацию, незаменимую для восстановления хода событий. Детальное документирование в конечном итоге ведет к экономии времени. Если, например, не зафиксировать телефонный звонок, Вы, скорее всего, забудете почти все, что Вам сообщили. В результате Вам придется звонить еще раз и повторно получать информацию. При этом будет потрачено и Ваше, и чужое время, что едва ли можно считать приемлемым. Фиксация деталей поможет и при проведении расследования. Далее, документирование инцидента позволяет оценить размер нанесенного ущерба (что необходимо и Вашему руководству, и правоохранительным органам) и организовать "разбор полетов", из которого Вы можете извлечь полезные уроки.

Как правило, на ранних стадиях инцидента невозможно определить, понадобится ли расследование, поэтому Вы должны вести документацию так, как будто собираете улики для судебного разбирательства. Необходимо зафиксировать по крайней мере следующее:

- Все системные события (приобщите к документации системный регистрационный журнал).
- Все Ваши действия (с указанием времени).
- Все телефонные переговоры (имя собеседника, дата, время и содержание разговора).

Самый простой способ сохранить документацию - записывать все в регистрационную книгу. Это избавит Вас от поиска среди разрозненных листов бумаги и предоставит в случае необходимости централизованный, упорядоченный по времени источник информации. Большая часть записанных сведений может понадобиться в случае судебного рассмотрения. Таким образом, если Вы начали подозревать, что инцидент приведет к расследованию, или когда расследование уже началось, Вы должны регулярно (например, ежедневно) относить в архив подписанные Вами копии страниц регистрационной книги вместе с другими

необходимыми носителями информации, чтобы сохранить их в надежном месте. Разумно потребовать квитанцию о сдаче документации на хранение, с подписью и датой. Если всего этого не сделать, суд может не принять Ваших показаний.

5.7. Выработка мер, предпринимаемых после нарушения

5.7.1. Обзор

После ликвидации нарушения режима информационной безопасности, необходимо предпринять ряд действий, а именно:

1. Произвести переучет системных активов, то есть тщательно проверить, как инцидент повлиял на состояние систем.
2. Уроки, извлеченные из инцидента, должны найти отражение в пересмотренной программе обеспечения безопасности, чтобы не допустить повторения аналогичного нарушения.
3. Произвести новый анализ риска с учетом информации, полученной вследствие инцидента.
4. Должно быть начато следствие против виновников инцидента, если это признано необходимым.
5. Перечисленные шаги направлены на обеспечение комитета по политике безопасности предприятия обратной связью, чтобы политика оперативно пересматривалась и подправлялась.

5.7.2. Устранение слабостей

Устранить все слабости, сделавшие возможным нарушение режима безопасности, весьма непросто. Ключевым моментом здесь является понимание механизма вторжения. В некоторых случаях разумно как можно быстрее отключить доступ ко всей системе или к некоторым из ее функциональных возможностей, а затем поэтапно возвращать ее в нормальное состояние. Учтите, что полное отключение доступа во время инцидента заметят все пользователи, в том числе и предполагаемые виновники; системные администраторы должны помнить об этом.

Естественно, ранняя огласка может помешать следствию. Однако продолжение инцидента порой чревато увеличением ущерба, усугублением ситуации или даже привлечением к административной или уголовной ответственности.

Если установлено, что вторжение стало возможным вследствие дефектов аппаратного или программного обеспечения, следует как можно быстрее уведомить производителя (или поставщика), а также группу реагирования CERT. Настоятельно рекомендуется включить в текст политики безопасности соответствующие телефонные (факсовые) номера, а также адреса электронной почты. Чтобы можно было оперативно уяснить суть проблемы, дефект нужно описать максимально детально (включая информацию об использовании дефекта нарушителем).

После вторжения к системе в целом и к каждому компоненту следует относиться с подозрением. В первую очередь это касается системных программ. Ключевым элементом восстановления скомпрометированной системы является предварительная подготовка. Сюда входит вычисление контрольных сумм для всех лент, полученных от поставщика

(желательно, чтобы алгоритм вычисления контрольных сумм был устойчив к попыткам взлома). Взяв полученные от поставщика ленты, нужно начать анализ всех системных файлов, доводя до сведения всех вовлеченных в ликвидацию инцидента лиц информацию обо всех найденных отклонениях. Порой бывает трудно решить, с какой резервной копии восстанавливаться; помните, что до момента обнаружения инцидент мог продолжаться месяцы или даже годы, и что под подозрением может быть работник предприятия или иное лицо, располагавшее детальным знанием системы или доступом к ней. Во всех случаях предварительная подготовка позволит определить, что можно восстановить. В худшем случае самым благоразумным решением будет переустановка системы с носителей, полученных от поставщика. Извлекайте уроки из инцидента и всегда корректируйте политику и процедуры безопасности, чтобы отразить изменения, необходимость которых выявил инцидент.

5.7.2.1. Оценивая ущерб

Прежде чем начинать восстановительные работы, необходимо уяснить истинные размеры ущерба. Возможно, на это уйдет много времени, но зато появится понимание природы инцидента и будет заложена база для проведения расследования. Лучше всего сравнивать текущее состояние с резервными копиями или с лентами, полученными от поставщика; еще раз напомним: предварительная подготовка - ключевой элемент восстановления. Если система поддерживает централизованное ведение регистрационного журнала (как правило, так и бывает), перемещайтесь по журналу назад и отмечайте аномалии. Если ведется учет запускаемых процессов и времени сеансов, попытайтесь определить типичные профили использования системы. В меньшей степени способна пролить свет на инцидент статистика доступа к дискам. Учетная информация может дать богатую пищу для анализа инцидента и официального расследования.

5.7.2.2. Восстановительные работы

После оценки ущерба следует разработать план восстановительных работ. Как правило, лучше всего восстанавливать сервисы в порядке поступления заявок от пользователей, чтобы минимизировать причиняемые неудобства. Помните, что наличие подходящих процедур восстановления крайне важно; сами эти процедуры специфичны для каждой организации.

Возможно, придется вернуться к начальному состоянию системы с последующей ее настройкой. Чтобы облегчить действия даже в таком, наихудшем, случае, храните записи о начальных установках системы и обо всех внесенных изменениях.

5.7.2.3. "Разбор полетов"

После того, как система вроде бы приведена в "безопасное" состояние, в ней, возможно, продолжают таиться дыры или даже ловушки. На фазе "разбора полетов" система должна быть тщательно обследована, чтобы выявить проблемы, упущенные при восстановлении. В качестве отправной точки разумно воспользоваться программными средствами обнаружения слабостей конфигурации (такими как COPS, SATAN). Следует, однако, помнить, что эти средства не заменяют постоянного системного мониторинга и хороших административных процедур.

5.7.2.4. Ведите журнал безопасности

Как отмечалось в разделе 5.6, журнал безопасности наиболее полезен на этапе устранения уязвимых мест. В этой связи упомянем два момента. Во-первых, следует документировать процедуры, использованные для восстановления режима безопасности. В это число могут войти командные процедуры, предназначенные для периодического запуска с целью проверки надежности системной защиты. Во-вторых, регистрируйте важные системные события. Это может помочь оценить ущерб от инцидента.

5.7.3. Усвоение уроков

5.7.3.1. Понимание урока

По завершении инцидента целесообразно составить отчет, в котором описывается инцидент, способы его обнаружения, процедуры исправления ситуации, процедуры мониторинга и усвоенные уроки. Все это способствует ясному пониманию проблемы. Трудно извлечь уроки из инцидента, если его причины не были поняты.

5.7.3.2. Ресурсы

Дополнительные устройства и методы обеспечения безопасности

Безопасность - это динамический, а не статический процесс. Организации зависят от характера доступных в каждый момент времени защитных средств, устройств и методов. Слежение за новинками в области информационной безопасности поможет поставить новейшие технологии на службу интересам предприятия.

Хранилище книг, списков, источников информации

Собирайте книги, списки, источники информации и т.п. как руководства и справочники по защите систем. Все время пополняйте свое собрание. Помните, что вместе с изменениями систем меняются методы и проблемы безопасности.

Сформируйте подгруппу

Сформируйте подгруппу из числа системных администраторов, которая станет ядром службы информационной безопасности. Наличие подобного коллективного органа позволит проводить обсуждение вопросов безопасности и сопоставление различных точек зрения. Эта подгруппа может также разработать политику безопасности предприятия и периодически совершенствовать комплекс защитных мер.

5.7.4. Совершенствование политики и процедур

5.7.4.1. Сформируйте механизмы для изменения политики, процедур и инструментов

Если нарушение режима безопасности стало возможным из-за плохой политики, то пока политика не скорректирована, организация обречена на повторные неприятности. После ликвидации инцидента следует подвергнуть политику и процедуры пересмотру, чтобы очертить круг изменений, необходимых для недопущения аналогичных случаев. Даже если нарушений нет, разумно периодически пересматривать политику и процедуры, поскольку меняется сама современная компьютерная среда.

5.7.4.2. Процедуры доклада об инцидентах

Необходимо отладить процедуру доклада об инцидентах, чтобы иметь их детальное описание вместе с принятыми мерами. Каждый инцидент должен разбираться подгруппой информационной безопасности предприятия с целью уяснения его сути и выработки предложений по совершенствованию политики и процедур безопасности.

6. Примеры средств обеспечения безопасности в ОС Solaris

В этой главе мы рассмотрим основные моменты, связанные с безопасностью, применительно к ОС Solaris. В настоящее время эта операционная система является одной из наиболее распространенных. Во многом это связано с тем, что растет популярность рабочих станций и серверов Sun Microsystems, во многом с тем, что Solaris – весьма удачная и стабильная реализация UNIX. Кроме всего прочего, Solaris все чаще применяется в банковской сфере, в сфере научных исследований, в области работы с базами данных и автоматизации документооборота. Словом, в тех областях, где потребность в обеспечении безопасности данных особенно велика. А последние шаги Sun в области увеличения популярности Solaris, а именно – объявление этой ОС бесплатной для некоммерческого использования, позволяют рассчитывать на резкое увеличение числа рабочих станций и серверов под управлением это системы.

Операционная система Solaris располагает всеми необходимыми возможностями для построения надежных информационных систем с удобным централизованным управлением. Стандартная маска прав, принятая в ОС UNIX, дополнена расширенными списками контроля доступа - ACL, имеется встроенный механизм централизованного ведения системных журналов, встроена поддержка системы сетевого управления и аутентификации NIS+, имеется возможность защиты от некорректно написанных программ и т.д. Все эти механизмы призваны обеспечить простоту управления и настройки при высокой защищенности информации в системах и сетях на базе Solaris. Именно этим возможностям и посвящена эта глава.

6.1. Обзор

Операционная система Solaris рассчитана на работу в сетях клиент-сервер. Что это означает? Во-первых, то, что различные машины в сети неравноправны. Часть из них имеет большую память, большие дисковые пространства, и предназначена для предоставления этих ресурсов другим машинам сети. Свои ресурсы эти машины предоставляют с помощью различных сервисов: почтового, файлового, печати и т.д. Имеется так же некоторое количество сервисов безопасности. Из них мы рассмотрим только систему NIS+. Все эти сервисы относятся к уровню сервисов на модели безопасности сети. Остальные машины сети имеют значительно более скромные объемы памяти и дисков и являются клиентами серверов. Так, например, возможна загрузка операционной системы не с локальных дисков рабочих станций, а с сервера ОС.

При любом способе загрузки, желательно монтирование домашних каталогов пользователей рабочих станций с сервера NFS. Сочетание такого монтирования с использованием NIS+ позволяет организовать единое подключение к сети и единые пароли для пользователя на любой машине сети.

В принятой ранее модели рабочие станции относятся к уровню локальной сети, так как сами не несут критичной информации. Тем не менее, их защита от некоторых видов атак так же необходима, так как в противном случае, они могут стать причиной нарушение режима безопасности всей сети.

Итак, какова общая структура сети, основанной на ОС Solaris? Знание ответа на этот вопрос позволит легко производить настройку операционной системы. Основной службой

сетевого управления является NIS+. Использование, например, DNS, возможно, но не рекомендуется. Это связано с особенностями аутентификации. Кроме того, за счет использования криптографии NIS+ существенно надежнее, нежели DNS. Более того, станции и серверы, использующие DNS, не поддерживаются службой SunService. По умолчанию, Solaris так же использует удаленное хранение системных журналов. Почтовая служба в идеологии Solaris так же требует отдельного почтового сервера для обработки почты проходящей из локальной сети во внешний мир и обратно. Словом можно сказать, что Solaris по умолчанию использует некоторые механизмы защиты информации, описанные ранее.

При установке сети на базе Solaris, в первую очередь, необходимо обратить внимание на установку серверов и настройку сетевых служб. От полноты планирования правильности реализации на этом этапе зависит легкость установки и защита всех рабочих станций сети.

В типовой установке Solaris потребуется набор из 4-х сервисов:

- почтовый сервер
- файловый сервер
- сервер NIS+
- сервер ведения системных журналов

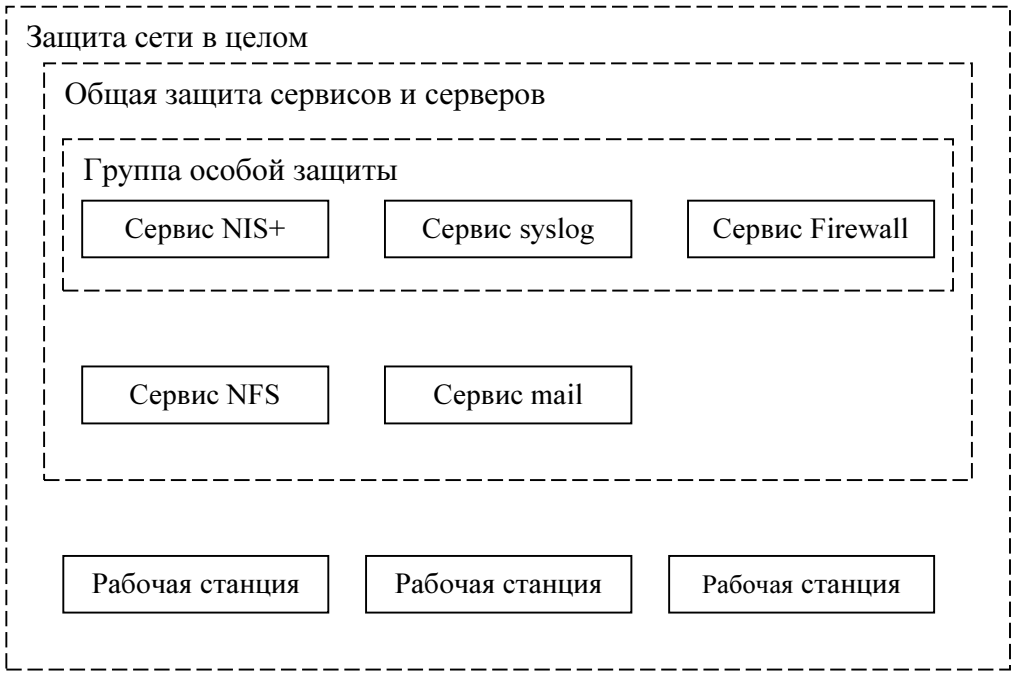


Рис 6.1. Иерархия защиты сервисов и серверов Solaris.

Для каждого из перечисленных сервисов желательно выделение отдельного сервера. Исключение могут составить только сервисы NIS+ и ведения системных журналов – их можно расположить на одном физическом сервере. Требования к этому серверу таковы – максимальная физическая защита и невозможность доступа к нему пользователей. Необходимо запретить на нем такие сервисы как, например, telnet или mail. Обязательно запрещение ftp, tftp, finger, nfs и т.д. Управление им может производиться только с защищенной консоли. Отметим, что этот сервер – ключ к безопасности всей сети в целом. Поэтому отношение к его защите должно быть соответствующее. Защита сервера, например, NFS, может быть несколько менее серьезной (например, невозможно запретить сервис nfs),

однако, сервисы, не имеющие отношения к функциональности, должны быть так же отключены.

Кроме этого, желательно выделение еще одного сервера для функционирования межсетевого экрана типа Solstice Firewall-1. Требования по защите этого сервера – такие же, как к серверу NIS+.

Теперь, рассмотрев в общем архитектуру построения сетей на базе Solaris, перейдем к более частным вопросам настройки отдельных сервисов в данной иерархии.

6.2. Права доступа и механизм ACL

6.2.1. Общие положения

Права доступа к файлам составляют основу безопасности OS Unix вообще, и Solaris в частности. Как правило, большинство проблем с безопасностью и несанкционированным доступом к данным связано с правами доступа. Поэтому этот вопрос надо осветить подробнее. Для рассмотрения прав доступа необходимо, прежде всего, понять и четко разобраться, что же такое файлы и каталоги в ОС Solaris.

Для начала пример (Курсивом выделены вводимые пользователем символы):

```
bash$ ls -l
total 783
-rwx------   1   eagle  users  1     Jan   25   18:28  19067haa
-rw-r--r--   1   dma    mail   1     Jan   16   12:38  filter.14428
-rw-------   1   yuri   root   3954  Jan   24   02:59  pop3a13598
-rw-------   1   yuri   root   3954  Jan   24   03:00  pop3a13600
```

Отметим некоторые особенности.

Во-первых, операционная система Unix использует прямой слеш «/»вместо обратного «\» как, например, dos.

Во-вторых, набранная нами команда – ls –l. В данном случае ключ –l означает, что мы получаем так называемый «длинный» список – с большим количеством информации о файлах. Если бы не использовали ключ –l, а набрали просто – ls, то получили бы такой результат:

```
bash$ ls
19067haa    filter.14428   pop3a13598 pop3a13600
```

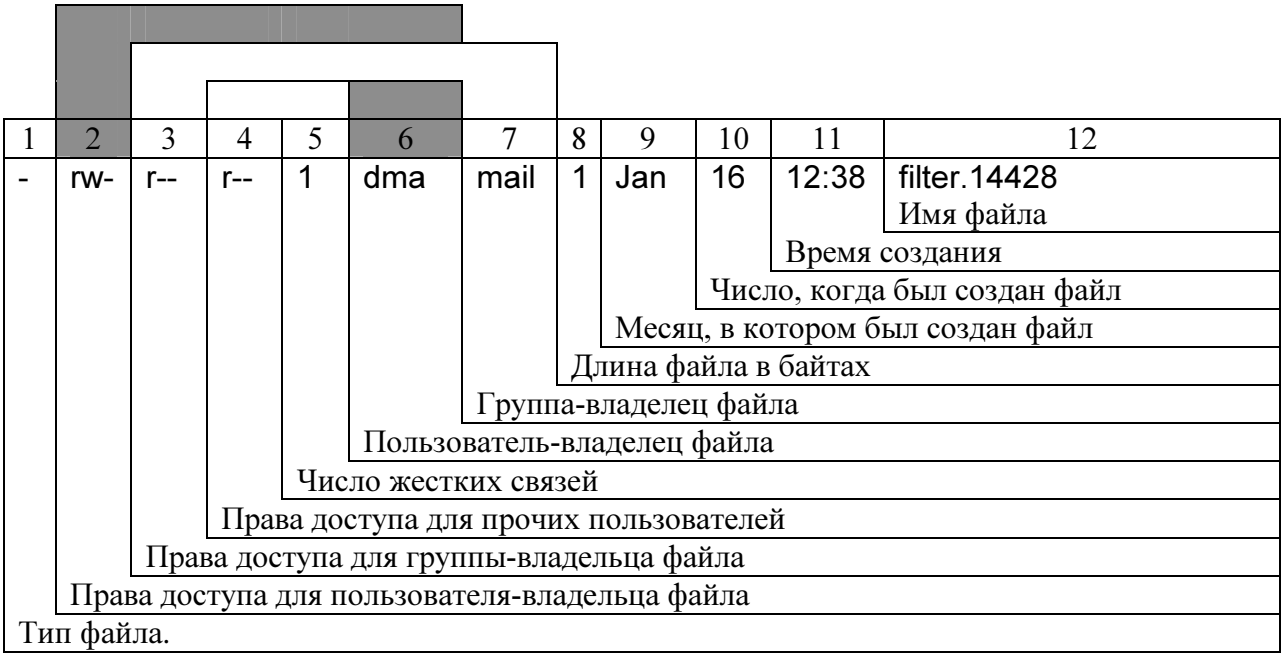
Как видите, такой формат списка файлов дает немного информации. Кроме опции –l, мы можем использовать опцию –al:

```
bash$ cd /tmp
bash$ ls -al
total 794
drwxrwxrwt4 root   root   8192  Jan   25   23:05  .
drwxr-xr-x22 root  root   1024  Dec   28   18:07  ..
-rwx------1  eagle  users  1     Jan   25   18:28  19067haa
-rw-r--r--1  dma    mail   1     Jan   16   12:38  filter.14428
-rw-------1  yuri   root   3954  Jan   24   02:59  pop3a13598
-rw-------1  yuri   root   3954  Jan   24   03:00  pop3a13600
```

Итак, список стал длиннее. Рассмотрим, что же означает полученная нами информация. Для примера возьмем одну строку:

```
-rw-r--r--1    dma   mail   1      Jan    16     12:38 filter.14428
```

Как можно отметить, она состоит из девяти полей, первое из которых описывает тип файла и права доступа к нему. Это поле можно, в свою очередь, разбить на 4 поля. Итого, получаем 12 полей – их значение приведено ниже:



6.2.2. Файлы ОС Solaris

Итак, какого типа бывают файлы ОС Solaris?

Существует 6 типов файлов, в скобках приведены значения первого поля, соответствующие типу [20]:

1. Обычный, или регулярный файл (-).
2. Каталог (d).
3. Файл устройства символьного(c) или блочного (b).
4. Именованный канал или FIFO (|).
5. Символическая связь (l).
6. Сокет (|).

