Гриняев С.Н.

Интеллектуальное противодействие информационному оружию

Средства и методы борьбы с алгоритмическим информационным оружием с использованием методов искусственного интеллекта

1999

Введение

Сейчас много говорят о том, что на современном этапе развития общества многие традиционные ресурсы человеческого прогресса постепенно утрачивают свое первоначальное значение, вместе с этим, все большее значение приобретает информация. Бесспорно, информация становится сегодня главным ресурсом научно-технического и социально-экономического развития мирового сообщества. Наверное, уже можно говорить о новом витке в развитии общественной формации - информационном обществе. В настоящее время хорошо налаженная распределенная сеть информационно-вычислительных комплексов способна сыграть такую же роль в общественной жизни, какую в свое время сыграли электрификация, телефонизация, радио и телевидение вместе взятые. Свидетелем этому человечество стало на примере развития глобальной сети *Internet*.

Уже сегодня, по заявлениям некоторых иностранных экспертов, отключение компьютерных систем приведет к разорению 20% средних компаний в течение нескольких часов, 48% потерпят крах в течение нескольких суток, остальные будут разорены в этом промежутке. Около 33% банков будут разорены спустя несколько часов после такой катастрофы, а 50% - всего через несколько суток [74].

Переход информации в разряд важнейших ресурсов человечества вызывает к жизни проблему борьбы за обладание этим ресурсом, и, как следствие, появление принципиально нового средства нападения и защиты – информационного оружия.

С конца 80-х годов в США неоднократно проводились исследования по изучению уязвимости автоматизированных систем управления (АСУ) различного назначения с точки зрения информационной безопасности.

По результатам проведенных исследований был опубликован ряд отчетных материалов, о которых можно узнать из [16, 37,104]. Согласно им по проблеме безопасности автоматизированных систем экспертами были сделаны следующие выводы.

В настоящее время в США функционируют 2,1 млн. электронно-вычислительных машин и 10000 локальных вычислительных сетей (ЛВС). Компьютеры и ЛВС имеют выход более чем в 100 международных информационных сетей. В сети *Internet* действуют 150000 персональных ЭВМ (ПЭВМ), принадлежащих Министерству Обороны США.

95% военных линий связи проходят по телефонным каналам связи общего пользования. 30% телекоммуникационных сетей, предназначенных для использования в случае военных действий, относятся к категории “засекреченных”.

По итогам 1995 года в США зафиксировано 250000 случаев вторжения в АСУ государственного назначения (кроме АСУ военного назначения). 162500 (65%) таких вторжений оказались успешными. Зафиксировано свыше 160000 вторжений в АСУ МО США. Количество вторжений в АСУ государственного и военного назначения каждый год увеличивается, в среднем, в два раза.

Возросло количество случаев намеренного внедрения компьютерных вирусов в АСУ государственного и военного назначения: с 583 в 1995 г. до 896 в 1996 г.

В 1996 году в США зафиксированы рост случаев несанкционированного доступа (НСД) в федеральных ведомствах и увеличение финансовых потерь. Средний ущерб от одного компьютерного преступления в США составляет 450 тыс. долларов. Ежегодные потери некоторых фирм США достигают 5 млрд. долларов.

Суммарный ущерб от компьютерных преступлений в странах Западной Европы за 1996 год - порядка 30 млрд. долларов.

Наиболее часто используемый канал, по которому осуществляются НСД,- сеть *Internet* (65% случаев). По результатам тестирования установлено, что 2/3 коммерческих и государственных узлов *Internet* не защищены от вторжения хакеров. За январь 1997 года зафиксировано три случая вторжения хакеров на государственные узлы особой важности, принадлежащие Министерству Юстиции, Центральному разведывательному управлению и Военно-воздушным силам США.

Результаты тестирования в 1995-1996 г.г. Министерством Обороны США 8932 АСУ военного назначения с применением средств проникновения, используемых хакерами, показали, что в 7860 (88%) случаях попытки проникновения обнаружены не были.

Проникновение в АСУ воздушным движением с использованием программы-вируса длиной 15-20 строк позволяет полностью парализовать крупный аэропорт за 10-15 минут.

Аналогичные исследования, проведенные в других странах (Великобритании, Франции, Швейцарии, Японии и др.), показали, что АСУ государственного и военного назначения в этих странах являются еще менее защищенными от информационных воздействий. Потенциальную возможность применения против этих АСУ информационного оружия подтверждают, например, такие факты: концепции развития АСУ и противодействия угрозе информационного нападения Японии базируются на соответствующих концепциях США; правительство Франции планирует активное использование сети *Internet* в государственных структурах.

В последние годы в США и в отдельных государствах Европы (Франции, Великобритании, Швейцарии) активно осуществляется реализация концепций комплексной защиты информационных инфраструктур государств, основой которых являются АСУ. В США на период 1997-2000 г.г. запланировано затратить 1,6 млрд. долларов на усиление защиты национальной информационной инфраструктуры [37]. Министерством Обороны США предложено выделить на эти цели 3 млрд. долларов дополнительно.

В то же время, повсеместно признано, что в настоящее время эффективность наступательных средств информационной войны - информационного оружия превосходит эффективность систем защиты информации АСУ.