Обычный файл — наиболее общий тип файлов, содержащий данные в некотором формате. Операционная система не располагает данными о формате файла, и поэтому все файлы для системы представлены как последовательность байт. Вся интерпретация содержимого файла производится прикладной программой, обрабатывающей файл.

Каталог — с помощью каталогов стоится файловая система Unix. Каталог – это тоже файл, но его содержимое интерпретируется ядром. Файл каталога содержит данные об имени файла и указатель на структуру файловой системы, называемую метаданными, и содержащую дополнительную информацию о файле, например, длину файла, права доступа к нему, число жестких связей и т.д.

Связь – еще один особый вид файла. На самом деле, связь – это файл, содержащий ссылку на какой-либо иной файл. Иногда файл связи называют полнее – символическая связь.

Файл устройства обеспечивает доступ к физическому устройству. В OS Unix различаются файлы устройств символьные и блочные. Доступ ко всем устройствам осуществляется путем открытия, чтения и записи в соответствующий файл устройства.

Символьные файлы устройств предназначены для небуферизованного обмена данными с устройством, тогда как блочные позволяют производить обмен в виде пакетов фиксированной длины – блоков.

Именованный канал или FIFO – это файл, служащий для осуществления взаимодействия между процессами. Впервые такие каналы появились в System V UNIX, однако, благодаря удобству, сейчас такой механизм имеется почти в любой разновидности OS Unix.

Сокет, как и именованный канал, предназначен для организации связи между процессами. Сокеты часто используются для доступа к сети TCP/IP. Механизм сокетов впервые появился в BSD Unix, где на их основе построен не только доступ у сети TCP/IP, но так же и реализован механизм межпроцессорного взаимодействия.

Для четкого уяснения как же на самом деле работает файловая система ОС Solaris, рассмотрим ее несколько более подробно.

Итак, для начала несколько утверждений.

1. Каталог не содержит никакой информации о файле, кроме имени файла и номера узла файловой системы (inode), содержащего метаданные файла, то есть все необходимые данные о файле, такие как: права доступа, длина, имя владельца и группы владельца время создания и некоторые другие. Это означает, что, по сути, каталог – это таблица, состоящая всего из двух полей – номер inode и имени файла. Это утверждение крайне важно!
2. Файл в метаданных не содержит никаких сведений о своем имени, то есть о том, под каким именем он представлен в системе.

По сути, эти два утверждения составляют основу управления файлами в ОС Solaris. Что они дают?

Во-первых, любой файл может иметь неограниченное число имен в файловой системе.

Во-вторых, и это важно с точки зрения безопасности, сколько бы имен файл не имел в системе, права доступа к нему и владельцы файла будут одни и те же во всех его представлениях.

В-третьих, файлу совершенно безразлично, сколько у него имен и какие именно это имена.

Пример.

В качестве примера рассмотрим файл, в следующей файловой системе на рис. 6.2.

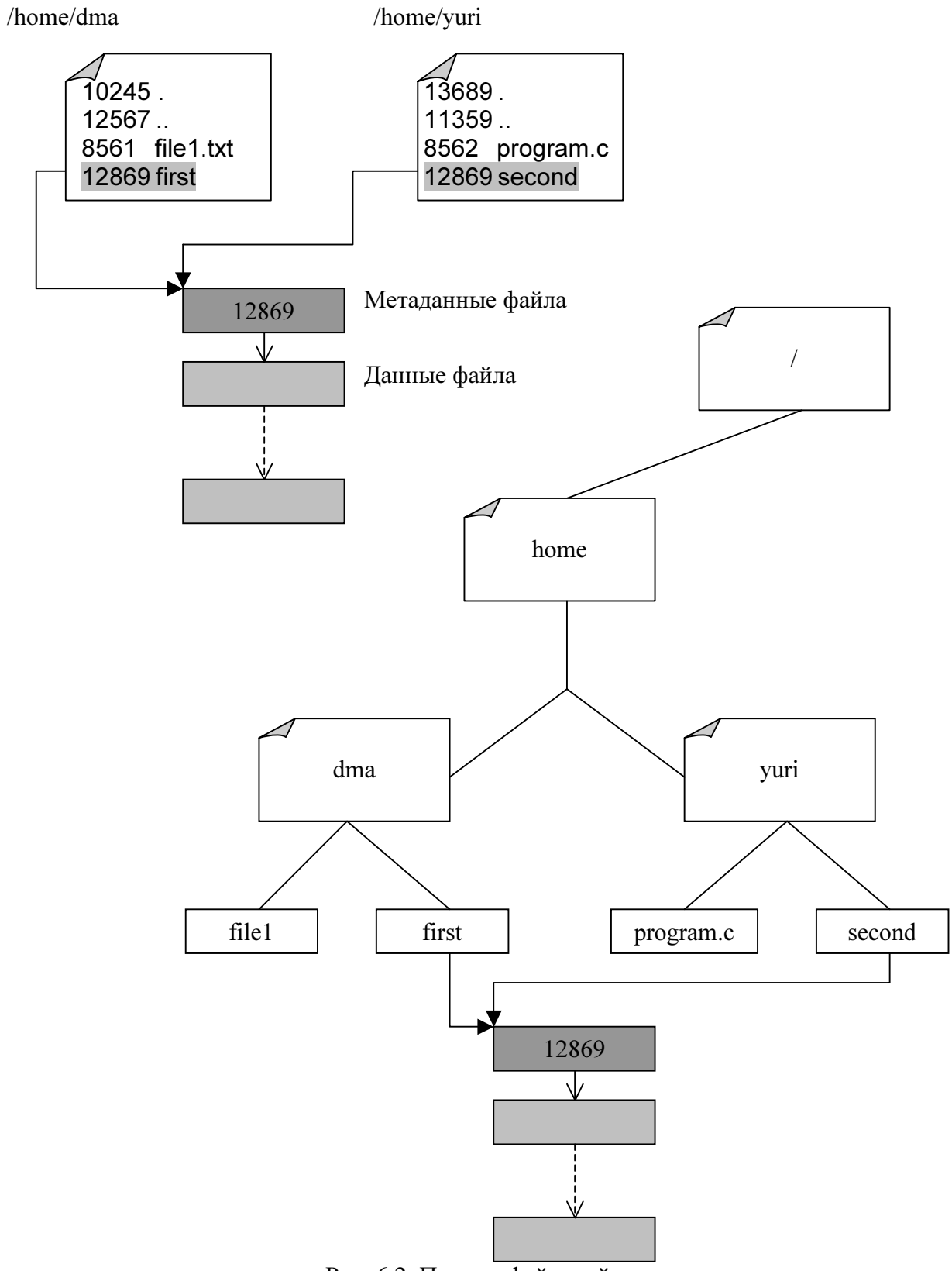


Рис. 6.2. Пример файловой системы

В рассмотренном примере, файлы /home/dma/first и /home/yuri/second на самом деле – один и тот же файл. Если, находясь в каталоге /home/dma мы выполним команду ls –la, то получим результат:

bash$ ***ls –la***

```
        total 5
drwxrwxrwx   4    root   root   8192  Jan   25    23:05  .
drwxr-xr-x   22   root   root   1024  Dec   28    18:07  ..
-rw-r--r--   1    dma    staff  6     Jan   25    23:05  file1.txt
-rw-r--r--   2    dma    staff  1024  Jan   25    23:05  first
bash$
```

Если мы выполним ту же команду, в каталоге /home/yuri, то получим:

```
bash$ ls –la
total 5
drwxrwxrwx   4    root   root   8192  Jan   25    23:05  .
drwxr-xr-x   22   root   root   1024  Dec   28    18:07  ..
-rw-r--r--   1    yuri   staff  6     Jan   25    23:05  program.c
-rw-r--r--   2    dma    staff  1024  Jan   25    23:05  first
bash$
```

Отметим особо, что и права доступа и владелец файла в обоих случаях оказались одни и те же.

Наличие у одного файла более одного имени называют так же *жесткой связью*. Что характерно, никогда невозможно сказать, что какое-то имя у файла является «главным». Удаляется же файл только тогда, когда исчезает последнее указание на него в дереве каталогов.

Кстати сказать, наличие у одного файла нескольких имен или жесткая связь, возможны только в пределах одной файловой системы. То есть чаще всего – в пределах одного жесткого диска, а точнее даже – в пределах одного раздела. Если части общей файловой системы расположены не на одном разделе, то жесткая связь между файлами в таком случае невозможна.

Таким образом, жесткие связи являются крайне неудобным способом управлять файлами. Не только в силу упомянутого ограничения, но и в силу того, что невозможно установить, откуда именно делается еще одна ссылка на файл. То есть, зная одно имя файла, невозможно быстро и точно указать, под какими еще именами файл хранится в файловой системе. Хотя метаданные содержат сведения об общем количестве имен данного файла в системе.

Вышеупомянутые ограничения привели к появлению еще одного типа связей в Solaris. А именно – *символические связи*, или *линки*. По сути, символическая связь – это файл, содержащий полный или относительный путь к файлу, на который эта связь указывает. Обозначается, как это было упомянуто ранее, такая связь символом l в поле типа файла. Права доступа к файлу связи всегда полные, то есть rwxrwxrwx. Реальные права определяются правами файла, на который эта связь ссылается.

6.2.3. Файловая система ОС Solaris

Под термином «файловая система» в данном случае понимаются правила именования и расположения основных системных каталогов. Это существенно облегчает как работу в операционной системе, так и ее администрирование. Эта структура используется в работе системы: например, при ее инициализации и конфигурировании, работе почтовой системы,

системы печати и т.д. Нарушение файловой системы может привести к неработоспособности отдельных компонент системы или, в худшем случае, всей ОС целиком. Примерное строение файловой системы Solaris приведено на рис. 6.3.

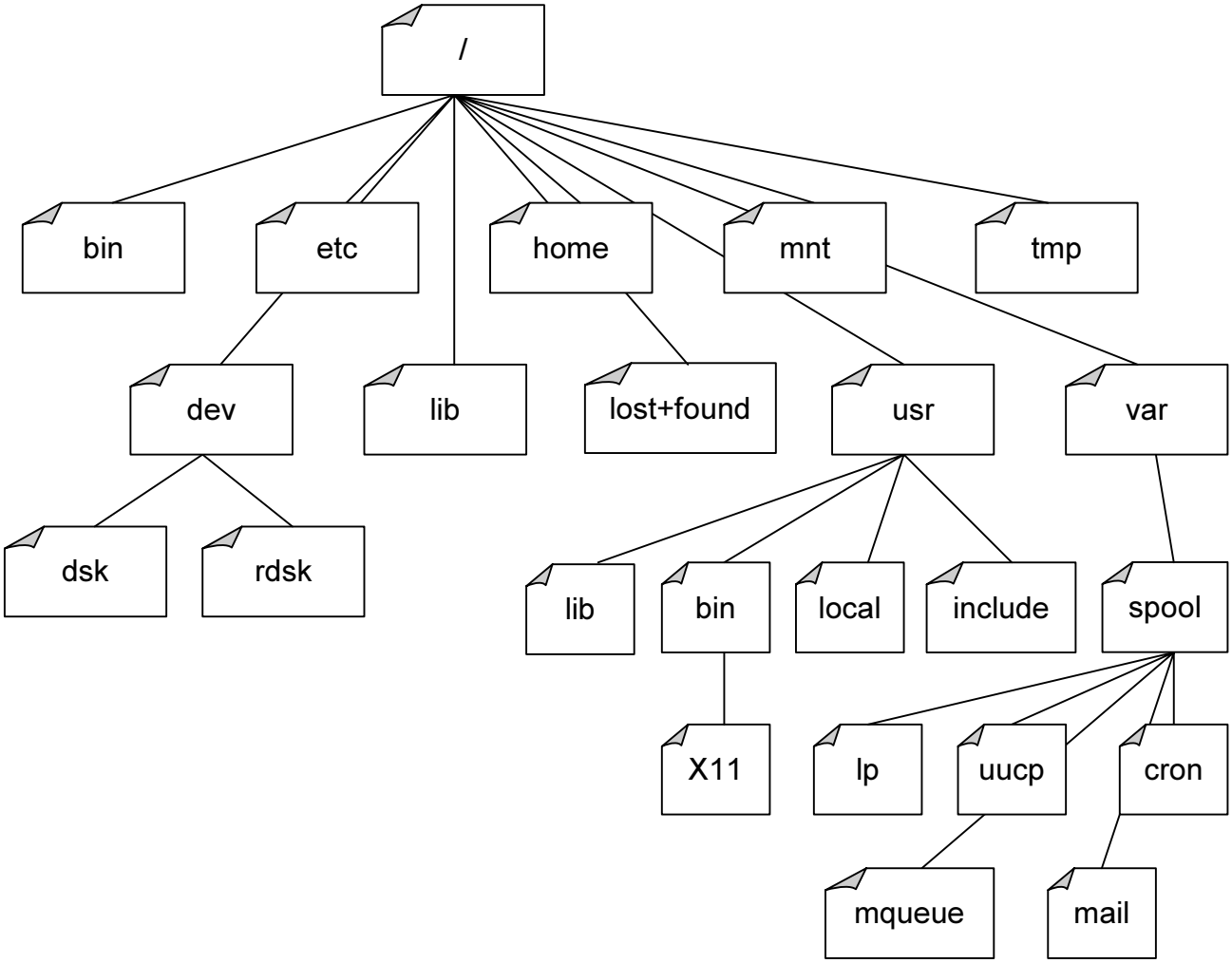


Рис. 6.3. Приблизительное строение файловой системы ОС Solaris

6.2.4. Краткое описание основных каталогов

Корневой каталог.

Корневой каталог / является основой файловой системы Solaris. Все остальные файлы и каталоги располагаются в рамках структуры, порожденной корневым каталогом, независимо от их физического расположения.

/bin

В каталоге /bin находятся файлы и утилиты общего пользования

/dev

Каталог /dev содержит специальные файлы устройств, являющиеся интерфейсом доступа к периферийным устройствам.

Каталог /dev может, в свою очередь, содержать несколько подкаталогов, группирующих устройства одного типа. Например, имеются каталоги /dev/dsk и /dev/rdsk,

представляющие диски системы как блочные устройства и как символьные устройства, соответственно.

/etc

В этом каталоге находятся системные конфигурационные файлы и многие утилиты администрирования. Наиболее важные для системы – скрипты инициализации системы. Эти скрипты находятся в каталоге /etc/rc0.d, /etc/rc1.d, /etc/rc2.d и т.д., соответственно уровням выполнения системы. Отметим, что в Solaris большинство утилит администрирования перенесено в каталог /sbin.

/lib

В каталоге /lib находятся библиотечные файлы языка С и других языков программирования. Часть библиотечных файлов может так же находиться в /usr/lib.

/lost+found

Каталог “потерянных” файлов. Ошибки целостности файловой системы Solaris, возникающие при неправильно останове системы или аппаратных сбоях, могут привести к появлению так называемых «безымянных» файлов – структура и содержимое которых являются правильными, но на них нет ссылок ни в одном из каталогов. Программа проверки восстановления файловой системы, например fsck, помещают такие файлы в каталог /lost+found под системными числовыми именами.

/mnt

Стандартный каталог для временного связывания, или “монтирования” физических файловых систем к корневой для получения единого дерева каталогов. Обычно содержимое каталога /mnt пусто. Но оно перекрывается связанной («подмонтированной») файловой системой.

/home

Общепринятый каталог для размещения пользовательских личных каталогов.

/usr

В этом каталоге находятся подкаталоги различных сервисных подсистем – системы печати, электронной почты и т.д., дополнительные библиотеки (/usr/lib), дополнительные программы и утилиты(/usr/bin, /usr/sbin), исходные тексты программ (/usr/src), справочная система man (/usr/man) и т.д.

/var

В Solaris этот каталог является хранилищем временных файлов (/var/spool, /var/tmp), файлов-логов (/var/adm, var/log) и т.д.

/tmp

Каталог хранения временных файлов, необходимых для работы различных подсистем Solaris. Обычно этот каталог открыт на запись для всех пользователей системы.

6.2.5. Владельцы файлов

В Solaris каждый файл имеет двух владельцев – пользователя-владельца и группу-владельца. Важно отметить, что пользователь-владелец файла не обязательно должен являться членом группы-владельца. Учитывая это обстоятельство, создав достаточно большое число групп можно добиться требуемой, например, стандартом С2, индивидуальной гранулярности прав, однако при практическом использовании это нереально, да, впрочем, и не нужно.

Используя механизм групп можно организовать совместное использование файлов для любого состава пользователей системы. Для этого необходимо создать группу, и установить для этой группы соответствующие права доступа к необходимым файлам. Чтобы предоставить некому пользователю доступ к этим файлам достаточно включить его в эту группу. Исключение же из группы автоматически лишает пользователя прав доступа к файлам.

Для определения владельцев файла можно воспользоваться так же командой ls –la. В приведенном выше примере-таблице поле 6 указывает пользователя-владельца файла, а поле 7 – группу-владельца.

Необходимо отметить одну особенность. На самом деле, как уже упоминалось выше, и права и владельцы файла хранятся в метаданных самого файла, и поэтому одинаковы для любого имени файла (в том случае, если файл имеет несколько имен в системе). Кроме этого, и владелец и группа-владелец хранятся не в текстовом виде, а в числовом. То есть реально хранятся номера (идентификатор пользователя(UID) и идентификатор группы-владельца(GID)). Программа ls, использованная в нашем примере просто сопоставляет номер владельца с его именем в системных таблицах и выдает уже текстовое значение. Если по каким-либо причинам номеру владельца или группы нет соответствующей записи в системных таблицах, программа ls выдаст просто числовое значение.

Владельцем файла при создании становится пользователь, создавший файл. В Solaris группа-владелец устанавливается по группе создателя файла. Поскольку один и тот же пользователь может быть членом нескольких групп, то владельцем-группой нового файла будет так называемая primary group, или первичная группа создателя. Однако имеется возможность устанавливать владельца-группу по тому же принципу, как в BSD. Эта возможность устанавливается для каждого каталога отдельно и достигается установкой дополнительных атрибутов на этот каталог.

Смена владельцев файла делается с помощью команд chown и chgrp. Например

Изменить владельца файла file1 на пользователя dma:

bash# ***chown dma file1***

Изменить группу-владельца на группу staff.

bash# ***chgrp staff file1***

Изменить сразу и владельца и группу-владельца:

bash# ***chown dma:staff file1***

Операции изменения владельца и группы-владельца может производить владелец файла. Строго говоря, реализацию Solaris в этом вопросе нельзя признать удачной,

поскольку пользователь имеет возможность маскировать свои файлы под других пользователей, обходя, таким образом, системные ограничения (например, по использованию дискового пространства).

6.2.5. Права доступа к файлу

В Solaris существует три базовых класса ограничения доступа к файлам и каталогам:

- для владельца файла
- для группы-владельца
- и для остальных пользователей

Solaris поддерживает так же 3 типа доступа к файлам:

- на чтение (r)
- на запись (w)
- на выполнение (x)

Таким образом, для трех классов существует по 3 типа доступа. В приведенной выше таблице доступ к файлу для всех трех классов описывается колонками 2,3 и 4. Наличие у данного класса определенного права обозначается соответствующим символом «r», «w» или «x», а отсутствие – символом «-».

Значение прав доступа различно для разных типов файлов. Для файлов операции, которые можно производить с этим файлом следуют из названия самих прав. Чтобы просмотреть содержимое файла, достаточно иметь право на чтение. Для того чтобы иметь возможность редактировать файл, необходимо кроме права на чтение, иметь еще и права на выполнение данного файла. Наконец, чтобы можно было выполнить файл-программу необходимо иметь право на выполнение. Если исполняемый файл является командным скриптом, тогда для его выполнения потребуется так же и право на чтение.

Все сказанное, за исключением права на выполнение, имеет смысл не только для обычных файлов, но так же и для файлов устройств, каналов и сокетов. Для каталогов вышеупомянутые права доступа имеют несколько иной вид, а для символических связей они вообще не используются, так как права доступа контролируются по метаданным самого файла, а не связи.

Как уже упоминалось выше, для каталогов права доступа имеют несколько иной вид, чем для обычных файлов. Это связано с тем, что Solaris трактует операции, проводимые с каталогами несколько отлично от иных файлов. Право чтения каталога дает право прочитать содержимое каталога, но не метаданных файлов. То есть получить список файлов в каталоге можно с помощью команды ls. Однако если мы хотим получить больше информации о файле (используя, например ls –la), то потребуется уже просмотреть метаданные файлов. Права на чтение каталога для этого уже недостаточно. Потребуется право на чтение метаданных файлов. Это право обеспечивает «право на выполнение» — ‘x’ для каталога. Неоднократно можно услышать, что право на выполнение для каталогов означает право на сканирование содержимого каталога. На самом деле, как мы видим, это не так. Понятие «сканирование» в Solaris вообще отсутствует. Наконец, отметим, что права на чтение и на выполнение действуют независимо.

Совершенно особого внимания требует право на запись в каталог. Прежде всего, отметим, что право на запись в каталог – это очень и очень большое право. Это право позволяет

создавать новые файлы и удалять уже имеющиеся в данном каталоге. При этом, чтобы удалить файл, вовсе не обязательно иметь право на запись в него! Отметим это особо. Если вы не имеете прав на запись в файл (а в принципе – даже никаких прав на данный файл, даже на чтение), но имеете право на запись в том каталоге, где этот файл находится, вы можете совершенно спокойно удалить его. Поэтому к предоставлению права на запись следует относиться осторожно. Конечно, существует возможность несколько обезопасить себя в том случае, если предоставление права на запись необходимо. Достигается это установкой особого атрибута на каталог, так называемого «Sticky Bit».

Итак, для выполнения операций над файлом имеют значение класс доступа, к которому вы принадлежите (владелец, член группы-владельца или прочие) и права доступа, определенные для вашего класса. При этом надо иметь ввиду следующее обстоятельство. Операционная система Solaris проверяет права по классам до первого совпадения. Что это означает?

При попытке пользователем произвести некоторые операции с файлом, проверки проводятся в следующем порядке:

1. Если операция запрашивается суперпользователем, то никаких проверок прав доступа не производится. Это позволяет суперпользователю иметь неограниченный доступ к файловой системе.
2. Если операция запрашивается владельцем файла, то проверяются права доступа для владельца – если операция позволена правами, она выполняется, если нет – в доступе отказывается. Никаких дополнительных проверок не производится.
3. Если операция запрашивается пользователем-членом группы-владельца, то проверяются права доступа для группы-владельца — если операция позволена правами, она выполняется, если нет – в доступе отказывается. Никаких дополнительных проверок не производится.
4. Если операция позволена правами доступа для прочих пользователей – она выполняется, в противном случае в доступе отказывается.

Как можно заметить, права в Solaris не являются аддитивными. То есть если пользователь является владельцем файла, то доступ определяется исключительно правами для владельца, права группы-владельца не проверяются, даже если пользователь является членом группы-владельца.

6.2.6. Дополнительные атрибуты файлов и их значение

Помимо рассмотренных ранее основных прав (или атрибутов) файлов и каталогов, существует еще несколько дополнительных атрибутов, изменяющих обычное выполнение операций с файлами. К ним относятся:

для обычных файлов:

sticky bit	сохранить образ задачи в памяти по завершению
SUID	установить UID процесса при выполнении
SGID	установить GID процесса при выполнении
блокирование	установить обязательное блокирование файла

для каталогов:

sticky bit	позволяет пользователю удалять только те файлы, владельцем которых он является

SGID позволяет присваивать создаваемым файлам GID каталога, аналогично BSD

Рассмотрим значение этих атрибутов для файлов.

Sticky bit. В настоящее время этот атрибут для файлов используется редко. Настоящее его название – “save text mode”. Установка этого атрибута означает, что при завершении выполнения программы ее образ, то есть данные и код, не будут удалены из памяти. Это позволяет заметно ускорить запуск часто используемых приложений, например, командного интерпретатора, на медленных машинах.

Атрибуты SUID и SGID устанавливаются только для исполняемых файлов. Они позволяют производить выполнение программ с иными правами, чем обладает пользователь. Программа с установленными атрибутами SUID и SGID выполняется с правами владельца и группы-владельца, соответственно. Это позволяет в редких случаях обычным пользователям производить операции, обычно им недоступные. Как правило, установка SUID и SGID имеет смысл для программ, владельцем которых является суперпользователь.

Следует особо отметить, что если программа в процессе выполнения запускает другие задачи, то они будут наследовать ее права. Поэтому установка SUID и SGID требует большой осторожности и доверия к программе, для которой устанавливаются эти атрибуты. Большинство проблем с безопасностью систем вызвано ненадежностью программ, имеющих атрибут SUID или SGID.

Наконец, *атрибут блокирования* позволяет заблокировать файл, если его использует какой-то пользователь. Благодаря этому существует возможность избежать конфликтов при одновременном открытии. Атрибут этот используется достаточно редко, но иной раз бывает просто необходим.

Как уже говорилось выше, для каталогов эти атрибуты имеют особое значение.

Sticky bit. При обсуждении прав доступа к каталогам говорилось, что предоставление права на запись в каталог дает пользователю очень большие права – возможно даже удаление файла, по отношению к содержимому которого этот пользователь не имеет никакого отношения. Установка атрибута sticky bit на каталог позволяет избежать подобного «беспредела». Из каталога с установленным sticky bit пользователь, даже имея права на запись, может удалить только файлы, владельцем которых он является или имеет права на запись. Примером каталога, для которого необходима установка атрибута «sticky bit» может служить каталог /tmp, используемый для хранения временных файлов. Все пользователи системы должны иметь право на запись в этот каталог, но удаление чужих временных файлов крайне нежелательно. Именно эту ситуацию и позволяет разрешить установка «Sticky bit».

Атрибут SGID для каталогов так же имеет иное значение, чем для файлов. В системах стандарта System V он позволяет имитировать поведение систем BSD, когда файл, создаваемый в каталоге, наследует группу-владельца по группе-владельцу каталога. В некоторых ситуациях такое поведение может оказаться полезным.

Наконец, кто же имеет право менять права доступа и атрибуты файлов? Установку всех прав доступа может изменять либо владелец файла, либо суперпользователь.

Установку Sticky bit имеет право делать только суперпользователь, независимо от того, устанавливается этот атрибут на каталог или на файл.

Установить атрибут SGID может либо суперпользователь, либо владелец файла, но только в том случае, если его главная группа (primary group) совпадает с группой-владельцем файла.

6.2.7. Управление правами с помощью ACL

По мере распространения Solaris в коммерческих вычислительных комплексах появляется необходимость в использовании более гибких схем обеспечения безопасности доступа к файловым системам. Для этого существуют "списки доступа" или ACL, которые были впервые введены в Solaris 2.5.1[21].

ACL позволяет владельцу файла управлять правами на файлы и каталоги в UFS (Unix File System – файловая система по умолчанию в Solaris) для отдельных пользователей и групп. Кроме того, владелец файла может определять набор прав по умолчанию в каталоге, так что все файлы, создаваемые в этом каталоге, получают одинаковый набор ACL. Поддержка для ACL существует сегодня в Solaris для следующих типов файловых систем: UFS (Unix File System), NFS (Network File System, Version 2 и Version 3), CacheFS (Cache File System) и LOFS (Loopback File System).

Другие типы файловых систем в Solaris ничего не знают об ACL, и, следовательно, не могут осуществлять защиту в соответствии с ACL файла. Кроме того, ACL могут применяться только к каталогам, нормальным файлам, FIFOs и символическим ссылкам (symbolic links).

Формат для ACL файла состоит из двух или трех колонок, разделенных двоеточием:

```
entry_type:[uid|gid]:perms
```

Первая колонка, entry_type, определяет ACL для пользователя, группы, других или маску (ACL mask). Вторая колонка - это (возможно) пользовательский ID (UID) или имя пользователя, или group ID (GID) или имя группы. Для типов "другие" или mask, эта колонка неприменима и потому не требуется. Третья колонка предназначена для файловых прав (file permissions). File permissions принимают форму либо обычных rwx (read/write/execute), либо числовую форму в восьмеричной системе (напр. 7 для rwx, 4 для r--, 6 для rw-, и т.д.), что полностью эквивалентно формату в chmod(1) [23]. Внутреннее представление в виде структуры данных выглядит так (из /usr/include/sys/acl.h):

```
typedef struct acl {
        int a_type;     /* entry type */
        uid_t a_id;     /* UID | GID  */
    o_mode_t a_perm;    /* permissions */
} aclent_t;
```

Каждое поле в указанной структуре соответствует частичному полю ACL, описанному выше.

Когда username (или UID) отсутствует, и поле groupname (или GID) пусто, то применяются традиционные файловые права Solaris (мы увидим это в примере ниже).

Пользователи устанавливают, модифицируют и удаляют ACL с помощью команды setfacl(1), и проверяют файловые ACL с помощью команды getfacl(1). Короткий пример ниже демонстрирует использование файловых ACL.

```
1  sunsys> ls -l file1
2  -rwxr-xr--   1 jim      staff        130 May 25 22:13 file1
3  sunsys> chmod 000 file1
4  sunsys> ls -l file1
5  ----------   1 jim      staff        130 May 25 22:13 file1
6  sunsys> setfacl -s user::rw-,group::r--,other:r-- file1
7  sunsys> ls -l file1
8  -rw-r--r--   1 jim      staff        130 May 25 22:13 file1
9  sunsys> getfacl file1
10 # file: file1
11 # owner: jim
12 # group: staff
13 user::rw-
14 group::r--            #effective:r--
15 mask:r--
16 other:r--
17 sunsys> setfacl -m user:moe:rw- file1
18 sunsys> ls -l file1
19 -rw-r--r--+  1 jim      staff        130 May 25 22:13 file1
20 sunsys> getfacl file1
21 # file: file1
22 # owner: jim
23 # group: staff
24 user::rw-
25 user:mode:rw-          #effective:r--
26 group::r--            #effective:r--
27 mask:r--
28 other:r--
```

Строка 6 демонстрирует команду setfacl(1) с опцией -s, которая означает установку ACL. Команда setfacl(1) на строке 6 не определяет специальных прав для конкретных пользователей или групп. Это в основном эквивалентно исполнению команды chmod 644 над тем же файлом.

С помощью команды getfacl(1) (строка 9) мы отображаем файловые права в формате ACL. Фактически мы в этот момент не имеем ACL для этого файла. Ядро Solaris отслеживает установку ACL, и для простых случаев, когда команда setfacl(1) просто устанавливает традиционные файловые права для user, group и other, реальный ACL для файла не создается. Ядро устанавливает права, определенные командой setfacl(1) в соответствующем поле в inode [22].

На строке 17 мы устанавливаем права read/write для пользователя Moe, используя m(modify) опцию с командой setfacl(1). Теперь при запуске команды getfacl(1) (строка 20), мы видим добавление для пользователя moe в ACL.

Есть несколько важных моментов по поводу поведения ACL и правил старшинства. Наш пример показывает ACL mask с r--. Маска ACL определяет максимальные права, выделенные всем, кроме владельца файла; в данном примере все, кроме владельца файла

имеют права только на чтение. Однако в ACL определены read и write права для Moe. Так что же случится, если Moe попытается записать в файл – что окажется главнее - его специальные права или маска ACL? Ответ заключается в том, что "победит" ACL mask. Вот еще один короткий пример.

```
1  sunsys> getfacl file1
2  # file: file1
3  # owner: jim
4  # group: staff
5  user::rw-
6  user:moe:rw-            #effective:r--
7  group::r--              #effective:r--
8  mask:r--
9  other:r--
10 sunsys> su moe
11 Password:********
12 $ id
13 uid=2001(moe) gid=22(stooges)
14 $ echo "write test" >> file1
15 file1: cannot create
16 $ exit
17 sunsys> setfacl -m m:rw- file1
18 sunsys> getfacl file1
19 # file: file1
20 # owner: jim
21 # group: staff
22 user::rw-
23 user:moe:rw-            #effective:rw-
24 group::r--              #effective:r--
25 mask:rw-
26 other:r--
27 sunsys> su moe
28 Password: ********
29 $ echo "write test" >> file1
30 $ exit
31 sunsys>
```

Мы не рассматривали, конечно, все варианты установки прав ACL. Нашей целью было лишь дать достаточно начальной информации, для начала работы с ACL. Кроме того, ACL пока еще редко применяются администраторами и пользователями, так как механизм это новый, а для большинства практических задач хватает обычных атрибутов. Отметим так же, что механизм ACL сложнее администрировать, нежели традиционные механизмы управления правами, так как ACL менее нагляден. Более подробная информация содержится в оналйн-документации на setfacl(1) и getfacl(1), а также в наборе документации Solaris. Для программных вызовов ACL существуют acl(2) и facl(2), а также библиотечные функции aclcheck(3) и aclsort(3).

6.3. Атаки с использованием переполнения буфера и защита от них

В последнее время для атаки на самые разные компьютерные системы все чаще и чаще используется механизм, получивший название buffer overflow exploit. Наиболее часто страдают от него системы, подключенные к сети Интернет, но ими дело не ограничивается [24]. Использование переполнения буфера может привести к компрометации даже сильно защищенных систем, с минимумом пользователей. Причем необходимо особо отметить, что использование переполнения буфера может сразу предоставить атакующему права суперпользователя. Характерным примером является прошедшая еще недавно в сети Интернет волна атак на сервис POP3, когда ошибка в реализации этого сервиса позволила многим злоумышленникам легко получать права суперпользователя на сотнях и тысячах серверов. Механизм переполнения буфера особенно опасен тем, что это не конкретная реализация, а целый класс атак, защититься от которых иной раз крайне сложно. Рассмотрим этот механизм подробнее.

Итак, предположим, что мы имеем некий сервер, предоставляющий в мир сервис, например, POP3. Программа, реализующая этот сервис, вызывается при обращении на порт 110 по протоколу TCP/IP. Как правило, вызов этой программы происходит из программы типа TCP Wrapper, в качестве которой используется inetd. То есть происходит следующее. Inetd работает постоянно и «прослушивает» внешние соединения на определенных портах (в нашем случае – порт 110). При обнаружении попытки соединения inetd запускает программу pop3d, обеспечивающую поддержку протокола POP3. После этого клиент, обратившийся к сервису POP3, далее общается уже с pop3d.

Для того чтобы inetd мог прослушивать порт 110, то есть порт с номером меньше 1024, ему необходимо иметь права суперпользователя. То есть в нашем примере inetd является процессом, выполняющимся с привилегиями суперпользователя. Запускаемый им процесс pop3d, как правило, тоже работает с привилегиями суперпользователя. Если каким-либо образом вынудить pop3d запустить командную оболочку (shell) – она унаследует его права и мы получим права суперпользователя в системе. Но как это сделать? Обычно никак. Программа pop3d не способна производить запуск каких-либо других программ. Вот тут на помощь и может прийти механизм переполнения буфера.

Предположим, что при написании программы программист посчитал, что, например, 1024 байта – вполне достаточно для временного буфера для передачи параметров при вызове какой-то внутренней процедуры. Хорошо, если это так. Но возможны ситуации, когда это допущение неверно. Причем, это может быть искусственно созданная ситуация. Рассмотрим пример кода сервиса pop3.

```
…….
int svr_auth(state,inbuf)
int state;
char *inbuf;
{
……..
char cli_user[256];
  if (strncmp(inbuf,"quit",4) == 0)
        return(svr_shutdown());
  if (strncmp(inbuf,"user",4) == 0) {
```

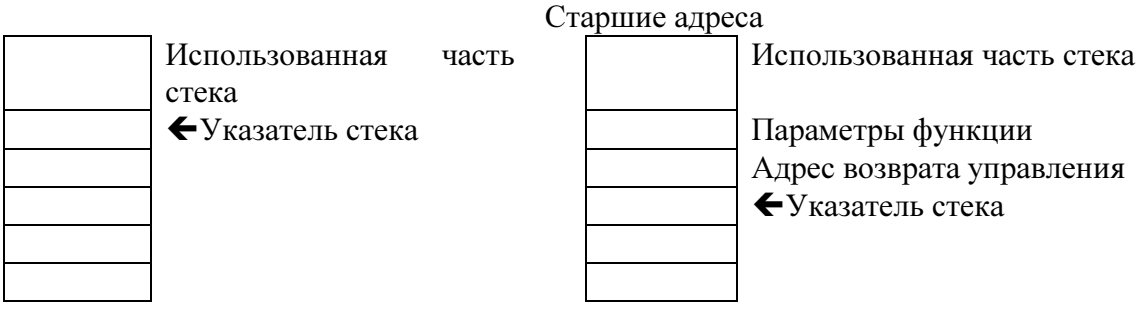
```
        ……..
strcpy(cli_user,inbuf);
                }
                strcpy(svr_buf,"+OK please send PASS command\r\n");
                state = SVR_PASS_STATE;
        ……..
return(state);
        }


        ……..
        main( int argc, char *argv[])
        {
          int svr_state = SVR_LISTEN_STATE;         /* State of POP3 server */
        ……
          fprintf(stdout,"+OK %s POP3 Server (Version %s) ready at %s\r\n",
                svr_hostname,VERSION,apop_timestamp());
          fflush(stdout);
          svr_state = SVR_AUTH_STATE;
          for ( ; ; ) {
                /* Wait for command from client */
                alarm(SVR_TIMEOUT_CLI);
        …….
                switch(svr_state) {
                case SVR_AUTH_STATE:    /* Expecting USER or APOP command */
…….
svr_state = svr_auth(svr_state,cli_buf);
                        break;
        ….……
                default:
                        fail(FAIL_CONFUSION);               /* Wont return */
                        break;
                }
        ……..


          }
             fld_release();                 /* [1.003] Make sure folder is released */
          exit(0);
        }
```

Подобные примеры, когда при вызове функции в качестве аргумента передается строка, а в самой процедуре используются буфер ограниченного размера и процедура strcpy(), можно встретить в большом количестве программ. Рассмотрим, что происходит при этом.

До вызова процедуры svr_auth() стек имеет вид рис. 6.4а. При вызове процедуры, ее параметры (или указатели на них) заносятся в стек. Туда же заносится и адрес возврата, чтобы после исполнения функции управление вернулось функции main. Таким образом, при передаче управления функции svr_auth стек будет иметь вид рис. 6.4 б.

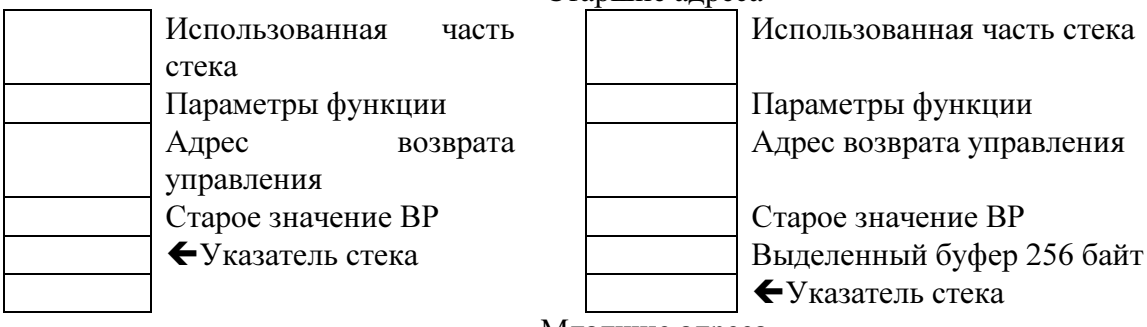


а) до занесения параметров б) после передачи управления

Рис 6.4. Схема передачи параметров через стек.

После передачи управления, для ссылки на параметры будет использоваться регистр BP. Поэтому программа сохранит его значение в стеке, а потом присвоит BP значение регистра SP. После этого произойдет резервирование места для буфера cli_user, размером 256 байт. Поскольку для выделения памяти не использовались функции malloc() или new() и не указывался модификатор static, то выделение произойдет так же в стеке (см. рис. 6.5а и 6.5б).

Далее выполняется код функции. Опасность скрывается в библиотечной функции strcpy(). Эта функция копирует фрагмент из второго буфера в первый. В нашем случае – из inbuf в cli_user. Все замечательно до тех пор, пока размер inbuf не превышает отведенных 256 байт. Функция strcpy() не проверяет длину отведенных буферов, а копирует пока не встретит символ конца строки 0. Поэтому, если inbuf превышает по длине 256 байт, то в стеке окажутся затертыми старое значение BP и адрес возврата управления.



а) до выделения буфера б) после выделения буфера

Рис 6.5. Схема выделения буфера в стеке.

Однако, это не вызовет ошибки до тех пор, пока не произойдет возврат из функции return(). При возврате управления функция прочитает из стека измененное значение адреса возврата и выполнение программы продолжится с этого момента. Если ошибка

переполнения буфера произошла случайно, то ничего страшного не произойдет – скорее всего, программа просто будет аварийно завершена ядром.

Однако ничто не мешает передать в качестве аргумента специально подготовленную строку, содержащую некий код, и затирающую адрес возврата таким образом, чтобы управление передавалось этому коду. Это не всегда просто, но очень часто возможно. В этом случае, управление будет передано некоторому коду, не контролируемому автором программы. Это и есть возможность взлома.

Конечно, необходимо соблюдение нескольких условий. Во-первых, необходимо примерно знать адрес указателя стека, чтобы корректно сформировать адрес возврата. Во-вторых, код не должен содержать 0 – это прервет выполнение strcpy(), и код будет скопирован не весь. В-третьих, необходимо точно соотнести длину кода и размеры буфера, чтобы не произошло аварийного завершения программы раньше времени. Тем не менее, задача не настолько сложна, чтобы быть неразрешимой.

Таким образом, указанный метод работает везде где:

- есть стек;
- имеется возможность вызвать систему через некоторый system call;
- адрес возврата из процедуры кладется в стек;
- локальные переменные размещаются в стеке;
- возможно выполнение кода, находящегося в стеке;
- существуют программы, выполняемые в режиме суперпользователя, доступные обычному пользователю и/или сервисы, содержащие ошибки.

Как известно, этим требованиям удовлетворяют практически все операционные системы - именно поэтому проблема является столь глобальной.

Теперь вернемся к сказанному ранее – сервис, например, pop3 выполняется с привилегиями суперпользователя. Это означает, что зловредный код автоматически получает права суперпользователя в системе. При этом злоумышленнику не обязательно ни иметь доступ к машине физически, ни иметь учетную запись на атакуемой машине. Что именно будет делать этот код – уже не принципиально. Важно то, что вся защита, так старательно возводимая, оказывается бессмысленной из-за ошибки в реализации отдельного сервиса.

Как бороться с этой проблемой? Во-первых, конечно, путем написания корректных программ. К сожалению, это не всегда в наших силах. Многие программы поставляются без исходных текстов. Но даже при наличии текстов отыскать опасное место не так просто.

Казалось бы, средствами операционной системы исправить эту проблему нельзя, однако все не так плохо. Обратим внимание на то, что для использования переполнения буфера для взлома системы, необходима возможность исполнения кода в стеке. Именно тут способна повлиять операционная система. В Solaris, начиная с версии 2.6, существует опция настройки ядра, при которой стек объявляется неисполняемым. Для этого в файле настроек ядра /etc/system необходимо указать опцию no_stack_executable. Отметим, что данная опция пока работает только на машинах архитектуры SPARC. Кроме того, возможны проблемы с функционированием некоторых программ, использующих в работе исполнение кода в стеке. К сожалению, других общих методов решения этой проблемы не существует.