Вышесказанное свидетельствует о том, что сейчас, как никогда актуальна проблема информационного вторжения с применением информационного оружия, которое, по сути, представляет собой средства осуществления несанкционированного доступа (программные, аппаратные, алгоритмические и др.) к информационным ресурсам противника. Выход этой угрозы на первый план связан с тем, что современные системы управления являются системами критических приложений с высоким уровнем компьютеризации. Они могут оказаться весьма уязвимыми с точки зрения воздействия информационного оружия, как в военное, так и в мирное время. Последнее может привести к тому, что к угрожаемому периоду оружие сдерживания страны за счет скрытого внедрения в программное обеспечение систем управления им программных закладок окажется полностью или частично заблокированным. О реальности этого утверждения свидетельствует опыт войны в Персидском заливе. Ирак практически не смог применить закупленные во Франции системы ПВО, потому что их программное обеспечение содержало логические бомбы, которые были активизированы с началом боевых действий.

Разнообразие информационного оружия, форм и способов его воздействия, особенности появления и применения породили сложнейшие задачи защиты от него. Иностранные специалисты первые поняли и оценили значение информационного оружия. Поэтому они стремятся захватить лидирующие позиции в решении этих задач. Американские военные считают, что преимущество в информационном оружии должно упрочить мировое лидерство США в следующем веке. Этим объясняется большой интерес и активность американцев в исследовании проблем информационной войны. Все сказанное подтверждается докладами и дискуссиями на 5$^{-й}$ Международной конференции по информационной войне, большую часть участников которой составляли сотрудники государственных учреждений, армии и разведывательного сообщества США - АБН, ЦРУ, ФБР [37].

Информационное оружие стало одной из составляющих военного потенциала США, позволяющей выигрывать малые войны и разрешать военные конфликты без применения обычных вооруженных сил. В войсках создаются специальные подразделения и структуры управления для ведения информационной войны. Во всех военных учебных заведениях США введены специальные курсы по информационным войнам и налажен выпуск офицеров по специальности этого профиля.

Это не может не повлиять на то, что при разработке перспективных компьютерных систем и сетей обмена информацией (СОИ) еще большее внимание должно быть уделено вопросам защиты информации и не только от пресловутых компьютерных вирусов, но и от грозного информационного оружия. При этом ***сегодня такая защита уже не может и не должна оставаться пассивной, ей необходимо придать форму активного информационного противодействия противнику.***

Предложенные в настоящей работе методы построения систем защиты информации собственно и предназначены для создания таких систем. Мы исходили из того факта, что любая СОИ должна быть приспособлена к отражению возможных информационных ударов не только в чрезвычайных ситуациях военного времени, но и в относительно спокойное мирное время.

Сейчас многие крупные корпорации делают ставки в успехе своего бизнеса на развитие широкой инфраструктуры сбора, обработки и хранения информации. Безусловно, в век информатизации это позволяет получить значительное преимущество над конкурентами. Вместе с тем, широкая и разветвленная телекоммуникационная сеть с выходом на открытые телекоммуникации типа *Internet* является потенциальной угрозой для всей корпорации при отсутствии достаточных мер безопасности информации. К огромному сожалению, большинство существующих средств и методов защиты не позволяют достичь требуемого уровня безопасности. Можно привести массу примеров, по которым несложно сделать вывод о том, что это действительно так. В настоящее время тратятся огромные средства на, так называемую «сертификацию» программных продуктов и аппаратуры на предмет наличия в них различного рода шпионских «закладок».

Однако, можно с уверенностью сказать, что ввиду огромной сложности современных программных и аппаратных средств, несовершенства методов анализа, применяемых для сертификации, могут быть проверены на наличие «закладок» за приемлемое время лишь очень немногие продукты. Это приводит к тому, что с повышением категории секретности информации и ужесточением требований по сертификации приходится работать на допотопной технике и устаревших программах, так как на сертификацию современных продуктов, меняющихся на рынке с огромной скоростью, просто не хватает средств.

Описанные в работе методы построения систем защиты информации в информационно-вычислительных сетях корпоративного масштаба позволяют частично отказаться от сертификации, предположив, что все информационные процессы в сети потенциально опасны.

Информационное воздействие на ресурсы информационно-вычислительной сети (ИВС), как основы информационной инфраструктуры практически любой корпорации, можно охарактеризовать увеличением загрузки сети при осуществлении противником информационных атак и вероятностью их успешного осуществления. Последняя характеристика прямо зависит от времени бесконтрольного присутствия противника в ИВС, то есть от времени, когда системе безопасности неизвестна стратегия атаки или она не можем влиять на ее результат.

Задача снижения вероятности успешных информационных атак противника до настоящего времени решалась в большей части традиционными методами обеспечения безопасности информации (ОБИ). При этом было ярко выражено стремление, перекрыть все возможные каналы несанкционированного доступа к данным, хранимым, обрабатываемым и передаваемым в сети. Основной недостаток такого подхода к ОБИ заключается в том, что противник знает результат своей атаки и в случае неудачи может выбрать такую стратегию атаки и (или) канал ее реализации, которая не будет обнаружена соответствующей системой обеспечения информационной безопасности сети. Этот недостаток теории становится все более существенным с увеличением масштабов и разнородности сети, спектра предоставляемых услуг, количества и качественного состава информационных ресурсов. Положение еще более усугубляется с ростом числа способов и изощренности информационных атак, которые уже направлены не только на данные, но и на многочисленные сетевые службы и могут осуществляться практически на всех уровнях эталонной модели взаимодействия открытых систем (ЭМВОС) [8, 18,19].