6.4. Почтовый сервис sendmail

Sun рекомендует использовать архитектуру клиент-сервер для почтового сервиса [25]. Соответственно, имеется два варианта настройки sendmail – в качестве почтового клиента и сервера. Для этого имеется два файла конфигурации sendmail в каталоге /etc/mail: main.cf и subsidiary.cf. Файл main.cf предназначен для установки на машины, являющиеся почтовыми серверами, и обеспечивает набор правил, необходимый для маршрутизации почты через Интернет.

Файл subsidiary.cf предназначен для почтовых клиентов. Он обеспечивает прямую доставку почты в пределах одного домена, но всю почту для отправки во внешний мир пересылает на почтовый сервер - mailhost. Такое построение почтового сервиса полностью соответствует рекомендуемому выше построению почтовой службы. Отметим только, что именно subsidiary.cf используется программой установки Solaris по умолчанию. В версиях ранее 2.5 в Solaris входило еще и две программы sendmail – одна со встроенной возможностью распознавания записей MX сервиса DNS, а другая – нет. Но теперь sendmail один – с поддержкой MX-записей.

Sendmail в системе Solaris несколько отличается от прочих версий sendmail. Так, в конфигурации sendmail существует термин «mailer», который описывает способ, которым sendmail будет осуществлять доставку почты. Некоторые значения mailer являются стандартными: «local» - для доставки почты между пользователями одной машины, «prog» - для доставки почты локальным программам. Вместе с тем, в sendmail системы Solaris используются еще несколько нестандартных определений. Они описываются директивой DM в sendmail.cf:

ddn – сокращение от Data Defense Network – первоначальное название Интернет. Это определение используется в конфигурации sendmail, работающего напрямую с сетью Интернет, вместо того, чтобы пересылать внешнюю почту на почтовый сервер. В этом случае поддерживаются соединения по протоколу SMTP, а так же полные имена доменов Интернет. Это обеспечивает работу в сети Интернет. Как правило, определение ddn используется в файле main.cf, то есть – как раз на почтовых серверах.

ether – определение, близкое по смыслу ddn, но используемое только для доставки почты в пределах локальной сети организации, в пределах одного домена. В этом случае так же поддерживаются соединения по протоколу SMTP, но вся внешняя почта отсылается на почтовый сервер. Обычно определение ether используется в файле subsidiary.cf, но, конечно, оно используется и в main.cf, чтобы почтовый сервер мог осуществлять доставку почты во внутренней сети.

smartuucp – это определение необходимо только на тех почтовых серверах, которые соединены с внешними сетями по протоколу UUCP.

6.5. Сервис ведения системных журналов syslog

Сервис ведения системных журналов syslog является важнейшим для накопления информации о работе операционной системы, ее служб и сервисов, а так же – о попытках нарушения режима безопасности. Для этого сервиса Sun так же рекомендует использовать архитектуру клиент-сервер [26]. Под этим подразумевается хранение системных журналов на удаленной машине. В этом случае, при компрометации одной из машин сети,

злоумышленник не имеет возможности исправить журналы и тем самым скрыть свое присутствие в системе и факт нарушения режима безопасности.

Конфигурация сервиса syslog задается файлом /etc/syslog.conf. Мы не будем останавливаться на конфигурировании сервиса syslog вообще, коснемся лишь вопросов, связанных с настройкой удаленного хранения журналов.

Чтобы обеспечить удаленное ведение системных журналов, в файле syslog.conf необходимо в поле «действие» (actions), указать адрес сервера ведения системных журналов:

```
auth.notice      @loghost
```

В данном случае loghost – это принятый в системе по умолчанию псевдоним (alias) для сервера системных журналов. При этом необходимо убедиться, что этот псевдоним корректно описан в настройке сервиса имен – будь то файл /etc/hosts, сервис DNS или NIS+. Простейшей проверкой является команда ping –s loghost.

Такие модификации необходимо произвести во всех строках файла syslog.conf, в которых описываются классы событий, которые должны храниться удаленно на сервере журналов. Заметим, что на самом сервере не требуется никаких модификаций файла syslog.conf, за исключением, конечно же, случая, когда необходимо изменить файлы, соответствующие классам событий. После внесения всех изменений необходимо перезапустить программу syslogd, чтобы эти изменения вступили в силу.

Необходимо убедиться, что все классы событий и уровни (степени событий) описаны правильно в файле syslog.conf. Например, если в syslog.conf описано:

```
auth.notice      /var/log/authlog
```

не следует искать сообщений об аутентификации в файле /var/adm/messages. Отметим особо, что надо четко представлять, что такое уровень события. Например, уровень “notice” в данном случае – это, например, успешное выполнение команды su. Неудачное выполнение su имеет уровень “emergency”, и будет записано в файл /var/adm/messages, если в файле syslog.conf есть строка:

```
*.emerg;user.none     *
```

Сервис syslog не отправляет почтовые сообщения пользователям! Вместо этого он может выдавать сообщения на терминалы пользователей, подобно команде wall. Если вам необходимо получать сообщения о происходящем в системе по почте, необходимо создать канал, через который будут передаваться эти сообщения, и программу, отсылающую соответствующую почту. Или, например. создать скрипт, который будет регулярно выполняться, анализировать содержимое журналов и отправлять результаты анализа по почте.

Еще несколько общих замечаний по поводу сервиса syslog. Необходимо иметь ввиду, что для доставки сообщений на сервер системных журналов используется протокол UDP, не гарантирующий доставку пакетов. Из-за этого может происходить потеря информации системных журналов. Допустить этого нельзя. Поэтому лучшим решением является хранить журналы в двух местах одновременно – локально и на сервере. Это повысит сохранность и позволит легче обнаруживать целенаправленные модификации журналов злоумышленником.

Однако, в файле syslog.conf нельзя использовать несколько мест хранения журналов в одной строке:

```
auth.notice     /var/adm/authlog,@loghost
```

Вместо этого необходимо указать две строки:

```
auth.notice     /var/adm/authlog
auth.notice     @loghost
```

Однако, можно указывать несколько классов событий, разделяя их точкой с запятой.

6.6. Служба сетевого управления NIS+

Сетевой информационный сервис (NIS) служит для централизованного управления информационными ресурсами в распределенной среде [27]. Строящий, подобно DNS-сервису, древовидную структуру имен объектов в сети, NIS+ функционально существенно богаче. Сервис NIS+ создавался для замены NIS, устаревшего и небезопасного сервиса. Отметим, что NIS+ в операционной системе Solaris может обслуживать NIS-клиентов в режиме, называемом «YP-совместимый». Однако это режим не является основным. Сервис NIS+ призван решить проблемы, решение которых NIS не под силу.

Очень важно отметить, что NIS+ и NIS – это совершенно разные сервисы. Между ними нет ничего общего. Команды, общая структура NIS и NIS+ существенно различаются.

Сервис NIS+ работает на основе набора таблиц, содержащих сведения о машинах, параметрах удаленной загрузки, паролях, ключах аутентификации машин локальной сети, пользователях и группах, подсетях, сетевых масках, адресах Ethernet, сервисах, протоколах, параметрах автоматического монтирования удаленных файловых систем и удаленного вызова процедур (RPC). Поскольку указанная информация является весьма важной с точки зрения безопасности информационной системы, ее надежная защита с точки зрения всех трех критериев (конфиденциальность, целостность, доступность) абсолютно необходима. Такая защита реализуется при задействовании механизмов авторизации и аутентификации NIS+.

Сервис NIS+ использует дополнительные методы аутентификации. Пользователи, как и прежде, имеют свой пароль для доступа к компьютерам. Вместе с тем, они имеют дополнительный сетевой пароль для доступа к сетевым ресурсам. Этот пароль необходим для доступа к базам данных NIS+, и именно он обеспечивает новый уровень безопасности в сети. Обычно пароль для доступа к машине и сетевой пароль одинаковы. В этом случае пользователь сразу получает доступ к сетевым ресурсам. Но они могут и отличаться. Тогда необходимо использовать команду keylogin для ввода сетевого пароля.

Объекты NIS+ - это те составляющие, из которых строится вся база данных NIS+. Структура базы данных NIS+ составляется из 5 типов объектов:

Объекты-директории. Подобно директориям в UNIX объекты-директории NIS+ могут содержать один или несколько других объектов. Объекты-директории образуют древовидную структуру, в которой корневой домен находится на верхушке общей структуры, а поддомены образуют ветви. Объекты-директории используются для разделения общего пространства имен на отдельные части.

Объекты-таблицы. Эти объекты подобны картам NIS. Собственно, в них и содержится различная информация. Таблицы могут быть пусты, а могут содержать один или несколько **объектов-элементов**. Всего в NIS+ по умолчанию определены 17 типов таблиц. Управление таблицами осуществляется командами nistbladm или nisaddent. Объекты-элементы представляют собой строку в таблице. Каждая строка хранит одну запись.

Объекты-группы. Группы NIS+ подобны обычным группам UNIX. Они используются для управления правами доступа к базе данных NIS+. Для управления группами применяется команда nisgrpadm.

Объекты-связи. Связи – это просто указатели на другие объекты. Чаще всего используются указатели на таблицы или элементы. Связи NIS+ по своему смыслу аналогичны символьным связям в файловой системе UNIX. Для управления ими используется команда nisln.

Права доступа в NIS+ определяют, какие операции доступны тому или иному NIS-клиенту. Операции в NIS+ подразделяются на четыре класса: чтение (Read), модификация (Modify), создание (Create), удаление (Destroy). Каждое взаимодействие между клиентом и сервером может сводиться к запросу на проведение действий какого-либо из этих классов, и соответственно авторизовываться.

В NIS+ различаются четыре категории сетевых субъектов: владелец (Owner), группа владельца (Group), все (World), никто (Nobody), причем эти категории не совпадают с упоминавшимися выше категориями пользователей ОС Solaris и администрируются независимо.

В NIS+ имеется три уровня режима безопасности. Уровень 0 используется для тестирования и установки начального набора имен. В этом режиме любому пользователю NIS+ разрешен доступ к любому из объектов. Уровень 1 так же применяется на стадии тестирования и отладки. В этом режиме используется тип аутентификации LOCAL, когда параметром служит идентификатор пользователя. В случае, если он не будет найден, клиенту предоставляются права Nobody. При этом все механизмы авторизации работают в полной мере. Наконец, на уровне 2 производится аутентификация клиентов NIS+ с использованием DES-ключей, которая и будет рассмотрена далее.

NIS+ разделяет клиентов, в зависимости от требуемого сервиса, на две категории: клиент-пользователь и клиент-машина. Клиент выступает как клиент-машина, если запрашиваемый им сервис предполагает получение полномочий привилегированного пользователя (root). Каждому клиенту соответствует удостоверение (credentials), при помощи которого и производится аутентификация. Удостоверение состоит из сетевого имени (unix.UserID@domainname для клиента-пользователя либо unix.HostName@domainname для клиента-машины) и поля верификации, с помощью которого проверяется подлинность и аутентичность удостоверения. Все удостоверения NIS+ хранятся в одной из таблиц базы данных сетевого информационного сервиса, называемой "Cred table". Для каждого клиента в таблице содержится его имя, тип аутентификации, сетевое имя, открытый (public) и секретный (private) ключи. Удостоверение для каждого из клиентов создается администратором сети (домена сети). Сервер, обслуживающий NIS+, называется master-сервером. Удостоверения клиентов создаются при помощи программы nisaddcred. Эта программа формирует из идентификатора пользователя и имени домена сетевое имя, после чего заносит его в соответствующую таблицу. Для генерации ключей nisaddcred использует

сетевой пароль клиента, который может и не совпадать с пользовательским паролем, записанным в файле /etc/shadow. На основании пароля генерируется пара 192-битных ключей в соответствии с алгоритмом Диффи-Хелмана. Ключи также заносятся в NIS+ таблицу. При посылке запроса на NIS-сервер по умолчанию предполагается, что используется уровень безопасности 2. Если это оказывается не так, последовательно посылаются удостоверения уровней 1 и 0. Перед генерацией удостоверений уровня 2 (DES-аутентификация) клиент обязан воспользоваться командой keylogin, предоставляющей доступ к его секретному ключу. Команда keylogin берет секретный ключ в таблице NIS+, расшифровывает его с помощью сетевого пароля и помещает в локальную базу данных. Клиент запоминает также открытый ключ сервера, необходимый для посылки последующих запросов.

Рассмотрим процесс аутентификации более подробно. В качестве первого шага клиент по алгоритму Диффи-Хелмана генерирует DES-ключ и порождает временной штамп. Этот штамп кодируется DES-ключом и присоединяется к прочей информации, содержащейся в удостоверении клиента, образуя поле верификации. При проверке подлинности клиента сервер по сетевому имени отыскивает в таблице открытый ключ клиента. Затем, используя собственный секретный ключ и открытый ключ клиента, по алгоритму Диффи-Хелмана генерирует DES-ключ, которым расшифровывает временной штамп, переданный в удостоверении. Если разница между временным штампом клиента и текущим временем находится в заданном интервале, аутентификация считается успешной. Для подтверждения своей подлинности, сервер возвращает клиенту временной штамп, увеличенный на единицу и зашифрованный аналогичным образом.

Рассмотрим теперь процедуру авторизации в NIS+. Как уже упоминалось, существуют четыре категории, в соответствии с принадлежностью к которым в NIS+ назначаются права доступа: владелец, группа владельца, все, никто. К первой категории может быть отнесен клиент, создавший данный объект или получивший его в собственность при помощи процедуры nischown. Как правило, для этой категории возможен любой из видов доступа - либо непосредственно, либо путем изменения соответствующих атрибутов, на что владелец также имеет право.

Один или несколько NIS+ клиентов могут составлять NIS+ группу, на которую распространяются права доступа, присущие категории Group, по аналогии с правами доступа в файловой системе.

Любой клиент, прошедший аутентификацию NIS+, получает права категории World. Если же аутентификация не была проведена (не поступило удостоверения клиента), клиент может получить права категории Nobody. Если же удостоверение поступило, но было неправильным, права Nobody на этого клиента не распространяются: ему автоматически запрещается получение сервиса. Таблицы NIS+ имеют дополнительный уровень защиты, не предоставляемый для других типов NIS+ объектов. Этот уровень построен на раздельном доступе к строкам и столбцам. Таким образом, для таблицы, в зависимости от требуемого клиентом сервиса, могут проверяться права на доступ к самой таблице, к данной строке и данному столбцу.

6.7. Удаленный вызов процедур и сетевая файловая система (RPC и NFS)

Аналогичная идеология аутентификации применяется при реализации другого важного сервиса – сетевой файловой системы (NFS). NFS - исключительно мощный и удобный сервис, однако его использование сопряжено с угрозами безопасности информационной системы. По умолчанию NFS-сервер проводит аутентификацию клиента-машины, а не клиента-пользователя, что дает возможность несанкционированно получить привилегии суперпользователя (root).

Для проведения аутентификации рекомендуется использовать гораздо более безопасный режим защищенных удаленных вызовов процедур, на основе которых строится защищенная сетевая файловая система (Secure NFS).

Как и в случае NIS+, каждый клиент-пользователь и клиент-машина имеют открытый и секретный ключи. При помощи программы keylogin проводится дополнительная аутентификация пользователя, расшифровывается его секретный ключ и передается программе keyserver, участвующей в процедуре RPC. Программа keyserver с секретным ключом пользователя ожидает запроса на RPC-сервис. С началом транзакции keyserver генерирует случайный сеансовый ключ. Этим ключом шифруется временной штамп машины-клиента. Затем находится открытый ключ сервера и вырабатывается DES-ключ, с помощью которого шифруется ключ сеанса. Формируется удостоверение клиента, включающее в себя сетевое имя, зашифрованный сеансовый ключ, допустимая разница времени между клиентом и сервером, и шифрованный временной штамп.

Когда сервер принимает запрос клиента, программа keyserver, установленная на машине-сервере, ищет открытый ключ клиента в своей базе данных и вырабатывает DES-ключ, используемый для расшифровки ключа сеанса. С помощью последнего расшифровывается временной штамп, полученный от клиента, и сравнивается с текущим временем сервера. Аутентификация считается удачной, если разность времени находится в пределах полученного "окна". Удостоверение клиента запоминается сервером, чтобы противостоять попыткам нелегальной аутентификации путем воспроизведения перехваченной информации.

В качестве свидетельства того, что аутентификация успешно проведена и притом "правильным" сервером, клиенту возвращаются его идентификационный номер в таблице сервера и временной штамп, уменьшенный на единицу и шифрованный сеансовым ключом. Последующие запросы клиент формирует, посылая серверу свой идентификационный номер и новые зашифрованные временные штампы. Таким образом, сеансовый ключ передается по сети один раз в шифрованном виде, причем расшифровать его можно, только зная пару соответствующих ключей клиента и сервера.

При использовании механизмов аутентификации, встроенных в Solaris 2.x, необходимо учитывать следующее:

- Если произошло нарушение безопасности сервера при работе в режиме защищенного RPC, все ключи сервера и клиентов, содержащиеся в базе keyserver, должны быть изменены.
- Должна быть обеспечена физическая защищенность сервера.
- В момент загрузки бездисковых станций режим защищенного RPC не работает, следовательно, возникает угроза безопасности.

Нельзя применять программу keylogin при удаленном доступе на ненадежные машины, поскольку в этом случае туда посылается секретный ключ пользователя.

6.8. Внешний экранирующий сервис Solstice Firewall-1

Экранирующий сервис (брандмауэр) Solstice Firewall-1 не является составной частью операционной системы Solaris. Тем не менее, его роль для обеспечения безопасности сети от вторжения извне с использованием Интернет очень велика. Этот продукт неоднократно получал высокие статусы в различных рейтингах подобных систем. Кроме того, именно его рекомендует применять Sun, так что рассмотрение вопроса обеспечения безопасности сетей на базе Solaris было бы неполным без рассмотрения Firewall-1. Отметим, что Solstice Firewall-1 – продукт очень сложный [28, 29]. Детальное его рассмотрение могло бы занять целую книгу. Поэтому мы ограничимся только общим обзором, и подчеркнем несколько деталей, которые отличают Firewall-1 от других продуктов аналогичного назначения.

6.8.1. Обзор

Когда локальная сеть организации подключается к Интернет, особенно важной становится проблема защиты этой сети от вторжения извне через Интернет. Наиболее эффективным путем осуществления такой защиты является установка брандмауэра между локальной сетью и Интернет, как это было описано в общих чертах выше. Брандмауэр является гарантом того, что все соединения, производимые между сетью организации и Интернет, соответствуют Политике безопасности.

Для того чтобы обеспечить действительно эффективную защиту, брандмауэр должен осуществлять слежение и управлять всеми соединениями, проходящими через него. Для того чтобы обеспечить возможность управления для базовых сервисов TCP/IP (например, пропустить пакет, отбросить, зашифровать или отметить попытку соединения в журнале), брандмауэр должен получать, хранить и обрабатывать информацию, получаемую со всех уровней протокола и от других приложений.

Нельзя считать достаточным проверку каждого пакета по отдельности на соответствие каким-либо условиям. При принятии решения о допущении или недопущении того или иного соединения необходимо принимать во внимание два важнейших фактора: информацию о состоянии соединений и информацию о состоянии тех или иных приложений

В общем, решение о том, какие действия предпринять с пакетом, должны приниматься на основании четырех факторов:

I. информация о пакете, получаемая со всех уровней протокола.
II. информация о состоянии соединений. Например, информация о переданной команде PORT протокола ftp должна быть сохранена и учтена, чтобы входящий поток данных от сервера был допущен брандмауэром к клиенту.
III. информация о состоянии приложений. Это информация, сообщаемая брандмауэру прочими приложениями. Например, пользователь, прошедший один раз аутентификацию соответствующим сервисом, допускается к ресурсам сети. Если этого не учитывать, то аутентифицировать необходимо было бы каждый пакет.
IV. обработка информации. Это гибкий набор правил, согласно которым и перечисленным ранее условиям, брандмауэр принимает решение о разрешении соединения.

Чтобы обеспечить удовлетворение этих условий, традиционно используются два типа систем – пакетные фильтры и сервисы-посредники или прокси. Такой подход был описан и в модели безопасности, рассмотренной выше. Однако использование только этих систем не позволяет полностью удовлетворить всем четырем условиям.

Пакетные фильтры работают только на сетевом уровне, и наиболее серьезное ограничение при их использовании заключается в том, что они не способны обеспечить безопасность для большинства широко используемых сервисов. Пакетные фильтры сами по себе не способны обеспечить безопасность сети, потому что они не удовлетворяют следующим требованиям:

1. не способны поддерживать и учитывать информацию о соединениях, поскольку обрабатывают только часть заголовка пакета;
2. не способны учитывать состояние приложений, так как рассматривают пакеты по отдельности;
3. ограниченны возможности обработки информации. Это ограничение непосредственно следует из вышеназванных.

Кроме всего этого, пакетные фильтры, как правило, довольно сложны в настройке и управлении и не способны обеспечить требуемый уровень журналирования информации о соединениях.

Сервисы-посредники так же сами по себе не способны обеспечить безопасность. Такие сервисы работают на уровне приложений. Существенным преимуществом перед пакетными фильтрами является возможность учитывать состояние соединений и приложений. Сервисы-посредники так же обладают существенно более развитыми возможностями по обработке информации, то есть позволяют создать гибкий набор правил.

Наиболее существенными недостатками сервисов посредников являются:

I. ограниченность обслуживания. Каждый сетевой сервис требует специального сервиса-посредника. Существует ограниченное количество сервисов-посредников, из-за чего какой-либо сетевой протокол может оказаться недоступным. Кроме того, ограничена масштабируемость этих сервисов.
II. Технологические ограничения. Невозможность создать сервисы-посредники для таких протоколов, как, например, UDP или RPC.
III. Ограничение производительности. Реализация сервисов-посредников требует больших вычислительных ресурсов, что ограничивает производительность.

Кроме перечисленных недостатков, сервисы-посредники могут сами по себе содержать ошибки, не учитывают информацию, содержащуюся на нижних уровнях протоколов. Они также непрозрачны для пользователя и могут потребовать специальных, модифицированных клиентских программ.