В связи с нарастающей опасностью информационных войн, которые, как мы уже отмечали, могут вестись и в мирное время, ИВС любой корпорации должна быть способна к ведению информационной борьбы, то есть к адекватному реагированию на информационные воздействия противника и решению других специфических задач информационной борьбы, направленных на достижение превосходства над противником в информационной войне. Это позволит сохранить свои информационные ресурсы, а значит сохранить свой бизнес и свою репутацию перед клиентами.

В рамках информационной борьбы, как и при любых конфликтных действиях, можно выделить оборонительную и наступательную составляющие [37].

Оборонительная составляющая информационной борьбы реализуется методами обеспечения безопасности, а наступательная может рассматриваться как воздействие на информационно-вычислительные сети корпораций через межсетевые соединения, активный поиск и обнаружение несанкционированных действий в ИВС и, наконец, информационное противодействие.

Информационное противодействие имеет целью снизить время бесконтрольного присутствия противника в сети и, в зависимости от целей информационной борьбы и задач, стоящих перед службой безопасности корпорации, осуществить дезинформацию, дезориентацию, а также другие действия, направленные на все виды ресурсов противника, включая информационные.

В качестве основных критериев эффективности информационного противодействия в работе выбрано время бесконтрольного присутствия противника в сети, а также нагрузка на сеть со стороны потоков пакетов, порождаемых противником.

В работе принят интеллектуальный подход к противодействию, заключающийся во всестороннем анализе сложившейся ситуации, выборе оптимального варианта реагирования и осуществляемый с использованием средств искусственного интеллекта. А так как существенной составляющей времени бесконтрольного присутствия противника в сети является время подготовки противодействия, то основное внимание в работе уделено разработке методов противодействия, минимизирующих этот показатель.

Подход к осуществлению интеллектуального противодействия основан на понятии ложного объекта атаки (ЛОА), имитирующего результат работы или процесс функционирования объекта атаки (ОА), выбранного противником. Отдавая приоритет интеллектуальному противодействию, в сравнении с простым реагированием, мы не исключаем из сети информационные потоки, порождаемые противником. Поэтому в целях снижения их влияния на сеть, необходимо выбрать в сети зону расположения ЛОА и перенаправить на него поток, порождаемый противником таким образом, чтобы снизить нагрузку на сеть.

К настоящему моменту в России накоплен значительный опыт по теории и практике обеспечения безопасности автоматизированных систем. Наряду с общеизвестными работами в этой области [25, 33, 34, 75], хотелось бы отметить работы Герасименко В.А. [19-22], Ухлинова Л.М. [78-80,39, 11], Зегжды П.Д.[36, 64,74], а также работы Платонова В.В., Петряева А.Б., Гаценко О.Ю., Максимова Ю.Н. [50-52], Воробьева А.А.. Анализу информационных атак на современные информационно-вычислительные сети посвящен ряд работ Центра защиты информации Санкт-Петербургского государственного технического университета [36, 64, 74].

Хотя в настоящее время появились работы, исследующие проблему информационной борьбы в компьютерных сетях, однако вопросы интеллектуального противодействия, имеющие целью ввести противника в заблуждение относительно объекта атаки, к сожалению не нашли достойного отражения. А ведь именно методы интеллектуального противодействия могут снизить время бесконтрольного присутствия противника в ИВС, а значит - сократить вероятность успешного осуществления им атак на сеть.

Как уже говорилось выше, целью нашей работы является разработка методов интеллектуального противодействия, минимизирующих время бесконтрольного присутствия противника в информационно-вычислительной сети некоторой корпорации и его влияние на пропускную способность сети.

Надеемся, что результаты нашей работы послужат толчком к активизации работ по применению методов искусственного интеллекта к решению задач создания действительно безопасных систем защиты для корпоративных сетей.

Глава 1. Информационная борьба

1.1. Основные понятия и определения информационной борьбы

1.1.1. Определение информационного ресурса

Определение1.1. Информацией называется свойство материи выступать в виде некоторого разнообразия (объектов, элементов, характеристик) и отражать это разнообразие, проявляющееся, в частности, в сведениях, извлекаемых людьми (машинами) в результате соответствующей обработки наблюдений окружающего мира, анализируемых ими и используемых в целях коммуникации (связи) и управления [69].

Если из этого определения убрать философский аспект, то определение информации, пригодное для практического использования формулируется следующим образом.

Определение 1.2. Информация - это сведения о лицах, предметах, фактах, событиях, явлениях и процессах, независимо от формы их представления [43].

При создании систем защиты информации необходимо, прежде всего, выяснить, какие единицы или объекты (информационные ресурсы) подлежат защите [44].

Что такое информационные ресурсы? В проекте закона РФ “Об информации, информатизации и защите информации” дано следующее определение информационных ресурсов [81, 26].