Исторически сложилось, что серверы-посредники наиболее широко используются для обеспечения безопасности сетей. Однако Интернет очень быстро развивается и растет. Растет и количество используемых протоколов, так что традиционные сервисы-посредники уже не в состоянии обеспечить адекватное взаимодействие сети организации с Интернет.

6.8.2. Технология «проверки состояния соединения»

Для разрешения перечисленных проблем, связанных с традиционными способами обеспечения безопасности при подключении к Интернет, брандмауэр Firewall-1 использует

новую технологию «проверки состояния», которая позволяет удовлетворить всем четырем требованиям к брандмауэру, высказанным в начале параграфа.

Для реализации «проверки состояния» модуль-инспектор брандмауэра анализирует информацию, получаемую на всех уровнях протоколов. Это «состояние» или «контекст» данных сохраняется и динамически обновляется, обеспечивая полное слежение за всеми соединениями и их состоянием, включая протоколы без установления соединения как такового (например, UDP или RPC). При принятии решения брандмауэром о реакции на ту или иную попытку соединения, учитывается вся информация – состояние соединений, состояние приложений, конфигурация сети, правила безопасности и т.д. На основании их принимается решение – пропустить пакет, отбросить его или зашифровать (расшифровать). Любые пакеты, любой трафик, явно не разрешенный правилами безопасности, отбрасывается, кроме того, администратору передаются сообщения о попытках несанкционированных соединений с тем, чтобы он мог предпринять какие-то действия.

Таблица 6.1.

Сравнение технологий

Требования к брандмауэру	Пакетные фильтры	Сервисы-посредники	Технология «состояния соединения»
Информация о пакете	Частично	частично	да
Информация о соединении	Нет	частично	да
Информация о состоянии приложений	Нет	да	да
Обработка информации	Частично	да	да

Обобщая, можно сказать, что технология «проверки состояния», реализованная в Firewall-1 объединяет прозрачность для пользователя, надежность, высокую производительность, свойственные пакетным фильтрам, с гибкостью и возможностью создать очень сложные правила безопасности, свойственные сервисам-посредникам.

6.8.3. Архитектура брандмауэра Firewall-1

Firewall-1 открывает новые возможности в обеспечении полного и легкого контроля доступа ко всем сервисам и компьютерам сети. Firewall-1 поддерживает полный набор возможностей протокола TCP/IP.

Брандмауэр проверяет каждый пакет, проходящий через ключевые точки сети – шлюзы Интернет, серверы, рабочие станции, маршрутизаторы или пакетные фильтры, обнаруживая и пресекая любые попытки несанкционированных соединений. Мощный механизм аудита позволяет централизовать хранение системных журналов брандмауэров сети и производить их анализ на одной рабочей станции администратора.

Firewall-1 полностью прозрачен как для пользователей, так и для приложений. Производительность систем практически не уменьшается, нет необходимости производить какие-либо изменения в конфигурации установленных программных продуктов. Кроме того, Firewall-1 может сосуществовать с другими системами обеспечения безопасности.

Модули системы Firewall-1 устанавливаются на компьютер при инсталляции соответствующего программного обеспечения и загружаются в ядро операционной системы Solaris. Модуль управления позволяет создать набор правил безопасности, который

впоследствии загружается на системы с модулем проверки. Модули проверки могут быть установлены на компьютерах, серверах или шлюзах.

При установке модуля проверки на межсетевом шлюзе, этот модуль следит за всем трафиком, проходящим между сетями. Поскольку модуль проверки находится непосредственно в ядре операционной системы, он располагается между уровнями 2 и 3 в модели ISO (то есть, между уровнем данных и сетевым уровнем). Поскольку сетевой уровень в данном случае представляет сетевая плата, а сетевой уровень – это первый уровень стека протоколов (например, TCP/IP), то можно сказать, что Frewall-1 является самым нижним программным уровнем.

Установка модуля именно таким образом позволяет Firewall-1 перехватывать и анализировать весь входящий и исходящий трафик. Ни один пакет не будет обработан ядром операционной системы, если он не соответствует правилам безопасности. Модуль проверки использует IP адреса, номера портов, а так же любую другую информацию, необходимую, чтобы определить соответствует пакет правилам безопасности или нет. Модуль проверки Firewall-1 способен разобраться во внутренней структуре протоколов семейства TCP/IP и приложений, построенных на базе этого протокола, и таким образом, способен выделить из пакета содержимое, относящееся к состоянию приложения и сохранить эту информацию для того, чтобы в дальнейшем при анализе других пакетов, возможно, учесть ее. Особенно полезным такой подход является при анализе пакетов протоколов «без установления соединения» (UDP или RPC).

Модуль управления Firewall-1 используется для настройки правил безопасности для всей сети в целом, управления модулями проверки, установленными на различных серверах и шлюзах, а так же для ведения журналов и из анализа. Модули управления существуют для различных сред – OpenLook, Windows, X/Motif. Набор утилит, используемых из командной строки, позволяет осуществлять управление без графического интерфейса, со стандартного терминала.

Модули Firewall-1 работают независимо от сетевых интерфейсов, следовательно, поддерживаются все интерфейсы, поддерживаемые операционной системой.

Эффективная, независимая от конкретного протокола технология «проверки состояния соединения», реализованная в Firewall-1, обеспечивает единое комплексное решение многих проблем, связанных с безопасностью. Использование Firewall-1 позволяет во многих случаях обойтись без отдельных пакетных фильтров или сложных правил на маршрутизаторах. Кроме того, Firewall-1 обеспечивает возможность шифрования, аутентификации, ведения системных журналов и выдачи предупреждений администратору об опасных ситуациях, наконец, простую установку и настройку. Графический интерфейс системы позволяет достаточно просто описать правила безопасности в формальном виде.

Firewall-1 обеспечивает полностью интегрированное управление безопасностью в масштабах предприятия. Другими словами, правила безопасности могут управлять многими объектами сети и реализовываться на многих уровнях. Но при этом это будут одни и те же правила безопасности и единая система журналирования.

Firewall-1 состоит из двух основных модулей: Модуля управления и Модуля брандмауэра.

Модуль управления.

Этот модуль включает в себя две части – графический интерфейс и сервер управления. Как мы видим, технология клиент-сервер используется и тут. При этом графический интерфейс является клиентом и может работать под управлением Solaris или Windows, а сервер управления работает под управлением Solaris.

Модуль брандмауэра.

Модуль брандмауэра включает в себя модуль проверки, программы-сервисы и серверы безопасности. Рисунок 6.6 демонстрирует пример, на котором один модуль управления (клиент-сервер) управляет тремя модулями брандмауэра, которые, в свою очередь, защищают три различные сети.

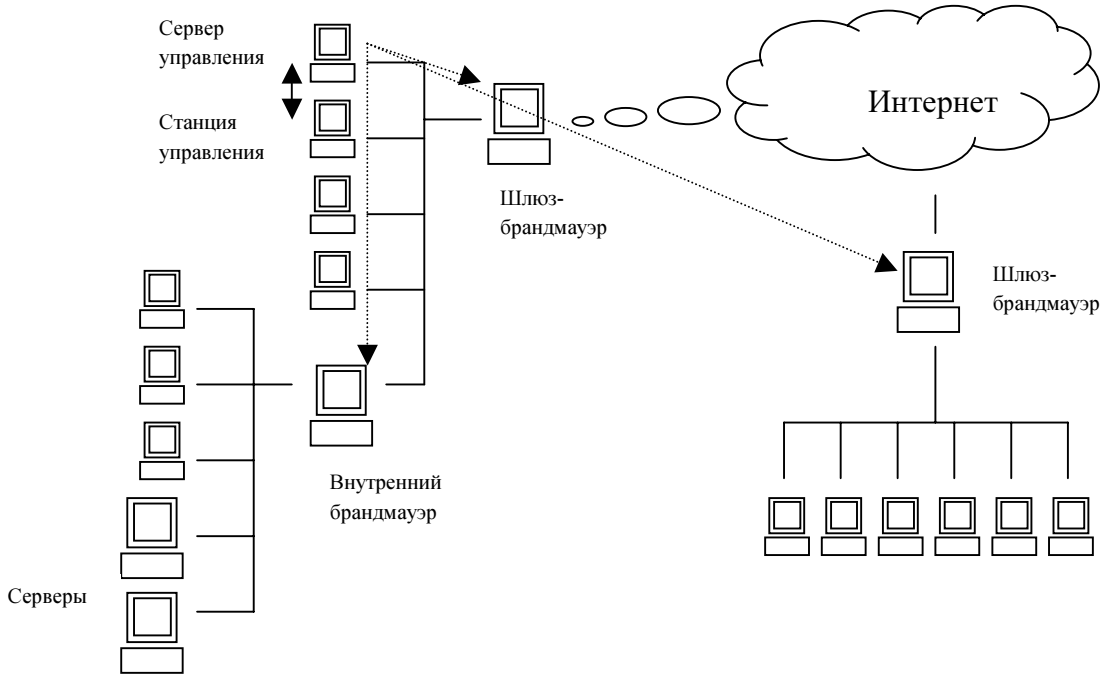


Рис. 6.6 Распределенная конфигурация Firewall-1

Правила безопасности.

Правила безопасности Firewall-1 описываются в терминах «брандмауэры», «сервисы», «пользователи», «ресурсы» и т.д. и представляют собой набор условий, определяющих взаимодействие между ними. После того, как эти правила созданы, создается описание их на внутреннем языке Firewall-1 – INSPECT, после чего скомпилированные правила передаются на шлюзы, компьютеры, серверы, маршрутизаторы и пакетные фильтры, на которых установлены модули брандмауэра, для того, чтобы они проверяли трафик на соответствие описанным правилам.

Один модуль управления может управлять многими модулями брандмауэра. Модуль брандмауэра работает независимо от модуля управления. Отметим, что модули брандмауэра могут быть установлены кроме шлюзов во внешние сети, на особо критичные сервера, на шлюзы между сетями отделов, обеспечивая защиту не только по периметру сети, но и внутри сети и между отделами. Разумеется, и модуль управления и модуль брандмауэра могут быть установлены на одной машине.

6.8.4. Модуль управления

6.8.4.1. База данных правил безопасности.

База данных правил безопасности – это упорядоченный набор правил, который полностью определяет правила, согласно которым производится управление соединениями. Каждое правило определяется в терминах источника, приемника пакета, сервиса, протокола, и в необходимых действиях – принять, отбросить, шифровать. Кроме того, описывается необходимость отметки в журналах попытки соединения.

База данных правил и свойства объектов сети (маршрутизаторов, серверов, станций, пользователей и т.д.) используются для создания командного скрипта модуля проверки, который в свою очередь, преобразуется в специальный код проверки. Этот код проверки передается по защищенным алгоритмами шифрования каналам с сервера управления на модули брандмауэра.

В случае применения пакетного фильтра, из базы данных правил безопасности выделяется список управления доступом и передается на пакетные фильтры. Для маршрутизаторов Cisco используется для этих целей протокол telnet.

Маршрутизаторы BayNetworks могут выполнять функции не только пакетных фильтров, но и модулей проверки, правда, с некоторыми ограничениями: невозможно выполнение шифрования, аутентификации и трансляции адресов.

Модуль брандмауэра использует только ту часть кода, сгенерированного на базе правил безопасности, которая касается его. При этом все журналы и предупреждения пересылаются на так называемый главный сервер. Кроме того, главный сервер содержит полный код проверки для каждого брандмауэра, которым он управляет. Как правило, главный сервер и сервер управления – это одна и та же машина, хотя возможны исключения, так как серверов управления может быть несколько. В этом случае возможно указание вторичного главного сервера на тот случай, если по каким-то причинам главный сервер будет недоступен.

6.8.4.2. Модуль управления объектами сети.

На самом деле этот модуль не управляет объектами сети, но позволяет создать описание объектов для использования в правилах безопасности брандмауэра. Только те объекты, которые описаны в модуле управления объектами, могут использоваться при создании базы данных правил безопасности. Сюда входят:

- сети и подсети;
- шлюзы;
- серверы;
- маршрутизаторы;
- коммутаторы;
- домены Интернет;
- логические серверы (серверы, состоящие их нескольких физических серверов с распределением нагрузки);
- группы вышеперечисленных сущностей.

Каждый из объектов имеет некоторый набор атрибутов, например, сетевой адрес, маску подсети и т.д. Некоторые из этих атрибутов могут быть установлены пользователем, некоторые Firewall-1 получает автоматически, из баз данных сетевой информации, таких как

файлы hosts или networks, а так же сетевой сервис NIS или NIS+. Используются так же и агенты SNMP для получения такой информации, как конфигурация интерфейсов и сетей на компьютерах и серверах, шлюзах, маршрутизаторах, коммутаторах и пакетных фильтрах. Любые объекты могут быть объединены в группы. В группы могут так же входить другие группы. Это позволяет создавать иерархические структуры объектов.

6.8.4.3. Модуль управления пользователями.

Firewall-1 позволяет управлять доступом к сети как отдельных пользователей, так и групп пользователей. Модуль управления пользователями предназначен для создания групп пользователей, управления привилегиями, паролями, способами аутентификации и т.д.

6.8.4.4. Модуль управления сервисами.

Этот модуль позволяет определить сервисы, известные брандмауэру и используемые в правилах безопасности. Все сетевые сервисы отображаются и управляются, даже если они не определены явно. Уже при установке брандмауэра определены и описаны следующие сервисы:

- Полный набор стандартных сервисов: telnet, ftp, smtp и т.д.
- Беркли r-сервисы: rlogin, rsh и т.д.
- SunRPC сервисы: NIS, NFS и т.д.
- Протоколы Интернет: http, gopher, archie и т.д.
- IP-сервисы: ICMP, RIP, SNMP и т.д.

Новые типы сервисов могут быть определены путем выбора типа сервиса и установки атрибутов сервиса. Типы сервисов включают:

- TCP
- UDP
- RPC
- ICMP
- Другие – позволяет определить сервисы и протоколы, которые не поддерживают стандартный набор атрибутов. Сервисы определяются с использованием специальных макросов и выражений.

6.8.4.5. Монитор состояния систем

Модуль брандмауэра Firewall-1 имеет возможности ведения журналов, аудита текущего состояния, проверки надежности. Модуль монитора состояния систем позволяет в одном окне посмотреть состояние всех модулей брандмауэров сети в текущий момент. Состояние включает так же в себя статистику по количеству принятых, отброшенных пакетов и отметок в журнале.

Firewall-1 устанавливает собственные агенты SNMP на каждом компьютере и маршрутизаторе, на которых работает модуль брандмауэра. Данные этих модулей могут использоваться для интеграции с платформами управления сетями, такими как OpenView или Optivity. Кроме того, Firewall-1 может использовать механизм SNMP для посылки администратору предупреждений.

6.8.4.6. Анализатор журналов.

При создании базы данных правил безопасности, администратор может указать для каких-то событий необходимость пометки в журналах. Анализатор журналов предназначен

для просмотра и анализа информации, хранимой в этих журналах. Форматы записей в журналы открыты, и могут быть легко изменены пользователем по желанию. Стандартный формат включает в себя следующие данные: источник и приемник пакета, сервис, протокол, дата и время попытки, порт источника, произведенные действия, тип записи, номер правила базы данных, приведшего к появлению записи в журнале, пользователь, и информацию о том, с какого именно модуля брандмауэра зафиксирована эта запись.

Анализатор журналов показывает список всех зафиксированных в журналах попыток соединений, изменений базы данных правил безопасности, выключения систем и т.д. Имеется возможность фильтрации записей, поиска записей, создания и печати отчетов. Широкие возможности поиска позволяют администратору легко и просто отбирать информацию, интересующую его. Отчеты создаются путем применения к определенным полям заданных критериев отбора. Полученные отчеты могут быть распечатаны, выведены в стандартный ASCII-файл или просто просмотрены.

Анализатор журналов осуществляет постоянное отслеживание изменений в журналах. Таким образом, он позволяет наблюдать в реальном времени попытки соединений. Если какие-то узлы сети не сообщают информацию в системные журналы в течение какого-то времени, возможно проверить их работоспособность с помощью SNMP прямо из анализатора журналов.

6.8.5. Модуль брандмауэра

Модуль брандмауэра в свою очередь так же не является цельным. Он состоит из модуля проверки, программ-сервисов и серверов безопасности. Как правило, модуль брандмауэра устанавливается на шлюзы, однако, может быть установлен и на сервера. Поскольку модуль проверки загружается непосредственно в ядро операционной системы, он перехватывает пакеты до того, как они будут обработаны ядром операционной системы. Кроме этого, нет необходимости удалять какие-либо сервисы или процессы на шлюзе, поскольку брандмауэр отслеживает соединения на самом нижнем уровне, непосредственно перед сетевым уровнем.

6.8.5.1. Архитектура модуля

Когда модуль брандмауэра устанавливается на шлюзе, он контролирует весь трафик между соединяемыми сетями. Модуль проверки загружается в ядро операционной системы между уровнями данных и сетевым уровнем. Поскольку уровень данных является низшим в стеке протоколов, можно сказать, что Firewall-1 является низшим программным уровнем обработки пакетов.

Работая на таком низком уровне, можно гарантировать, что проверяются все пакеты входящие и исходящие, на всех сетевых интерфейсах. Никакой пакет не будет обработан более высокими уровнями стека протоколов, до тех пор, пока модуль проверки не убедится, что он соответствует правилам безопасности.

Поскольку модуль проверки имеет доступ к пакету целиком, он может осуществить проверку всей информации, передаваемой в пакете, включая информацию, относящуюся к старшим уровням протоколов. Модуль проверки использует адрес IP, номер порта, а также любую иную необходимую информацию в пакете, для того чтобы определить соответствует пакет существующим правилам, описанным кодом проверки или нет.

Firewall-1 способен распознать информацию о внутренней структуре протоколов IP и приложений, построенных на их основе. Для протоколов без установки соединения, таких как UDP или RPC, Firewall-1 создает и хранит в памяти некий контекст соединения. Эти данные и данные, получаемые из очередного пакета, позволяют модулю проверки хранить и обновлять этот контекст, то есть поддерживать виртуальное соединение даже в том случае, если приложения работают без установления соединения. Наконец, Firewall-1 имеет возможность динамически изменять правила безопасности, чтобы разрешать или запрещать определенные соединения, если это необходимо.

Возможность доступа к полному содержимому пакета позволяет Firewall-1 запрещать отдельные команды внутри протоколов. Например, возможно разрешить ICMP ping, но при этом запретить ICMP redirect. Или разрешить прочтение информации с помощью SNMP, но запретить установку переменных SNMP. Кроме того, Firewall-1 может хранить в своих внутренних таблицах некоторые переменные и выполнять логические или арифметические операции над любой частью пакета. Кроме стандартных действий, определяемых с помощью правил безопасности, пользователи могут создавать свои алгоритмы обработки пакетов.

Пакеты, не разрешенные в правилах безопасности, отбрасываются согласно принципу «все, что явно не разрешено – запрещено».

6.8.5.2. Система проверки состояния соединения

Несмотря на то, что протокол ftp является одним из наиболее широко используемых, обеспечение его безопасности – задача довольно сложная. После того, как клиент инициирует соединение по ftp, сервер открывает обратное соединение к клиенту. Это соединение происходит со стороны сервера, то есть из внешнего мира, и производится на динамически выделяемый порт на стороне клиента. Этот номер порта в общем случае неизвестен, поскольку клиент меняет - открывает и закрывает порт довольно часто. Известно только, что номер порта больше 1023. Возможен подход, при котором брандмауэр разрешает соединения на все порты старше 1023 во внутренней сети. Несомненно, такой подход является серьезной угрозой безопасности.

Firewall-1 поступает по-другому. Модуль проверки отслеживает данные, передаваемые между клиентом и сервером по протоколу ftp. Когда клиент запрашивает у сервера обратное соединение (команда ftp PORT), модуль проверки выделяет номер порта и сохраняет его в контексте соединения. Когда происходит попытка открытия внешним ftp-сервером соединения с внутренней машиной, модуль проверки проверяет контекст соединения, и если действительно клиент запрашивал такое соединения, оно разрешается. Поскольку контекст соединения все время обновляется, то можно гарантировать, что только требуемые соединения на требуемые порты будут разрешены. Как только сессия ftp завершается, порты закрываются, больше модуль проверки не допустит соединения на этот адрес. Это позволяет гарантировать максимальную безопасность.

Протоколы UDP или RPC, как уже говорилось выше, представляют собой особую проблему безопасности, поскольку являются протоколами без установления соединения, то есть без контекста на уровне приложений. Firewall-1 обеспечивает безопасность таких протоколов путем создания и динамического поддержания виртуального контекста, основываясь на информации, передаваемой в пакетах.

Протоколы, основанные на UDP (например, DNS или WAIS) особенно сложно отфильтровать с помощью пакетных фильтров, поскольку в них нет никакой явной разницы между запросом и ответом. Таким образом, есть выбор – либо просто запретить все UDP соединения, либо разрешить практически все двунаправленные соединения по протоколу UDP, что влечет за собой опасность для безопасности сети.

Firewall-1 обеспечивает безопасность приложений на базе UDP путем установления и поддержки виртуального соединения. Модуль проверки создает некий контекст соединения и проверяет состояние каждого такого виртуального соединения, проходящего через шлюз. Каждый новый запрос по протоколу UDP записывается в таблице соединений, и каждый пакет UDP, приходящий в обратном направлении проверяется на соответствие этой таблице. Если ответ не приходит за заданный промежуток времени, запись в таблице удаляется. При этом считается, что вышло время ожидания пакета. Таким образом, можно гарантировать, что только запрошенные пакеты преодолеют брандмауэр.

Аналогично UDP, простое отслеживание номеров портов при обслуживании протокола RPC неэффективно, поскольку RPC не использует заранее определенных номеров портов. Распределение портов в этом протоколе – динамическое и часто меняется за короткие промежутки времени.