Определение1.3. Информационные ресурсы - отдельные документы и отдельные массивы документов, документы и массивы документов в информационных системах (библиотеках, архивах, фондах, банках данных, других информационных системах), являющиеся предметом отношений юридических и физических лиц, государства.

Применительно к автоматизированным системам основными единицами или объектами (информационными ресурсами) являются поля данных, записи в базах данных, файлы, программы, диски, дискеты, бобины с магнитными лентами и т.п.

1.1.2. Понятие информационной войны

Одна из реальных и относительно новых глобальных угроз безопасности компьютерных систем - это информационное оружие, проблемы разработки и использования которого уже стали высокоприоритетными в ряде западных стран и, прежде всего в США.

Многие системы управления становятся информационно-зависимыми. При этом даже незначительные случайные нарушения нормального функционирования компьютеров могут нанести существенный урон особо уязвимым системам управления и планирования.

Однако действительно серьезной проблемой становятся не случайные сбои компьютеров, а опасность специального и целенаправленного воздействия на информационные ресурсы противником.

Военные аналитики США сравнивают ущерб от нарушения функционирования вычислительных систем страны с последствиями применения ядерного оружия, так как поражение в информационной войне надолго отбрасывает “проигравшую” страну на обочину мировой истории. И наоборот, страны, добившиеся подавляющего преимущества в информационной области, смогут с достаточно высокой степенью вероятности моделировать поведение остальных стран, “заставлять” их делать определенные ходы. Одним словом - они получат неограниченные возможности управления побежденными странами, которым очень трудно “догонять”. Чтобы не оказаться в положении постоянно догоняющего, государство и общество должны признать наличие угрозы информационных боевых действий и постоянно заботиться о защите национальных информационных ресурсов и сохранении конфиденциальности информационного обмена через сети обмена информацией.

1.1.3. Информационная война. Определение и сфера действия

Несмотря на то, что об информационной войне все больше говорят и пишут, окончательного общепризнанного определения этого понятия пока нет.

Следует отличать информационную войну от компьютерной преступности. Любое компьютерное преступление представляет собой факт нарушения того или иного закона. Оно может быть случайным, а может быть специально спланированным; может быть обособленным, может быть составной частью обширного плана атаки. Напротив, ведение войны никогда не бывает случайным или обособленным (и может даже не являться нарушением закона), а подразумевает согласованную деятельность по использованию информации как оружия для ведения боевых действий.

Театр боевых информационных действий простирается на такие сферы как электронное поле боя, атаки сетей обмена информацией автоматизированных систем управления, инфраструктуры, промышленный шпионаж и другие виды разведки, конфиденциальность.

Электронное поле боя представлено постоянно растущим арсеналом электронных вооружений, преимущественно засекреченных. Говоря военным языком, они предназначены для боевых действий в области командования и управления войсками, или “штабной войны”.

На новые рубежи также выходит сбор разведывательной информации.

По мере появления возможности доступа к постоянно растущим объемам информации в постоянно растущем числе абонентских пунктов все более уязвимой становится конфиденциальность информации.

В министерстве обороны США было дано следующее определение информационной войны.

Определение 1.4. *Информационная война* представляет собой всеобъемлющую, целостную стратегию, призванную отдать должное значимости и ценности информации в вопросах командования, управления и выполнения приказов вооруженным силам и реализации национальной политики [37].

Это определение не очень строгое и достаточно расплывчатое. На наш взгляд, более удачное определение информационной войны следующее.

Определение 1.5. *Информационная война* – это соперничество и организованные действия конфликтующих сторон в области информационных потенциалов, проводимые с целью снижения возможностей по использованию имеющихся государственного, военного и боевого потенциалов соперника (противника) и сохранения (повышения) возможностей по использованию своих потенциалов [63].

В свою очередь информационный потенциал представляет собой совокупность информации, зафиксированной на материальных носителях или в любой другой форме, обеспечивающей ее передачу во времени и пространстве потребителям для решения широкого спектра задач, связанных с деятельностью государственных институтов, военно-промышленного комплекса и Вооруженных сил; а также силы и средства, используемые для получения, обработки, хранения и представления информации; умонастроения людей, использующих эту информацию и способных запускать и контролировать вещественно-энергетические процессы [25].

Информационная война состоит из действий, предпринимаемых для достижения информационного превосходства в обеспечении национальной военной стратегии путем воздействия на информацию и информационные системы противника с одновременным укреплением и защитой собственной информации и информационных систем.

Объектом внимания при ведении информационной войны становятся информационные системы и сети обмена информацией (включая соответствующие линии передач, обрабатывающие центры и человеческие факторы этих систем), а также информационные технологии, используемые в системах вооружений.

Поскольку информационная война, имеющая целью изменить расстановку сил в мире, связана с вопросами информации и коммуникаций, то, если смотреть в корень, это есть война за знания - за то, кому известны ответы на вопросы: что, когда, где, почему и насколько

надежным считает отдельно взятая страна и ее армия свои знания о себе и своих противниках [1.7].

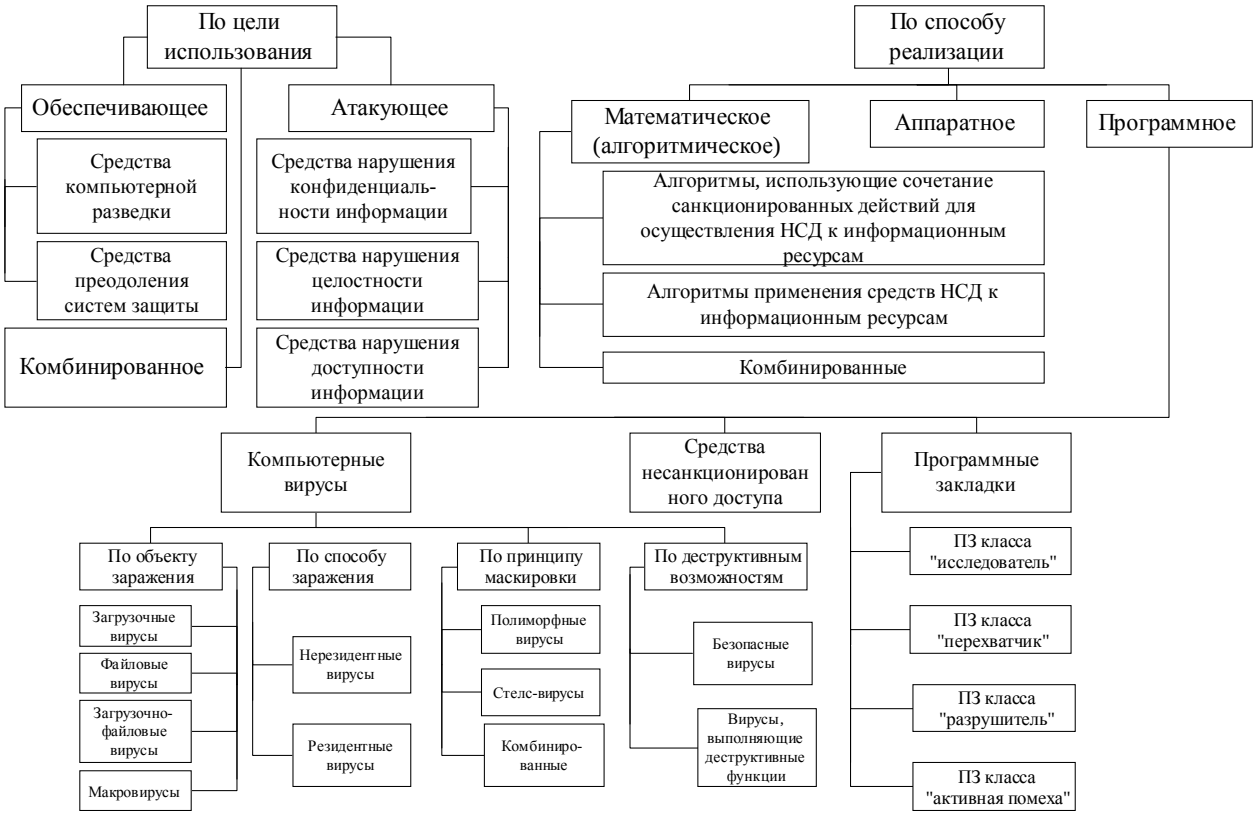


Рис. 1.1. Классификация информационного оружия

Если наступательная составляющая информационной войны связана с разработкой и использованием информационного оружия, то основными аспектами оборонительной составляющей являются, *обнаружение, реагирование и защита*.

1.2. Информационное нападение и информационное оружие

Информационное нападение представляет собой наступательную составляющую информационной войны, в том числе, и с использованием информационного оружия. Впервые информационное нападение посредством специализированного информационного обеспечения было предпринято хакерами для удовлетворения своих потребностей, чаще всего путем шантажа.

Сейчас нередко относят к информационному оружию широкий класс приемов и способов информационного воздействия на противника - от дезинформации и пропаганды до средств радиоэлектронной борьбы. Однако в работе рассматривается только часть возможного воздействия, отвечающего определ6ению.

Определение 1.6. *Информационное оружие* - это совокупность специально организованной информации, информационных технологий, позволяющая целенаправленно изменять (уничтожать, искажать), копировать, блокировать информацию, преодолевать системы защиты, ограничивать допуск законных пользователей, осуществлять дезинформацию, нарушать функционирование носителей информации, дезорганизовывать работу технических средств, компьютерных систем и информационно-вычислительных сетей, применяемая в ходе информационной борьбы (войны) для достижения поставленных целей.

Средства, используемые в качестве информационного оружия, называются средствами информационного воздействия (СИВ).

По цели использования информационное оружие делят на обеспечивающее и атакующее [58].

Определение 1.7. *Обеспечивающим* называется информационное оружие, с помощью которого осуществляются воздействия на средства защиты информации атакуемой ИВС.

В состав обеспечивающего информационного оружия входят (рис.1.1.):

- средства компьютерной разведки;
- средства преодоления системы защиты ИВС.

Определение 1.8. *Атакующим* называется информационное оружие, с помощью которого осуществляется воздействие на хранимую, обрабатываемую и передаваемую в ИВС информацию, нарушающее применяемые в ИВС информационные технологии.