Firewall-1 динамически и совершенно прозрачно для приложений отслеживает распределение портов RPC, путем слежения за программами распределения портов portmapper. Firewall-1 отслеживает обращения к portmapper и на основании их создает и поддерживает таблицу состояний RPC, в которой учтено какие порты использует каждый сервис в настоящий момент.

Когда модуль проверки проверяет очередной пакет RPC на соответствие правилам безопасности, он обращается к этой таблице, сравнивает номера портов в пакете и в соответствующей записи в таблице и на основании этого делает вывод к какому именно сервису направлен этот пакет.

Если номер порта в пакете не соответствует ни одной записи в таблице RPC – а это может случиться, если программа, пославшая пакет использует какие-то начальные сведения о номере порта и не обращается к portmapper, Firewall-1 самостоятельно делает запрос к portmapper и проверяет, что за сервис использует этот номер порта.

6.8.5.3. Аутентификация

Система Firewall-1 обеспечивает три типа аутентификации:

- Аутентификация пользователей, позволяет администраторам предоставлять определенные привилегии отдельным пользователям. Система аутентификации пользователей включает в себя специальные серверы аутентификации, например, сервер аутентификации http, предназначенный для проверки пользователей как при исходящих http-соединениях, так и при входящих.
- Аутентификация клиентов. Предоставляет механизм аутентификации пользователей для всех приложений, как стандартных, так и нестандартных.
- Аутентификация сессий. Обеспечивает механизм прозрачной аутентификации отдельный сессий, который может быть интегрирован с любыми TCP, UDP или RPC приложениями.

Аутентификация пользователей

Firewall-1 позволяет администратору при создании правил безопасности оперировать не только такими проверками, как адрес источника, адрес приемника пакета и номера портов или сервисов, но так же определять возможности использования определенных сервисов конкретными пользователями.

Серверы аутентификации Firewall-1 работают на уровне приложений. При попытке соединения с некоторым сервисом, серверы аутентификации, перед тем как предоставить доступ, производят аутентификацию пользователя по заранее заданной схеме. Схема аутентификации задается для каждого пользователя индивидуально с помощью модуля управления пользователями (см. выше). Если для пользователя не было указано никакой схемы аутентификации, но в правилах описано, что аутентификация необходима – в доступе отказывается.

Даже после того, как пользователь прошел аутентификацию, Firewall-1 не позволяет напрямую открыть соединение с требуемым адресом. Вместо этого открывается отдельная сессия между брандмауэром и этим адресом. Таким образом, пользователь работает все время через брандмауэр. Когда приходит очередной пакет от пользователя, он обрабатывается модулем проверки на низшем уровне, затем передается на уровень приложения – серверу аутентификации, после чего опять «спускается» вниз, к модулю проверки и только после этого передается на реальный адрес-приемник. На каждом шаге пакет может быть проверен и отмечен в журналах. При этом между пользователем и реальным сервером существует посредник – сервер аутентификации, однако, его работа совершенно прозрачна для пользователя, таким образом, пользователь может даже не знать о том, что он общается с ресурсом через посредника.

Несмотря на то, что правила безопасности реализуются на разных уровнях разными программами (модуль проверки и сервер аутентификации), используется общая база правил безопасности. Распределение функций происходит автоматически и прозрачно для администратора. Никакого дополнительного управления не требуется.

Аутентификация пользователей позволяет произвести аутентификацию только протоколов telnet, ftp, rlogin, http. Этого, конечно, недостаточно. Поэтому существует механизм аутентификации клиентов. Этот механизм позволяет аутентифицировать любые приложения – на основе протоколов TCP, UDP или RPC. При этом не требуется никаких изменений в программах ни на клиентской, ни на серверной стороне. Администратор может определить, каким образом будет проходить аутентификация конкретных клиентов, какие сервисы и приложения при этом будут доступны, в какой время и по каким датам, как много сессий может быть открыто одновременно. При этом после того, как аутентификация пройдена, модуль проверки обеспечивает максимальную производительность обработки пакетов. При этом не используется никаких сервисов-посредников.

Аутентификация сессий предназначена для проверки полномочий на каждую сессию с каким-либо сервисом. Процедура аутентификации при этом происходит следующим образом:

I. пользователь пытается открыть сессию напрямую к сервису.
II. Модуль проверки Firewall-1 производит соединение с агентом аутентификации сервисов, который возвращает необходимую для аутентификации информацию. При

этом агент аутентификации – это написанная пользователем программа, взаимодействующая с модулем проверки с помощью специального протокола.

III. Если аутентификация прошла успешно, модуль проверки позволяет пользователю произвести соединение.

6.8.5.4. Шифрование

Целью Политики безопасности организации является не только защита от «взлома». Поскольку, в современном мире довольно часто для связи между подразделениями организации используются сети общего пользования, необходимо обеспечить защиту передаваемой по ним информации. То есть обеспечить целостность и конфиденциальность. Ранее мы уже упоминали, что лучшим средством для достижения этих целей является шифрование.

Встроенные в пакет Firewall-1 возможности шифрования информации позволяют легко строить и поддерживать приватные сети организации через сети публичного пользования.

Алгоритмы шифрования Firewall-1 обладают очень высокой производительностью – до 10Мбит/сек при использовании обычной рабочей станции. Управление алгоритмами шифрования встроено в редактор базы правил безопасности, возможно направление сообщений о работе шифрования в системные журналы брандмауэра. Для шифрования могут использоваться алгоритмы – FWZ, SKIP, RC4.

6.8.5.5. Трансляция адресов

Необходимость в трансляции адресов – то есть в подмене адресов в заголовках IP-пакетов, может возникать в двух случаях:

- Администратор сети желает скрыть внутреннюю структуру своей сети. Для этого могут быть самые разные причины.
- Внутренняя сеть имеет фиктивные IP-адреса. Это возникает, в основном, по историческим причинам. Многие сети строились на базе протокола IP без подключения к Интернет. Когда же такое подключение осуществляется, наличие фиктивных адресов способно вносить неразбериху в Интернет, а перевод всей организации на действительные адреса стоит дорого и не оправдан.

В обоих случаях, внутренние IP адреса не могут быть использованы для доступа в Интернет. Однако, вполне возможно, что для внутренних машин доступ к Интернет необходимо обеспечить.

Исторически сложилось так, что для решения таких проблем использовались серверы-посредники или прокси. В этом случае, например, для осуществления подключения через telnet к внешнему адресу, необходимо сначала выполнить подключение к серверу-посреднику, а затем – к внешнему адресу. Firewall-1 может быть легко настроен для реализации подобной схемы для многих протоколов (ftp, telnet, http и т.д.). Более того, Firewall-1 способен обеспечить дополнительную аутентификацию пользователей на шлюзе.

С другой стороны, использование сервера-посредника имеет следующие недостатки:

- Сервер-посредник предназначен только для определенного типа сервиса. Невозможно использовать сервисы, для которых не существует серверов-посредников.

- Использование сервера-посредника не прозрачно для пользователей. В некоторых случаях, для использования прокси требуется модификация программного обеспечения клиентов. Кроме того, использование сервера-посредника создает очень высокую нагрузку на шлюз. То есть происходит нерациональное использование ресурсов.
- Довольно сложно создать прокси для протоколов, отличных от TCP: например, UDP или RPC.

Для разрешения этих недостатков, Firewall-1 использует схему трансляции адресов «на лету», что представляет собой полное и эффективное решение. Администратор принимает решение о том, какое подмножество внутренних адресов необходимо скрыть от внешней сети или от Интернет и задает диапазон адресов, которые могут использоваться вместо скрытых. Возможно также настроить брандмауэр таким образом, чтобы внутренние адреса (нелегальные) могли быть доступны из внешней сети. При этом создается таблица трансляции адресов, внешние клиенты пытаются выполнить соединение с реальным адресом, шлюз изменяет его на фиктивный при проходе пакета из внешней сети во внутреннюю.

Использование трансляции адресов возможно для осуществления «односторонней маршрутизации», то есть маршрутизация из внутренней сети наружу выполняется, а обратная маршрутизация невозможна.

Скорость подмены очень велика за счет эффективных алгоритмов. При использовании в качестве шлюза недорогой рабочей станции SPARC возможна работа со скоростями до 10Мбит/сек.

7. Средства обеспечения безопасности в ОС Linux

В этой главе мы рассмотрим основные моменты, связанные с безопасностью, применительно к ОС Linux. В настоящее время эта операционная система является одной из наиболее распространенных среди некоммерческих ОС. В то же время, несмотря на отсутствие поддержки, характерной для коммерческих ОС, Linux имеет одно большое преимущество: исходные тексты всех компонентов системы открыты для свободного использования. Открытость исходного кода дает возможность, при достаточной квалификации, самостоятельно модифицировать систему, что позволяет повысить ее безопасность. Результатом такой модификации является система SystemGuard, разработанная автором.

Отсутствие расширенных списков контроля доступа (ACL) не позволяет рекомендовать ОС Linux как файловый сервер, но открытость исходного кода позволяет модифицировать систему так, что дистанционная атака становится практически невозможной. Это позволяет использовать Linux в самых неблагоприятных с точки зрения безопасности условиях: на WWW-сервере компании. Действительно: на этом сервере (см. главу 5) не должно быть никаких других сервисов. Даже публичный FTP-сервис с возможностью записи не рекомендуется устанавливать на одной машине с www-сервисом. Работать с www-сервером должен специально выделенный пользователь – webmaster, и только он. Поэтому для данной машины хватает и стандартной маски прав UNIX. А разработанная автором система защиты стека от исполнения позволяет практически исключить возможность осуществления атаки с использованием переполнения буфера (наиболее популярного класса дистанционных атак).

В операционной системе Linux существует встроенный механизм централизованного ведения системных журналов, и прочие средства обеспечения безопасности, описанные в предыдущей главе. Все эти механизмы призваны обеспечить простоту управления и настройки при высокой защищенности информации в системах и сетях на базе Linux. Именно этим возможностям и посвящена эта глава.

Методы обеспечения безопасности Linux сходны с методами для Solaris, ведь Linux – тоже реализация OS UNIX. В этой главе приведены только специфические особенности Linux.

7.1. Обзор

Linux –свободно распространяемая на условиях GNU Public License реализация System V release 4. Основными ее достоинствами являются низкие требования к оборудованию, открытый исходный код и быстрое исправление ошибок. Недостатком данной ОС является отсутствие официальной поддержки. Это не такой уж и большой недостаток, принимая во внимание то, что практически в каждой фирме есть специалист, занимающийся поддержкой программного и аппаратного обеспечения, наличие большого количества документации, и то, что ОС Linux проста в установке и обслуживании.

7.2. Атаки, использующие особенности файловой системы UNIX, и защита от них

Файловая система ОС Linux очень похожа на файловую систему ОС Solaris. Поэтому мы не будем описывать ее структуру, а сразу перейдем к модификациям, целью которых является повышение устойчивости ОС к атакам через файловую систему.

Данный класс атак использует ссылки и символические ссылки для того, чтобы получить доступ к файлу, принадлежащему другому пользователю. Классическим примером является файл /etc/passwd, содержащий информацию о пользователях системы. Предположим, злоумышленник создает ссылку из директории /tmp на файл /etc/passwd, и называет данную ссылку, например, X11. Система X Window, запускаемая от имени суперпользователля, использует файл /tmp/X11для собственных нужд. Если теперь заставить X Window открыть файл /tmp/X11 – мы получим содержимое файла /etc/passwd – весьма ценную для атакующего информацию (имена пользователей, их идентификаторы, а если в системе нет затенения паролей – то и их пароли.).

Для защиты от данного класса атак автор внес изменения в ядро ОС, не позволяющие процессу следовать по ссылке, владельцем которой он не является, если на директорию установлен флаг «Sticky bit», а также запрещающие обычному пользователю создание жестких ссылок на чужие файлы. Данные изменения являются частью системы «SystemGuard», разработанной автором.

Ниже приведен модифицированный код ядра Linux, реализующий данные исправления. Данные изменения касаются файла /fs/namei.c из исходного кода ядра Linux версии 2.0.35

```
     …..
int follow_link(struct inode *dir, struct inode *inode, int flag, int mode,
struct inode **res_inode)
{
      if (!dir || !inode) {     /* Если номер узла каталога или файла =0 */
            iput(dir);          /* выбрасывем узлы из таблицы занятых    */
            iput(inode);        /* и                                     */
            *res_inode = NULL;  /* результат, естественно, никакой…      */
            return -ENOENT;     /* возвращаемся с ошибкой                */
      }
      if (!inode->i_op || !inode->i_op->follow_link) {
            iput(dir);          /* Если мы пришли сюда не за тем,            */
            *res_inode = inode; /* чтобы пойти по ссылке                     */
            return 0;           /* результат – тот узел, с которого мы пришли*/
      }
#ifdef CONFIG_SECURE_LINK
/*
 * Не ходить по ссылкам, которыми не владеем, в каталоге со Sticky bit
 * Ну, разве что если ссылка принадлежит суперпользователю…
 */
      if (S_ISLNK(inode->i_mode)             /* Не просто файл, а ссылка ;)   */
          && (dir->i_mode & S_ISVTX)         /* На директории – Sticky Bit    */
          && inode->i_uid                    /* И создал файл не root         */
          && current->fsuid != inode->i_uid) /* И пользуется ссылкой не хозяин*/
      {
            security_alert("symlink", printk(KERN_ALERT "Link owned by /*ВРАГИ*/
                           %d.%d.\n", inode->i_uid, inode->i_gid)); /*АТАКУЮТ*/
            iput(dir);                                              /* НАС*/
            iput(inode);           /*Освобождаем узлы.         */
            *res_inode = NULL;     /* Какой-такой результат??? */
            return -EPERM;         /* НЕ ПУЩАТЬ!               */
      }
#endif
```

```
        return inode->i_op->follow_link(dir,inode,flag,mode,res_inode);
}
      ......
      static int do_link(struct inode * oldinode, const char * newname)
      {
        struct inode * dir;
        const char * basename;
        int namelen, error;

        error = dir_namei(newname, &namelen, &basename, NULL, &dir);
        if (error) {              /* Нельзя создать жесткую ссылку на каталог! */
              iput(oldinode);     /* освободить узел                           */
              return error;       /* вернуть ошибку                            */
        }
        if (!namelen) {           /* Длина имени ссылки равна нулю ?           */
              iput(oldinode);     /* Что ж… хорошая ссылка… но неправильная     */
              iput(dir);          /* освободим узлы                            */
              return -EPERM;      /* и вернем ошибку                           */
        }
      #ifdef CONFIG_SECURE_LINK
      /*
       * Не позволим обычному пользователю создать жесткую ссылку на чужой файл!
       */
        if (current->fsuid != oldinode->i_uid    /* Если файл не наш            */
                    && !fsuser() && !suser()) { /* И мы - не суперпользователь */
              security_alert("hard link", printk(KERN_ALERT /* сообщить        */
                    "File owned by %d.%d.\n",              /* куда следует   */
                    oldinode->i_uid, oldinode->i_gid));
              iput(oldinode);                  /* Удалить узлы   */
              iput(dir);                       /*                */
              return -EPERM;                   /* Вернуть ошибку */
        }
      #endif
        if (IS_RDONLY(dir)) { /* Каталог только для чтения… какая жалость */
              iput(oldinode);
              iput(dir);
              return -EROFS;
        }
        if (dir->i_dev != oldinode->i_dev) { /* Цель и ссылка на разных */
              iput(dir);                     /* устройствах… так нельзя */
              iput(oldinode);
              return -EXDEV;
        }
        if ((error = permission(dir,MAY_WRITE | MAY_EXEC)) != 0) {
              iput(dir);      /* Надо же! Писать нельзя в каталог… */
              iput(oldinode); /* Как не повезло…                   */
              return error;
        }
        /*
         * Если файл только для добавления или неизменяемый - тоже нельзя
         */
        if (IS_APPEND(oldinode) || IS_IMMUTABLE(oldinode)) {
              iput(dir);
              iput(oldinode);
              return -EPERM;
        }
        if (!dir->i_op || !dir->i_op->link) {
              iput(dir);       /* жесткая ссылка на символическую? */
              iput(oldinode);  /* что за бред…                     */
              return -EPERM;
        }
        dir->i_count++; /* Ура! Все правильно! Делаем ссылку ;) */
        if (dir->i_sb && dir->i_sb->dq_op)
              dir->i_sb->dq_op->initialize(dir, -1);
        down(&dir->i_sem);
```

```
        error = dir->i_op->link(oldinode, dir, basename, namelen);
 up(&dir->i_sem);
 iput(dir);
 return error; /* Возвращаем результат создания ссылки. */
}
```

То же самое сделано и для именованных каналов, для предотвращения несанкционированного доступа к данным. Программная реализация данной модификации выглядит так:

Файл /fs/namei.c из исходного кода ядра Linux версии 2.0.35

```
……
int open_namei(const char * pathname, int flag, int mode,
               struct inode ** res_inode, struct inode * base)
……
#ifdef CONFIG_SECURE_PIPE
/*
 * Не давать пользователю писать в чужие именованные каналы,
 * если эти каналы создавал не суперпользователь
 */
 if ((S_ISFIFO(inode->i_mode)  /* Если открываем именованный канал      */
   || S_ISSOCK(inode->i_mode)) /* Или сокет                              */
   && (dir->i_mode & S_ISVTX) /* И установлен Sticky bit на каталог      */
   && (flag & FMODE_WRITE)   /* И открываем его для записи               */
   && inode->i_uid          /* И владелец процесса – не суперпользователь*/
   && current->fsuid != inode->i_uid /* И канал чужой                    */
   && current->uid != inode->i_uid) {
       security_alert("pipe", printk(KERN_ALERT /* Сообщить,    */
             "Pipe owned by %d.%d.\n",          /* куда следует */
             inode->i_uid, inode->i_gid));
       iput(inode);
       return -EPERM;         /* Не разрешать такое безобразие */
 }
#endif
.……
```

7.3. Атаки, использующие переполнение буфера, и защита от них

Как уже упоминалось в разделе 6.3, атаки с использованием переполнения буфера являются самым популярным классом атак. Система «SystemGuard» успешно используется для борьбы и с этим классом атак. Данная система защищает систему от атак с переполнением буфера следующими методами:

- запрещает исполнение кода в стеке
- перемещает точку размещения в адресном пространстве пользователя разделяемых библиотек так, чтобы первый байт адреса был равен 0. Это действует на атаки, использующие подмену адреса возврата адресом функции стандартной библиотеки. Встретившийся 0 прервет выполнение функции strcpy(), и копирование зловредного кода завершится досрочно, что в худшем случае приведет к вызову функции без параметров.

Перенос точки размещения разделяемых библиотек происходит следующим образом: в файле /include/asm-i386/processor.h модифицирована инлайн-функция MMAP_SEARCH_START. Ниже приведен модифицированный фрагмент кода.

```
#if defined(CONFIG_SECURE_STACK)&&defined(CONFIG_BINFMT_ELF)
```

```
extern struct linux_binfmt elf_format;
#define MMAP_SEARCH_START
(current->binfmt==&elf_format         /* Загружаем ELF формат         */
&&!(current->flags& PF_STACKEXEC_F)   /* И исполнение стека запрещено */
?0x00110000UL /* Библиотеки размещаем по адресу 0x00110000 */
:TASK_SIZE/3) /* Иначе - по адресу 1/3 пространства адресов пользователя */
#else
#define MMAP_SEARCH_START (TASK_SIZE/3)
#endif
```

Для запрета исполнения кода в стеке была проделана следующая работа:

В заголовок исполняемых файлов введен флаг, определяющий, можно ли данной программе выполнять код в стеке. Изменению подверглись следующие файлы:

/fs/binfmt_aout.c (Определение флага для формата a.out)
/fs/binfmt_elf.c (Определение флага для формата ELF)
/fs/exec.c (Установка флага для процесса при запуске)

Добавлены 2 сегмента в GDT (файл /arch/i386/kernel/head.S):

```
......
ENTRY(gdt)
  .quad 0x0000000000000000  /* NULL            */
  .quad 0x0000000000000000  /* не используется */
  .quad 0xc0c39a000000ffff  /* 0x10 1GB кода ядра с адреса 0xC0000000        */
  .quad 0xc0c392000000ffff  /* 0x18 1GB данных ядра с адреса 0xC0000000      */
#ifdef CONFIG_SECURE_STACK
  .quad 0x00cafa000000ffff  /* 0x23 2.75GB кода пользователя с адреса 0      */
  .quad 0x00cbf2000000ffff  /* 0x2b 3GB данных пользователя с адреса 0       */
  .quad 0x00cbda000000ffff  /* 0x32 3GB кода пользователя с адреса 0, DPL=2  */
  .quad 0x00cbd2000000ffff  /* 0x3a 3GB стека пользователя с адреса 0, DPL=2 */
#else
  .quad 0x00cbfa000000ffff  /* 0x23 3GB кода пользователя с адреса 0x00000000 */
  .quad 0x00cbf2000000ffff /* 0x2b 3GB данных пользователя с адреса 0x00000000*/
  .quad 0x0000000000000000  /* не используется */
  .quad 0x0000000000000000  /* не используется */
#endif
  .fill 2*NR_TASKS,8,0        /* место под LDT, TSS и прочее */
......
```