В составе атакующего информационного оружия выделяют четыре основных вида средств информационных воздействий (рис.1.1.):

- *средства нарушения конфиденциальности информации;*
- *средства нарушения целостности информации;*
- *средства нарушения доступности информации;*
- *средства психологических воздействий на абонентов ИВС.*

Применение атакующего информационного оружия направлено на срыв выполнения ИВС целевых задач.

Успешное применение обеспечивающего информационного оружия позволяет осуществлять деструктивные воздействия на хранимую, обрабатываемую и передаваемую в СОИ информацию с использованием атакующего информационного оружия.

По способу реализации информационное оружие можно разделить на три больших класса:

математическое (алгоритмическое);
программное;
аппаратное.

Информационное оружие, относящееся к разным классам, может применяться совместно, а также некоторые виды информационного оружия могут нести в себе черты нескольких классов.

Специфика данной работы состоит в том, что, рассматривая все три класса информационного оружия, основной акцент делается на защиту от алгоритмического и программного информационного оружия.

К *алгоритмическому* информационному оружию относят:

алгоритмы, использующие сочетание санкционированных действий для осуществления несанкционированного доступа к информационным ресурсам;

алгоритмы применения санкционированного (легального) программного обеспечения и программные средства несанкционированного доступа для осуществления незаконного доступа к информационным ресурсам.

К *программному* информационному оружию будем относить программы с потенциально опасными последствиями своей работы для информационных ресурсов СОИ.

Определение 1.9. Под *программой с потенциально опасными последствиями* понимается некоторая самостоятельная программа (набор инструкций), которая способна выполнить любое непустое подмножество перечисленных функций.

Скрывать признаки своего присутствия в программно-аппаратной среде СОИ.

Обладать способностью к самодублированию, ассоциированию себя с другими программами и/ или переносу своих фрагментов в иные области оперативной или внешней памяти.

Разрушать (искажать произвольным образом) код программ в оперативной памяти.

Сохранять фрагменты информации из оперативной памяти в некоторой области внешней памяти прямого доступа (локальной и удаленной).

Искажать произвольным образом, блокировать и/или подменять выводимую во внешнюю память или в канал связи массив информации, образовавшийся в результате работы прикладных программ, или уже находящиеся во внешней памяти массивы данных.

Подавлять информационный обмен в телекоммуникационных сетях, фальсифицировать информацию в каналах государственного и военного управления.

Нейтрализовывать работу тестовых программ и систем защиты информационных ресурсов.

При этом под *самодублированием* программы с потенциально опасными последствиями понимается процесс воспроизведения своего собственного кода в оперативной или внешней памяти персональной ЭВМ (ПЭВМ).

Ассоциирование с другой программой - интеграция своего кода, либо его части в код другой программы таким образом, чтобы при некоторых условиях управление передавалось на код программы с потенциально опасными последствиями.

Программы с потенциально опасными последствиями можно условно разделить на следующие классы: *компьютерные вирусы*, *средства несанкционированного доступа и программные закладки*.

Определение 1.10. *Компьютерный вирус* (КВ) - это программа, обладающая способностью к скрытому размножению в среде используемой операционной системы путем включения в исполняемые или хранящиеся программы своей, возможно модифицированной копии, которая сохраняет способность к дальнейшему размножению [56].

Заражая программы, вирус может распространяться по компьютерной системе или сети обмена информацией, используя полномочия пользователей для заражения их же программ [55, 56, 74].

Для использования компьютерных вирусов в качестве программного модуля системы информационного воздействия принципиальное значение имеют следующие классификационные признаки вирусов (рис. 1.1.) [56, 74, 85]:

объект воздействия (заражения);

способ заражения объекта;

принцип маскировки;

деструктивные возможности.

Особенностью КВ является его ненаправленность на конкретные программы и также то, что во главу угла ставится самодублирование вируса. Разрушение информации вирусом не направлено на конкретного рода программы и встречается не более чем у 10% такого рода программ [74].

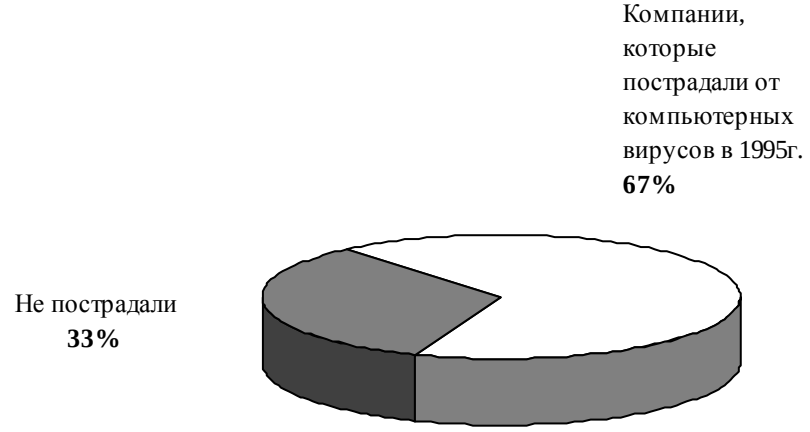


Рис. 1.2. Реальность компьютерного заражения

Таким образом, КВ способны размножаться, внедряться в программы, передаваться по линиям связи, сетям обмена информацией, выводить из строя системы управления и т.п.

В настоящее время, компьютерные вирусы представляют собой наибольшую опасность из всех видов информационного оружия. Согласно проведенному опросу [64], в 1995 году от компьютерных вирусов пострадало более 60% компаний, активно использующих информационные технологии (рис. 1.2).

Определение 1.11. *Средства несанкционированного доступа* (СНСД) - это класс программ с потенциально опасными последствиями, обязательно выполняющий функции 3-5, 7 определения 1.9.

К СНСД относится всевозможное штатное программное обеспечение СОИ, которое противник может использовать для нарушения целостности операционной системы или вычислительной среды. Часто этот тип программного обеспечения используется для анализа систем защиты, с целью их преодоления и реализации НСД к информационным ресурсам СОИ [54].

Определение 1.12. *Программные закладки* (ПЗ) - класс программ с потенциально опасными последствиями, обязательно выполняющий функции 3-5, определения 1.9.

Отличительный признак между средствами несанкционированного доступа и программными закладками - это наличие для первых и отсутствие для вторых функции преодоления защиты.

Выделяют несколько видов ПЗ: троянская программа, логическая бомба, логический люк, программная ловушка, программный червь [89, 74, 6].

Троянская программа - программа, имеющая законный доступ к системе, но выполняющая и скрытые (необъявленные) функции.

Логическая бомба - программа, осуществляющая злоумышленные действия при выполнении ряда определенных логических условий.

В качестве примера логических бомб можно назвать программные закладные устройства, заранее внедряемые в информационно-управляющие центры военной инфраструктуры, чтобы по сигналу или в установленное время привести их в действие.

Логический люк - механизм внутри операционной системы (программного обеспечения), позволяющий программе злоумышленника получить привилегированную функцию или режим работы (которые ему не были разрешены).

Логическими люками могут быть различного рода ошибки, сознательно вводимые злоумышленниками в программное обеспечение объекта.

Программная ловушка - программа, использующая ошибки или неоднозначности в программном обеспечении.

Программный червь - программа, маскирующаяся под системные средства поиска свободных вычислительных ресурсов в сети.

Сетевым червем называется компьютерный вирус, обладающий свойством самостоятельного распространения в СОИ и заражающий элементы СОИ, функциональные сегменты СОИ либо СОИ целиком.

Основные этапы функционирования сетевого червя следующие [74, 89]:

1) поиск в СОИ цели воздействия (в подавляющем большинстве случаев- ПЭВМ с известным сетевым адресом);

2) передача по СОИ АСУ своего кода на атакуемую ПЭВМ;

3) получение управления в операционной системе атакуемой ПЭВМ;

4) переход к п.1.

Основной проблемой при функционировании сетевого червя является получение управления в операционной системе атакуемой ПЭВМ. Для этого необходимо определить идентификатор и пароль абонента либо уязвимые места механизмов защиты информации. Поэтому сетевой червь должен содержать специальный программный модуль преодоления рубежей защиты (например, перехвата пароля). Часто упоминающийся в литературе сетевой червь Р. Морриса, распространившийся в сети *Internet* в 1988 году, использовал ошибки, допущенные при написании некоторых утилит, а также ошибки администрирования сетевых служб ОС UNIX [74].

Все существующие виды ПЗ можно разбить на классы в соответствии с целью их создания [74]:

ПЗ класса “исследователь”;

ПЗ класса “перехватчик”;

ПЗ класса “разрушитель”;

ПЗ класса “активная помеха”.

Кроме того, программные закладки можно классифицировать по методу и месту их внедрения и применения (то есть, по способу доставки в систему).

закладки, ассоциированные с программно-аппаратной средой (BIOS);

закладки, ассоциированные с программами первичной загрузки (находятся в MasterBoot Record или BOOT - секторах активных разделов);

закладки, ассоциированные с загрузкой драйверов, командного интерпретатора, сетевых драйверов, то есть загрузкой ОС;

закладки, ассоциированные с прикладным программным обеспечением общего назначения (встроенные в клавиатурные и экранные драйверы, программы тестирования ПЭВМ, утилиты и оболочки типа NORTON);

исполняемые модули, содержащие только код закладки (как правило, внедряемые в пакетные файлы типа .BAT);

модули-имитаторы, совпадающие с некоторыми программами, требующими ввода конфиденциальной информации, по внешнему виду;

закладки, маскируемые под программные средства оптимизационного назначения (архиваторы, ускорители и т.д.);

закладки, маскируемые под программные средства игрового и развлекательного назначения (как правило, используются для первичного внедрения закладок типа “исследователь”).

Таким образом, ПЗ имеют достаточно специфическую форму реализации *процедуры нападения*, выполнения функций разведки и исследования систем защиты (например, паролей доступа) элементов вычислительной среды.

Средствам несанкционированного доступа присущи все функции ПЗ, а также функция преодоления средств защиты СОИ. Другим основным элементом СНСД является функция нанесения ущерба, реализуемая, как правило, в виде процедуры копирования или искажения конфиденциальной информации.