Модифицирован файл /include/asm-i386/segment.h: определены названия для новых сегментов, «магические» адреса возврата для определения того, что мы возвращаемся из трамполайна GCC или из обработчика сигнала, усовершенствована функция подготовки сегментных регистров процесса.

```
......
#define USER_CS         0x23
#define USER_DS         0x2B

#ifdef CONFIG_SECURE_STACK
#define USER_HUGE_CS    0x32
#define USER_HUGE_SS    0x3A
#else
#define USER_HUGE_CS    0x23
#define USER_HUGE_SS    0x2B
#endif

/*
 * Специальные «магические» адреса для возврата в ядро из трамплайнов GCC
 * и обработчиков сигналов. Подойдет любой адрес за пределами пользовательского
 * сегмента кода. При переходе на данный адрес произойдет ошибка защиты, и
 * управление перейдет в ядро.
```

```
        */
#define MAGIC_SIGRETURN 0xC0DE0001 /* Возврат из трамплайна          */
#define MAGIC_TRAMP     0xC0DE0002 /* Возврат из обработчика сигнала */
......
static inline void start_thread(struct pt_regs * regs, unsigned long eip,
unsigned long esp)
/* Заполнение сегментных регистров для процесса */
{
#ifdef CONFIG_SECURE_STACK
      if (current->flags & PF_STACKEXEC_C) { /* Если стек исполняемый */
            regs->cs = USER_HUGE_CS; /* Заполняем сегментные регистры */
            regs->ss = USER_HUGE_SS; /* Стек - в специальном сегменте */
      } else {
            regs->cs = USER_CS; /* Заполняем сегментные регистры */
            regs->ss = USER_DS; /* Стек - в сегменте данных      */
      }
      regs->ds = regs->es = regs->fs = regs->gs = USER_DS;
#else
      regs->cs = USER_CS;
      regs->ds = regs->es = regs->fs = regs->gs = regs->ss = USER_DS;
#endif
      regs->eip = eip;
      regs->esp = esp;
}
```

Скорректированы также функции, использовавшие номера, а не имена дескрипторов, в следующих файлах:

```
/arch/i386/kernel/ptrace.c,
/arch/i386/kernel/signal.c
/arch/i386/kernel/traps.c
/arch/i386/math-emu/fpu_entry.c
/arch/i386/mm/fault.c
```

Модифицирована обработка сигналов: При подготовке стека для обработчика сигнала реальный адрес возврата заменяется в сетке на адрес MAGIC_SIGRETURN, чтобы передать по выходу из обработчика сигнала управление в ядро (через ошибку защиты).

```
static void setup_frame(struct sigaction * sa, struct pt_regs * regs,
                        int signr, unsigned long oldmask)
{
      unsigned long * frame;

      frame = (unsigned long *) regs->esp; /* Формируем указатель на стек*/
      if (regs->ss != USER_DS && regs->ss != USER_HUGE_SS && sa->sa_restorer)
            frame = (unsigned long *) sa->sa_restorer;
      frame -= 64;
      if (verify_area(VERIFY_WRITE,frame,64*4)) /* Можно ли писать в стек? */
            do_exit(SIGSEGV);

/* Готовим «нормальный стек» */
#ifdef CONFIG_SECURE_STACK
      put_user((unsigned long)MAGIC_SIGRETURN, frame);
      /* Заменяем адрес возврата на MAGIC_SIGRETURN, */
      /* если не разрешено исполнение стека          */
#else
#define __CODE ((unsigned long)(frame+24))
#define CODE(x) ((unsigned long *) ((x)+__CODE))
      put_user(__CODE,frame);
#endif
.. .. ..
/* Готовим стек… */
.. .. ..
```

```
#ifndef CONFIG_SECURE_STACK
/* Готовим код возврата как обычно, если стек исполняемый */
      put_user(0x0000b858, CODE(0));      /* popl %eax ; movl $,%eax */
      put_user(0x80cd0000, CODE(4));      /* int $0x80 */
      put_user(__NR_sigreturn, CODE(2));
#undef __CODE
#undef CODE
#endif

#ifdef CONFIG_SECURE_STACK
/*
 * Временно разрешаем исполнение стека, если мы пришли в обработчик
 * сигнала через трамполайн (обработчик сигнала - вложенная функция).
 * ОЧЕНЬ редкий случай. Но бывает.
 */
      if (((unsigned long)sa->sa_handler & 0xF0000000) == 0xB0000000)
            current->flags |= PF_STACKEXEC_C;
#endif

      /* Set up registers for signal handler */
      start_thread(regs, (unsigned long)sa->sa_handler, (unsigned long)frame);
      regs->eflags &= ~TF_MASK;
}
```

Модифицирован также обработчик общей ошибки защиты (файл /arch/i386/kernel/traps.c):

```
asmlinkage void do_general_protection(struct pt_regs * regs, long error_code)
{
#ifdef CONFIG_SECURE_STACK
      unsigned long retaddr;
#ifdef CONFIG_SECURE_STACK_SMART
      unsigned char regnum;
#endif
#endif
.. .. ..
#ifdef CONFIG_SECURE_STACK
/* А не из обработчика ли сигнала мы возвращаемся? */
      if (regs->cs == USER_CS || regs->cs == USER_HUGE_CS)
      if (get_seg_byte(USER_DS, (char *)regs->eip) == 0xC3)
      if (!verify_area(VERIFY_READ, (void *)regs->esp, 4))
      if ((retaddr = get_seg_long(USER_DS, (char *)regs->esp)) ==
          MAGIC_SIGRETURN) {
/*
 * Вызываем sys_sigreturn() для того, чтобы восстановить контекст процесса
 */
            regs->esp += 8;
            __asm__("movl %3,%%esi;"
                  "subl %1,%%esp;"
                  "movl %2,%%ecx;"
                  "movl %%esp,%%edi;"
                  "cld; rep; movsl;"
                  "call sys_sigreturn;"
                  "leal %3,%%edi;"
                  "addl %1,%%edi;"
                  "movl %%esp,%%esi;"
                  "movl (%%edi),%%edi;"
                  "movl %2,%%ecx;"
                  "cld; rep; movsl;"
                  "movl %%esi,%%esp"
            :
```

```
        /* %eax возвращаем тут */
             "=a" (regs->eax)
             :
             "i" (sizeof(*regs)),
             "i" (sizeof(*regs) >> 2),
             "m" (regs)
             :
             "cx", "dx", "si", "di", "cc", "memory");
             return;
      }

#ifdef CONFIG_SECURE_STACK_EMULATE
/* А не трамплайн ли это был? */
      else if (regs->cs == USER_HUGE_CS && retaddr == MAGIC_TRAMP) {
             current->flags &= ~PF_STACKEXEC_C;
             regs->cs = USER_CS; regs->ss = USER_DS;

             regs->eip = current->trampoline_retaddr;
             regs->esp += 4;
             return;
      }
#endif

/*
 * Проверка, не находится ли адрес возврата в стековой области.
 * Такое может случиться только в переполнения буфера
 */
      else if (regs->cs == USER_CS &&           /* Стек защищен от исполнения */
          (retaddr & 0xF0000000) == 0xB0000000) /* Адрес возврата в стеке     */
             security_alert("buffer overflow", {}); /* Сообщить, кому следует  */
#endif

      die_if_kernel("general protection",regs,error_code);
 /* Все же это была ошибка защиты… убиваем процесс, вызвавший ее */

#if defined(CONFIG_SECURE_STACK) && defined(CONFIG_SECURE_STACK_SMART)
/* Если выполнять стек запрещено, но трамплайны разрешены, то
 * переключаемся на большой сегмент кода (разрешаем выполнение стека), если
 * вызвавшая ошибку инструкция - call %reg, но не call %esp.
 */
      if (regs->cs == USER_CS)
      if (get_seg_byte(USER_DS, (char *)regs->eip) == 0xFF) {
             regnum = get_seg_byte(USER_DS, (char *)(regs->eip + 1));
             if ((regnum & 0xD8) == 0xD0 && regnum != 0xD4) {
                    current->flags |= PF_STACKEXEC_C;
                    regs->cs = USER_HUGE_CS; regs->ss = USER_HUGE_SS;

#ifdef CONFIG_SECURE_STACK_EMULATE
/*
 * Эмулируем вызов, чтобы вернуть управление в ядро при возврате из трамплайна
 * и запретить исполнение стека.
 */
                    current->trampoline_retaddr = regs->eip + 2;

                    regs->esp -= 4;
                    if (verify_area(VERIFY_WRITE, (void *)regs->esp, 4)) {
                           force_sig(SIGSEGV, current);
                           return;
                    }
                    put_user(MAGIC_TRAMP, (unsigned long *)regs->esp);
                    /* пишем MAGIC_TRAMP вместо адреса возврата */
                    switch (regnum) {
                           case 0xD0: regs->eip = regs->eax; break;
                           case 0xD1: regs->eip = regs->ecx; break;
                           case 0xD2: regs->eip = regs->edx; break;
```

```
                        case 0xD3: regs->eip = regs->ebx; break;
                        case 0xD5: regs->eip = regs->ebp; break;
                        case 0xD6: regs->eip = regs->esi; break;
                        case 0xD7: regs->eip = regs->edi;
                    } /* Эмулируем call через присваивание eip нового значения */
#endif
                    return; /* и возвращаемся к выполнению */
                }
        }
#endif

        current->tss.error_code = error_code;
        current->tss.trap_no = 13;
        force_sig(SIGSEGV, current);
}
```

Модуль защиты встроен в ядро ОС как опциональный, то есть при компиляции ядра защиту стека можно отключить, хотя автор не рекомендует это делать. Если какой – либо программе нужен исполняемый стек – рекомендуется подумать, стоит ли рисковать безопасностью сервера из-за неё. Если все же было принято решение, что рисковать стоит – лучше воспользоваться прилагаемой утилитой sse (Set Stack Execution) системы SystemGuard для установки соответствующего флага в заголовке бинарного файла, не отключая саму систему.

Данная модификация ядра позволила защититься от целого класса атак без нарушения работы существующих программ. Конечно, установка такой защиты не подразумевает, что можно совсем забыть об обновлении программного обеспечения. Это нужно делать хотя бы потому, что работа атакованного сервиса при атаке прекращается (это неизбежно, ведь нарушается структура стека процесса), а для перезапуска сервиса (даже если перезапуск осуществляется автоматически) требуется время.

8. Заключение

В работе рассмотрены основные стандарты, принятые в мире, для оценки безопасности информационных систем. К сожалению, ни один из стандартов не дает рекомендаций по поддержанию режима безопасности или оценке безопасности информационных систем.

Для того чтобы обеспечить возможность практического обеспечения режима безопасности в конкретных, подчас довольно сложных, системах, автором разработана и предложена модель безопасности информационной системы. Целью создания этой модели является декомпозиция произвольной сложной информационной системы на базе сетевых технологий на некие простые абстрактные составляющие. Выдвигая требования по обеспечению безопасности к таким составляющим, можно относительно просто составить требования к безопасности системы в целом. Аналогично, произведя анализ и оценку безопасности отдельных компонент, можно оценить безопасность всей системы.

На базе предложенной модели и на основании опыта работы и практического обеспечения безопасности информационных систем, автором разработаны и изложены рекомендации по оценке безопасности, управлению рисками и по созданию безопасных систем.

Особо необходимо отметить рекомендации по созданию и поддержанию Политики безопасности. Поскольку именно с этого момента разговор о безопасности системы становится в достаточной степени конкретным. К сожалению, в большинстве современной литературы политике безопасности уделяется немного внимания.

Изложенные в работе принципы и подходы получили практическое использование в работе Томского Инновационного Центра Западной Сибири. В течение более чем двух лет в этой организации автором проводились работы и исследования в области информационной безопасности. Достигнуты вполне конкретные практические результаты. Так, например, в течение только одного 1998 года сеть и система защиты информации ТИЦ ЗС выдержали три крупных атаки со стороны Интернет, отказы электропитания, ошибки персонала при работе с данными. Тем не менее, реализация мер по защите информации позволила полностью избежать потерь данных. Особо следует упомянуть крупномасштабную атаку на сеть Томского Государственного Университета в январе 1998 года, когда от действий злоумышленников пострадало более 10 серверов. Во время этой атаки были предприняты попытки преодоления защиты брандмауэра и www-сервера ТИЦ ЗС. Оба администратора в этот момент находились в командировке и не могли принять меры по противодействию атаке. Однако построенная защита с честью выдержала испытание и так и не была преодолена.

Конечно, многие и даже большинство рассмотренных вопросов относятся не только к UNIX, но так же могут быть учтены и в других системах. Каковы же основные моменты обеспечения безопасности?

Во-первых, для того, чтобы вообще имело смысл говорить о безопасности информации, необходимо понимание и поддержка в данном вопросе со стороны руководства организации. К сожалению, в наше время еще далеко не все руководители понимают важность этой проблемы. Это чревато двумя опасностями: с одной стороны, сдерживается внедрение автоматизированных систем из-за недоверия к ним, в частности в части безопасности

информации, с другой стороны, там, где внедрение проводится, существует опасность потерь информации, что так же чревато существенными убытками. Усилия же одного технического персонала, системных администраторов не могут дать требуемого результата без поддержки руководства.

Во-вторых, каждой организации необходимо четко сформулировать политику безопасности. Вероятно, кому-то такая формулировка и ее формальное утверждение руководством покажется излишней бюрократией, однако это не так. Четко высказанная политика безопасности позволяет координировать усилия, предпринимаемые в разных отделах, службах, и направленные на обеспечение безопасности. Кроме того, политика безопасности позволяет более эффективно управлять действиями персонала, поскольку определяет допустимое и недопустимое использование сетевых ресурсов. Наконец, политика безопасности окажет существенную помощь при выборе техники или программного обеспечения, поскольку из нее прямо следуют требования к закупаемому оборудованию и программному обеспечению по части защиты информации.

В-третьих, необходимо произвести учет всех информационных активов. Что может к ним относиться, приблизительно перечислено в параграфе 4.2. Необходимо выделить основные и вспомогательные сервисы, сформулировать требования по защите информации и каждого сервиса в отдельности. Несомненно, в первую очередь требования выдвигаются к основным сервисам.

В-четвертых, производится учет и идентификация угроз. При этом необходимо рассматривать не только внешние угрозы, но так же и внутренние. Очень часто большое внимание уделяется защите от внешнего вторжения, в то время как предельно мало – резервному копированию или защите от ошибок персонала. Напомним, что под защитой информации необходимо и важно понимать не только конфиденциальность, но так же целостность и доступность. Чем полнее будет перечень угроз (в пределах разумного, конечно), тем полнее можно рассмотреть вопрос о безопасности. После этого определяются возможности сервисов по части защиты информации.

Уже затем, если основные сервисы не способны обеспечить какой-либо функциональности в части защиты (и не только), рассматриваются вопросы о необходимости вспомогательных сервисов. При этом необходимо соблюдать принципы минимизации сервисов, разделения различных сервисов. Возможно, придется рассмотреть вопрос о приобретении дополнительного оборудования или программного обеспечения (например, для системы аутентификации с помощью токенов).

При рассмотрении защиты с помощью дополнительных экранирующих сервисов, необходимо обратить внимание на защиту самих экранирующих сервисов. Недостаточное внимание к этому вопросу способно свести на нет все усилия по защите информации.

В-пятых, всегда необходимо каким-либо образом отслеживать эффективность принимаемых мер по защите. Это поможет правильно распределять ресурсы, выделяемые на решение задач защиты. В противном случае, есть опасность существенных затрат в тех направлениях, которые не являются особенно критическими и опасными.

Наконец, еще раз подчеркнем тот факт, что защита информации и информационная безопасность вообще – это не разовое действие, а постоянная работа. Изменения в современном мире информационных технологий происходят слишком быстро, чтобы можно было в какой-то момент прекратить работу по защите информации, посчитав, что достигнут

необходимый уровень. Сам по себе этот уровень сохраняться не будет. Необходимо постоянно его проверять и поддерживать. Происходят установки новых программ, обнаруживаются ошибки в уже существующих и используемых, даются расширенные привилегии отдельным пользователям – и от былой защиты не остается и следа.

В мире вопросы информационной безопасности рассматриваются давно и плотно. В России до недавнего времени интерес к безопасности проявляли только банковские и государственные структуры. Приятно отметить, что в последнее время ситуация в этом вопросе существенно меняется. Многие фирмы и организации начинают понимать важность сохранности информации. Именно сохранности, поскольку многие приходят к пониманию информационной безопасности именно с точки зрения сохранности (что в наших терминах означает скорее целостность и частично - доступность).

Остается лишь порекомендовать руководителям организаций поскорее, не дожидаясь возможных убытков от потери информации, обратить внимание на вопросы информационной безопасности.

Список литературы

1. Гостехкомиссия России. Руководящий документ. Концепция защиты СВТ и АС от НСД к информации. - Москва, 1992.
2. Гостехкомиссия России. Руководящий документ. Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от НСД к информации. - Москва, 1992.
3. Гостехкомиссия России. Руководящий документ. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации. - Москва, 1992.
4. Гостехкомиссия России. Руководящий документ. Временное положение по организации разработки, изготовления и эксплуатации программных и технических средств защиты информации от НСД в автоматизированных системах и средствах вычислительной техники. - Москва, 1992.
5. Гостехкомиссия России. Руководящий документ. Защита от несанкционированного доступа к информации. Термины и определения. – Москва, 1992.
6. Department of Defense Trusted Computer System Evaluation Criteria. – DoD 5200.28-STD, 1993.
7. Information Technology Security Evaluation Criteria (ITSEC). Harmonized Criteria of France - Germany – the Netherlands - the United Kingdom. - Department of Trade and Industry, London, 1991.
8. National Computer Security Center. Trusted Network Interpretation. - NCSC-TG-005, 1987.
9. Security Architecture for Open Systems Interconnection for CCITT Applications. Recommendation X.800. - CCITT, Geneva, 1991.
10. Site security handbook, RFC2196, SEI/CMU, Sep 1997
11. Specifications for Guideline for The Analysis Local Area Network Security. - Federal Information Processing Standards Publication 191, 1994.
12. M. Fites, P. Kratz, and A. Brebner, "Control and Security of Computer Information Systems", Computer Science Press, 1989.
13. C. Pfleeger, "Security in Computing", Prentice-Hall, Englewood Cliffs, NJ, 1989.
14. An Introduction to Computer Security: The NIST Handbook. Draft. - National Institute of Standards and Technology, Technology Administration, U.S. Department of Commerce, 1994.
15. Гайкович В., Першин А. Безопасность электронных банковских систем. - Москва, "Единая Европа", 1994.
16. Bassham L.E., Polk W.T. Threat Assessment of Malicious Code and Human Threats (NISTIR 4939). – National Institute of Standards and Technology, Computer Security Division, 1992.
17. Stang D.J., Moon S. Network Security Secrets. - IDG Books Worldwide Inc., 1993.
18. Галатенко В. Информационная безопасность – обзор основных положений. Jet Infosystems, Москва – 1996г.
19. Левин В.К. Защита информации в информационно-вычислительных системах и сетях. - "Программирование", 5, 1994, с. 5-16.
20. Робачевский А.М. Операционная система UNIX – СПб, 1997.
21. Goodheart, B. & Cox, J. The Magic Garden Explained: The Internals of Unix System V Release 4, Prentice Hall

22. Vahalia, Uresh. Unix Internals: The New Frontiers, Prentice-Hall
23. Stevens, Richard W. Advanced Programming in the Unix Environment, Addison-Wesley
24. Колонцов В. Найти, проверить и обезвредить. Открытые системы, 1996г. №6.
25. Product Support Document (PSD) for Sun Sendmail, SunService, 1997
26. Setting up and debugging logging to remote hosts, SunService, 1996
27. SunService Tip Sheet for SUN NIS+, SunService, 1996
28. Getting started with Firewall-1, SunSoft, 1997
29. Solstice Firewall-1 Architecture and Administration, SunSoft, 1997

Приложение 1. Архитектурная модель безопасности ИС и сетей на базе UNIX

№ уровня	Название	Функциональное описание
7	Политика безопасности	Общие правила обеспечения безопасности предприятия
6	Персонал	Люди, использующие оборудование и данные (как правило, работники компании)
5	Локальная сеть	Компьютерное оборудование и внутренние линии связи: этажные коммутационные щиты, коммутационные комнаты, концентраторы и/или коммутаторы, маршрутизаторы т.д.
4	Сетевые сервисы	Сервисы баз данных, электронной почты, сетевых имен, файловые, печати, аутентификации, журналирования и т.д.
3	Брандмауэр	Firewall, прокси-серверы и т.д. – устройства, обеспечивающие защиту на уровнях 7,6,5,4 модели OSI
2	Пакетный фильтр	Маршрутизатор или иное устройство, работающее на уровнях 3,2,1 модели OSI
1	Внешние каналы связи	Каналы связи и оборудование, используемое для подключения к глобальным или иным сетям (модемы, выделенные или телефонные линии и т.д.)

Приложение 2. Пример политики безопасности

Описание аспекта. Информация, циркулирующая в рамках локальной сети, является критически важной. Локальная сеть позволяет пользователям разделять программы и данные; это увеличивает риск. Следовательно, каждый из компьютеров, входящих в сеть, нуждается в более сильной защите, чем отдельная машина. Эти повышенные меры безопасности и являются предметом данного документа.

Документ преследует две главные цели - продемонстрировать сотрудникам XYZ важность защиты сетевой среды и описать их роль в обеспечении безопасности, а также распределить конкретные обязанности по защите информации, циркулирующей в сети, равно как и самой сети.

Область применения. В сферу действия данной политики попадают все аппаратные, программные и информационные ресурсы, входящие в локальную сеть предприятия. Политика ориентирована также на людей, работающих с сетью, в том числе на пользователей, субподрядчиков и поставщиков.

Позиция организации. Целью организации XYZ является обеспечение целостности, доступности и конфиденциальности данных, а также их полноты и актуальности. Более частными целями являются:

- Обеспечение уровня безопасности, соответствующего нормативным документам.
- Следование экономической целесообразности в выборе защитных мер (расходы на защиту не должны превосходить предполагаемый ущерб от нарушения информационной безопасности).
- Обеспечение безопасности в каждой функциональной области локальной сети.
- Обеспечение подотчетности всех действий пользователей с информацией и ресурсами.
- Обеспечение анализа регистрационной информации.
- Предоставление пользователям достаточной информации для сознательного поддержания режима безопасности.
- Выработка планов восстановления после аварий и иных критических ситуаций для всех функциональных областей с целью обеспечения непрерывности работы сети.
- Обеспечение соответствия с имеющимися законами и общеорганизационной политикой безопасности.

Роли и обязанности (общие положения). Следующие группы людей отвечают за реализацию сформулированных выше целей. Детально их обязанности будут описаны ниже.

- Руководители подразделений. Они отвечают за доведение положений политики безопасности до пользователей и за контакты с пользователями.
- Администраторы локальной сети. Они обеспечивают непрерывное функционирование сети и отвечают за реализацию технических мер, необходимых для проведения в жизнь политики безопасности.
- Администраторы сервисов. Они отвечают за конкретные сервисы и, в частности, за то, что их защита построена в соответствии с общей политикой безопасности.

- Пользователи. Они обязаны использовать локальную сеть в соответствии с политикой безопасности, подчиняться распоряжениям лиц, отвечающих за отдельные аспекты безопасности, ставить в известность руководство обо всех подозрительных ситуациях.

Законопослушность. Нарушение политики безопасности может подвергнуть локальную сеть и циркулирующую в ней информацию недопустимому риску. Случаи нарушения со стороны персонала будут рассматриваться руководством для принятия мер вплоть до увольнения.

Роли и обязанности (детальное изложение).

Руководители подразделений обязаны:

- Постоянно держать в поле зрения вопросы безопасности. Следить за тем, чтобы то же делали их подчиненные.
- Проводить анализ рисков, выявляя активы, требующие защиты, и уязвимые места систем, оценивая размер возможного ущерба от нарушения режима безопасности и выбирая эффективные средства защиты.
- Организовать обучение персонала мерам безопасности. Обратить особое внимание на вопросы, связанные с антивирусным контролем.
- Информировать администраторов локальной сети и администраторов сервисов об изменении статуса каждого из подчиненных (переход на другую работу, увольнение и т.п.).
- Обеспечить, чтобы каждый компьютер в их подразделениях имел хозяина или системного администратора, отвечающего за его безопасность и имеющего достаточную квалификацию для выполнения этой роли.

Администраторы локальной сети обязаны:

- Информировать руководство об эффективности существующей политики безопасности и о технических мерах, которые могут улучшить защиту.
- Обеспечить защиту оборудования локальной сети, в том числе интерфейсов с другими сетями.
- Оперативно и эффективно реагировать на события, таящие угрозу. Информировать администраторов сервисов о попытках нарушения защиты. Оказывать помощь в отражении угрозы, выявлении нарушителей и предоставлении информации для их наказания.
- Использовать проверенные средства аудита и обнаружения подозрительных ситуаций.
- Ежедневно анализировать регистрационную информацию, относящуюся к сети в целом и к файловым серверам в особенности.
- Следить за новинками в области информационной безопасности, информировать о них пользователей и руководство.
- Не злоупотреблять данными им большими полномочиями. Пользователи имеют право на тайну.

- Разработать процедуры и подготовить инструкции для защиты локальной сети от зловредного программного обеспечения. Оказывать помощь в обнаружении и ликвидации зловредного кода.
- Регулярно выполнять резервное копирование информации, хранящейся на файловых серверах.
- Выполнять все изменения сетевой аппаратно-программной конфигурации.
- Гарантировать обязательность процедуры идентификации и аутентификации для доступа к сетевым ресурсам.
- Выделять пользователям входные имена и начальные пароли только после заполнения регистрационных форм.
- Периодически производить проверку надежности защиты локальной сети. Не допускать получения привилегий неавторизованными пользователями.

Администраторы сервисов обязаны:

- Управлять правами доступа пользователей к обслуживаемым объектам.
- Оперативно и эффективно реагировать на события, таящие угрозу. Информировать администраторов локальной сети о попытках нарушения защиты. Оказывать помощь в отражении угрозы, выявлении нарушителей и предоставлении информации для их наказания.
- Регулярно выполнять резервное копирование информации, обрабатываемой сервисом.
- Выделять пользователям входные имена и начальные пароли только после заполнения регистрационных форм.
- Ежедневно анализировать регистрационную информацию, относящуюся к сервису.
- Регулярно контролировать сервис на предмет зловредного программного обеспечения.
- Периодически производить проверку надежности защиты сервиса. Не допускать получения привилегий неавторизованными пользователями.

Пользователи обязаны:

- Знать и соблюдать законы, правила, принятые в XYZ, политику безопасности, процедуры безопасности.
- Использовать доступные защитные механизмы для обеспечения конфиденциальности и целостности своей информации.
- Использовать механизм защиты файлов и должным образом задавать права доступа.
- Выбирать хорошие пароли, регулярно менять их. Не записывать пароли на бумаге, не сообщать их другим лицам.
- Помогать другим пользователям соблюдать меры безопасности. Указывать им на замеченные упущения с их стороны.
- Информировать администраторов или руководство о нарушениях безопасности и иных подозрительных ситуациях.
- Не использовать слабости в защите сервисов и локальной сети в целом.
- Не совершать неавторизованной работы с данными, не создавать помех другим пользователям.

- Всегда сообщать корректную идентификационную и аутентификационную информацию, не пытаться работать от имени других пользователей.
- Обеспечивать резервное копирование информации с жесткого диска своего компьютера.
- Знать принципы работы зловредного программного обеспечения, пути его проникновения и распространения, слабости, которые при этом могут использоваться.
- Знать и соблюдать процедуры для предупреждения проникновения зловредного кода, для его обнаружения и уничтожения.
- Знать слабости, которые используются для неавторизованного доступа.
- Знать способы выявления ненормального поведения конкретных систем, последовательность дальнейших действий, точки контакта с ответственными лицами.
- Знать и соблюдать правила поведения в экстренных ситуациях, последовательность действий при ликвидации последствий аварий.

Приложение 3. Обобщенная методика обеспечения безопасности

Для того чтобы обеспечить требуемый уровень безопасности информации в системе, необходимо предпринять следующие шаги:

I. Разработать Политику безопасности верхнего уровня.

Политика безопасности – это формальное выражение правил, согласно которым обязан действовать персонал, имеющий доступ к информационным ресурсам организации.

Формированием Политики безопасности должен заниматься специально созданный временный комитет, в состав которого входят:

1. Администратора безопасности организации и/или сети
2. Специалистов по информационным технологиям (то есть специалистов отдела АСУ и/или информатизации)
3. Административных работников, в подчинении которых находится большое число пользователей (например, начальники отделов)
4. Группа реагирования на случаи нарушения режима безопасности
5. Представителей групп пользователей, которых в дальнейшем будет касаться действие Политики безопасности
6. Руководителей предприятия или организации
7. Юридических консультантов

Политика безопасности верхнего уровня формулируется в виде документа, в общем виде описывающего цели и задачи мероприятий по защите информации, а так же обязанности и ответственность персонала, работающего с ценными данными.

Для крупных предприятий имеет смысл детальная разработка документов по безопасности данных, которые могут включать в себя:

1. Руководство по закупке компьютерного и информационного оборудования, которое определяет требуемые или желательные средства обеспечения безопасности, которыми должно обладать закупаемое оборудование или программное обеспечение.
2. Правила безопасности, которые определяют разумное ограничение прав пользователей на тайну. Например, регистрация и анализ команд, подаваемых пользователем, учет за электронной почты, учет доступа к файлам и т.д.
3. Правила доступа, которые определяют ограничение права доступа и привилегий пользователей, для защиты данных и оборудования от потерь, повреждения или уничтожения. А так же определение допустимого использования сетевых ресурсов и сервисов для пользователей и обслуживающего персонала. Так же в этом разделе должны содержаться правила, описывающие внешние соединения и их использование, обмен данными, подключение новых устройств и ресурсов к сети, а так же установку нового программного обеспечения. Здесь же желательно специфицировать уведомления (например, сообщение при соединении с удаленной машиной должно предупреждать о необходимости авторизованного использования и возможности наблюдения за линией, а не просто содержать строчку “Welcome to”).

4. Правила Учета – определяют ответственность пользователей, администраторов и руководящего персонала. В данном документе необходимо определить возможности учета и аудита, и обеспечивать руководство по предотвращению случаев нарушение режима безопасности (например, с кем следует связаться и взаимодействовать в случае подозрений на попытку нарушения режима безопасности)
5. Правила аутентификации. Эти правила устанавливают степень доверия методам аутентификации, путем, например, предъявления требований к процедуре установления пароля и самому паролю. Кроме того, здесь же необходимо описать правила доступа и использования устройств аутентификации (например, устройств для генерации одноразовых паролей).
6. Условия доступности. Определяют тот уровень доступности сетевых ресурсов, который пользователи вправе ожидать. Здесь необходимо определить такие моменты, как резервирование, восстановление после возможных сбоев, а так же часы гарантированной работы оборудования и возможные перерывы на тех. обслуживание – их расписание и продолжительность.
7. Правила обслуживания сети и информационной системы в целом. Описывают в какой степени собственный и внешний обслуживающий персонал имеет доступ к данным и технологиям, обеспечивающим работу предприятия. Особенно важно четко определить допустимо ли удаленное управление сетевыми ресурсами внешним персоналом, и каким образом отслеживается тот уровень доступа, который имеют люди, не состоящие в организации, но выполняющие те или иные работы по обслуживанию.
8. Правила сообщения о нарушениях. Эти правила описывают, о каких именно типах нарушений правил безопасности следует сообщать и кто должен это делать. Отметим, что в условиях невысокого риска и возможности анонимного сообщения о замеченных нарушениях работниками предприятия, существует высокая вероятность того, что о замеченных нарушениях будет сообщено сразу же.
9. Информация о поддержке. В этом разделе политики безопасности описывается – с кем следует связаться в случае нарушения политики безопасности, каким образом определить такую попытку, какую информацию следует считать конфиденциальной или частной, а так же приводятся ссылки на соответствующие процедуры безопасности, пункты правил безопасности предприятия и, быть может, законы, относящиеся к этой области.

Политика безопасности утверждается руководством предприятия как руководящий документ. Каждый пользователь должен быть ознакомлен с Политикой безопасности. Желательно так же, произвести регистрацию того, что пользователи ознакомились и согласны поддерживать Политику безопасности, в виде росписи в специальном журнале.

II. Произвести учет активов информационной сети предприятия

Учет активов необходим для того, чтобы иметь полное и точное представление о том, что же именно мы хотим защищать и какими возможностями располагаем. Учет активов производится по нескольким статьям:

1. Оборудование: процессоры, платы, клавиатуры, терминалы, рабочие станции, персональные компьютеры, принтеры, дисковые накопители, линии связи, терминальные серверы, маршрутизаторы и т.д.

2. Программное обеспечение: утилиты, диагностические программы, операционные системы, прикладное программное обеспечение и т.д.
3. Данные: находящиеся в реальном использовании, хранимые в режиме on-line, данные в архивах (ленты и CD-ROM), системные журналы, базы данных, транзитные данные, проходящие по линиям коммуникации и т.д.
4. Персонал: пользователи, администраторы сетей, сервисов и серверов.
5. Документация: относящаяся к программам, оборудованию, системам, местные административные процедуры.
6. Носители информации: бумага, бланки и документы, магнитные носители и т.д.

III. Составить список угроз безопасности данных

Цель этого этапа – прояснить, что именно представляет опасность для данных, циркулирующих и хранящихся в сети предприятия. Перечень угроз может сильно разниться для разных предприятий и организаций. В общем случае он включает в себя:

1. Ошибки пользователей
2. Кражи оборудования
3. Аварии и низкое качество электропитания
4. Возможность пожара
5. Возможность неавторизованного доступа к данным
6. Отказ оборудования
7. Опасность доступа к сети предприятия постороннего оборудования
8. и т.д.

Перечень возможных угроз может быть очень велик. В общем случае, чем тщательнее анализируются угрозы, тем более эффективную защиту можно создать.

При учете угроз необходимо не просто перечислить угрозы, но сразу соотнести их с активами. Например, угроза отказа жесткого диска присуща файловым серверам и серверам приложений. Угроза повреждения кабеля присуща кабельной системе и не относится к серверам.

В конечном итоге, каждому элементу или группе элементов из списка активов соотносится определенный список угроз.

IV. Оценить вероятность и возможный ущерб активам от соответствующих угроз

На этом этапе необходимо в денежном или ином выражении (например, по 5-ти балльной шкале) оценить вероятность той или иной угрозы и ущерба от нее. Оценка вероятности и ущерба – особая проблема, поскольку часто невозможно достаточно точно оценить ни вероятность события, ни ущерб. В этом случае не только допустимо, но и желательно пользоваться балльной оценкой, поскольку расчеты денежных потерь и точных значений вероятности не осуществимы на практике даже для средней по размерам организации, не говоря уже о крупных. В этом случае при принятии решений необходимо иметь ввиду приблизительность оценки и возможные ошибки.

V. Составить перечень рисков

В качестве значения риска для той или иной угрозы принимается произведение возможного ущерба на вероятность угрозы. После ранжирования списка по величине риска сразу можно выделить наиболее слабые места в защите информации.

После этапа составления перечня рисков результатом произведенной работы должен явиться список наиболее опасных угроз – то есть тех, для которых значение риска имеет наибольшее значение. При этом важно помнить приблизительность оценки, так что угрозы с немного меньшими значениями риска на деле могут быть опаснее тех, для которых значение риска больше. В случае небольших отличий, желательно произвести более точный анализ риска для этих угроз.

После выделения подмножества наиболее опасных угроз желательно произвести более точную денежную оценку их опасности. Это поможет в дальнейшем оценить действенность и стоимость мероприятий по снижению риска.

VI. Определить возможности по защите информации от наиболее опасных угроз

Как правило, существует несколько возможных способов снижения того или иного риска. Каждый из способов характеризуется стоимостью. Стоимость защитного механизма может состоять из стоимости закупки оборудования, стоимости программного обеспечения, обучения персонала, а так же стоимости уменьшения удобства работы с системой. Последнее, впрочем, весьма сложно оценить. Соотношение стоимости внедрения защитных мер и возможного ущерба, а так же соотношение между вероятностью той или иной угрозы до и после внедрения защитных мер должно служить оценкой эффективности выбранных защитных механизмов.

Важным обстоятельством является совместимость нового средства со сложившейся операционной и аппаратно-программной инфраструктурой. Меры безопасности, как правило, носят недружественный характер, что может отрицательно сказаться на энтузиазме работников. Порой сохранение духа открытости может оказаться важнее минимизации материальных потерь. Впрочем, такого рода ориентиры должны быть расставлены в Политике безопасности.

Можно представить себе ситуацию, когда для уменьшения риска не существует эффективных и приемлемых по цене мер. Например, компания, базирующаяся в сейсмически опасной зоне, не всегда может позволить себе строительство защищенной штаб-квартиры. В таком случае приходится поднимать планку приемлемого риска и переносить центр тяжести на смягчение последствий и выработку планов восстановления после аварий, стихийных бедствий и иных происшествий. Продолжая пример с сейсмоопасностью, можно рекомендовать регулярное тиражирование данных в другой город и овладение средствами восстановления первичной базы данных.

VII. Составить план по внедрению дополнительных механизмов обеспечения безопасности

Внедрение любых дополнительных механизмов по обеспечению безопасности информации приведет к некоторому изменению в информационной структуре. Критически важно не потерять контроль таких изменений. Поэтому внедрение любых новых средств должно производиться по заранее разработанному плану. Кроме того, внедрение некоторых средств

может привести к временной неработоспособности сети, так что план должен учитывать время проведения работ по внедрению новых средств. Возможно, что такие работы придется производить в нерабочее время, когда возможен останов сети.

План внедрения новых средств обеспечения безопасности должен быть согласован с заинтересованными отделами и включен в общий план по проведению работ по обслуживанию и администрированию сети.

VIII. Произвести работы по внедрению новых средств обеспечения безопасности

Этот этап подразумевает выполнение работ по установке/настройке нового программного обеспечения или оборудования согласно плану, разработанному на предыдущем шаге.

IX. Оценить эффективность принятых мер

После внедрения новых средств обеспечения безопасности необходимо соотнести понесенные затраты со степенью уменьшения соответствующего риска. На этом же этапе производится пересмотр списка угроз и рисков, составленного при выполнении шага V, с учетом влияния новых средств обеспечения безопасности.

X. Повторить весь путь с шага II

Цель этого этапа – показать, что работа по обеспечению безопасности не является разовой, а должна выполняться постоянно. Например, внедрение новых средств обеспечения безопасности приводит к изменениям информационных активов. А именно – появляется новое оборудование или программное обеспечение. Соответственно, необходимо заново пройти весь путь по обеспечению безопасности уже изменившейся сетевой конфигурации. Конечно, выполнить эту работу будет уже не столь сложно как в первый раз, поскольку большая часть угроз и рисков уже оценена, и необходимо лишь внести изменения в оценки согласно произошедшим изменениям.

Приложение 4. Руководство пользователя системы «SystemGuard»

Данная система предназначена для использования администраторами серверов на базе ОС Linux.

Для экономии дискового пространства модифицированное ядро поставляется в виде файла отличий исходных текстов модифицированного ядра Linux версии 2.0.35 от стандартного ядра Linux той же версии. Исходные тексты ядра Linux могут быть получены бесплатно по FTP со следующих адресов:

- ftp://linux.kernel.org/pub/linux/kernel/v2.0/linux-2.0.35.tar.gz
- ftp://ftp.ru.kernel.org/pub/linux/kernel/v2.0/linux-2.0.35.tar.gz

Для модификации ядра необходимо сделать следующее:

1) Получите исходные тексты ядра Linux 2.0.35 и положите архив в директорию /usr/src/
2) Зарегистрируйтесь в системе как суперпользователь
3) Находясь в директории /usr/src/, выполните команду «tar –xzf linux-2.0.35.tar.gz», и подождите завершения распаковки архива
4) Удалите символическую ссылку /usr/src/linux/, если она существует
5) В той же директории введите команду «ln –s linux-2.0.35 linux»
6) Скопируйте файл systemguard-2.0.35.diff в каталог /usr/src/
7) В том же каталоге выполните команду «patch –p0 < systemguard-2.0.35.diff», и подождите ее исполнения
8) Модификация ядра закончена. Конфигурируйте, компилируйте и устанавливайте новое ядро, как обычно. При конфигурации появится новое подменю «Security options».

Вместе с системой защиты поставляется также исходный код утилиты sse, просматривающей и изменяющей состояние флага HF_STACKEXEC, контролирующего исполнение кода в стеке. Утилита компилируется компилятором gcc версии 2.7.2 или старше.

Использовать утилиту следует так:

sse –v <имя файла> - выдать состояние флага защиты стека

sse –e <имя файла> - запретить исполнение кода в стеке

sse –d <имя файла> - разрешить исполнение кода в стеке

Приложение 5. Руководство программиста системы «SystemGuard»

Для экономии места система поставляется в виде diff-файла. Данный diff-файл изменяет следующие файлы исходного текста ядра ОС Linux (имена приводятся относительно пути /usr/src/linux/):

/Documentation/Configure.help – добавлена секция интерактивной помощи по новым опциям конфигурации

/arch/i386/defconfig – добавлены значения по умолчанию для новых опций

/arch/i386/config.in – добавлены опции

/arch/i386/head.S – модифицирована GDT (добавлен 1 дескриптор сегмента кода и 1 дескриптор сегмента стека)

/arch/i386/kernel/ptrace.c – модифицирован код, использующий номера дескрипторов GDT в функции putreg(…)

/arch/i386/kernel/signal.c - модифицирован код, использующий номера дескрипторов GDT, процедуры обработки сигналов (функции setup_frame(…), sys_sigreturn(…))

/arch/i386/kernel/traps.c – модифицирован код, обрабатывающий ошибки защиты (функция do_general_protection(…). Добавлено 2 переменных:

- retaddr (длинное целое без знака) – реальный адрес возврата при подготовке стека обработчика сигнала
- regnum (байт) – номер регистра, в котором хранится адрес возврата для эмуляции трамплайна

)

/arch/i386/math-emu/fpu_entry.c - модифицирован код, использующий имена сегментов в функции math_emulate(…)

/arch/i386/mm/fault.c – модифицирована функция do_page_fault (…), использующая номера дескрипторов GDT

/fs/binfmt_aout.c – добавлена поддержка флага F_STACKEXEC

/fs/binfmt_elf.c – добавлена поддержка флага EF_STACKEXEC

/fs/exec.c – добавлена установка флага защиты стека процесса в соответствии с флагом защиты стека заголовка исполнимого файла

/fs/namei.c – модифицированы функции follow_link(…), open_namei(…), do_link(…).

/include/asm-i386/processor.h – дополнительно подключены 2 файла (/include/linux/binfmts.h и /include/linux/config.h), изменена инлайн-функция MMAP_SEARCH_START для перемещения адреса размещения разделяемых библиотек, модифицирована инлайн-функция start_thread(…)

/include/asm-i386/segment.h - дополнительно подключен файл /include/linux/config.h, определены константы USER_HUGE_CS, USER_HUGE_SS, MAGIC_SIGRETURN, MAGIC_TRAMP

/include/linux/a.out.h – задано значение флага F_STACKEXEC

/include/linux/elf.h - задано значение флага EF_STACKEXEC

/include/linux/kernel.h – добавлена инлайн-функция security_alert(…) для записи сообщений о попытках нарушения политики безопасности, используя сервис syslog.

/include/linux/sched.h – добавлена переменная trampoline_retaddr (длинное беззнаковое целое) – адрес возврата из трамплайна, константы

- PF_STACEXEC_F - принудительное исполнение стека
- PF_STACEXEC_C – стек сейчас исполняем
- PF_STACEXEC_M - маска флагов защиты стека

и создает файлы

security/Config.in – определение подменю «Security options»

security/Common.in – опции защиты, общие для всех архитектур

Приложение 6. Список файлов на дискете

/systemguard-2.0.35.diff – система «SystemGuard» для ядра ОС Linux версии 2.0.35
/sse.c – исходный код утилиты SSE (Set Stack Executability)