Таким образом, наиболее опасным программным информационным оружием являются ПЗ и СНСД в виду сложности защиты от них, по сравнению с КВ, и возможности управления их работой со стороны злоумышленника (противника).

Правда, при этом необходимо отметить, что одним из последних видов информационного оружия стали макровирусы. Это обусловлено тем, что переход на использование электронного документооборота вызвал широкое применение приложений Microsoft Office. В этих приложениях для написания макросов используется встроенный язык WordBasic. Именно этот язык и используется для написания макровирусов.

Язык WordBasic, на котором написано большинство макровирусов, очень прост в освоении даже для начинающих пользователей компьютеров, а исходный текст работоспособного макровируса может состоять всего из нескольких строк.

Одним из наиболее наглядных примеров использования возможностей WordBasic для создания действительно эффективного информационного оружия явился вирус ShareFun, который активизирует Microsoft Mail (если эта программа была запущена до открытия зараженного документа, а это часто происходит, если она была интегрирована в Microsoft Office). При этом вирус имитирует нажатие управляющих клавиш для программы Microsoft Mail, что приводит к отсылке копий открытого документа трем адресатам, случайно выбранным из «адресной книги», причем не обязательно это будут те люди, которые имеют право на доступ к тому документу, который был открыт для просмотра или редактирования.

По аналогии с описанным выше механизмом работы вируса может быть построено соответствующее атакующее информационное оружие, которое уже не будет выбирать адре-

сата «случайным образом», а будет функционировать по строго заданной программе, возможно используя для транзита несанкционированно полученных информационных ресурсов обеспечивающее (вспомогательное) информационное оружие.

Применение макровирусов может осуществляться в сочетании с другими видами ИО. При этом достигаются следующие эффекты:

получение доступа к конфиденциальной информации в СОИ;

разрушение важной информации в СОИ;

снижение эффективности работы пользователей СОИ.

Рассмотренные виды информационного оружия в сгруппированном виде приведены на рис.1.1.

Универсальность, скрытность, многовариантность форм программно-аппаратной реализации, радикальность воздействия, достаточный выбор времени и места применения, наконец, экономичность делают информационное оружие чрезвычайно опасным: оно легко маскируется под средство защиты и даже позволяет вести наступательные действия анонимно.

В докладе Объединенной комиссии по безопасности, созданной по распоряжению министра обороны и директора Центрального разведывательного управления (ЦРУ) в США в июне 1993 года и завершившей свою работу в феврале 1994 года, говорится : ***"... Уже признано, что сети передачи данных превращаются в поле битвы будущего. Информационное оружие, стратегию и тактику применения которого еще предстоит тщательно разработать, будет использоваться с "электронными скоростями" при обороне и нападении. Информационные технологии позволят обеспечить разрешение геополитических кризисов, не производя ни одного выстрела. Наша политика обеспечения национальной безопасности и процедуры ее реализации должны быть направлены на защиту наших возможностей по ведению информационных войн и на создание всех необходимых условий для воспрещения противоборствующим США государствам вести такие войны ..."*** [16, 37].

Анализ публикаций в иностранной печати, материалов конференций позволяет сделать следующие выводы:

- информационное оружие стало одной из важных составляющих военного потенциала;

- в первую очередь новое оружие нацелено на вооруженные силы, предприятия оборонного комплекса, структуры, ответственные за внешнюю и внутреннюю безопасность страны.

Конечно, спецслужбы располагают необходимыми средствами и известным опытом предотвращения перехвата, утечек, искажения и уничтожения информации в информационных системах и телекоммуникационных сетях. Но темпы совершенствования информационного оружия (как, впрочем, и любого вида атакующего вооружения) превышают темпы развития технологий защиты и противодействия. Вот почему задача нейтрализации информационного оружия, парирования угрозы его применения должна рассматриваться как одна из приоритетных задач в обеспечении национальной безопасности страны.

Средства информационного воздействия должны сочетать в себе функции, выполняемые различными видами ИО. Такие средства образуют *комбинированные* средства информационного воздействия.

Например, в [74] представлено средство информационного воздействия, названное "сетевой шпион".

Основой сетевого шпиона являются сетевой червь. Основные этапы функционирования сетевого шпиона следующие [74]:

1) инсталляция в памяти атакуемой ПЭВМ;

2) ожидание запроса с удаленной атакующей ПЭВМ и обмен с ней сообщениями о готовности;

3) передача перехваченной информации на атакующую ПЭВМ и предоставление атакующей ПЭВМ контроля над атакуемой.

Сетевой шпион выполняет следующие основные функции:

- перехват и передача вводимой с клавиатуры информации на атакующую ПЭВМ (преодоление СЗИ, нарушение конфиденциальности информации);
- перехват и передача экранной информации на атакующую ПЭВМ (нарушение конфиденциальности информации);
- перехват и передача на атакующую ПЭВМ информации об атакуемой ПЭВМ, например, типе ОС, параметрах ПЭВМ, выполняемых программах (ведение компьютерной разведки);
- передача контроля над атакуемой ПЭВМ атакующему компьютеру. Результатом такой передачи могут быть удаленный запуск программ, уничтожение или модификация информации и другие деструктивные воздействия.

Подводя итог, отметим следующие основные результаты применения информационного оружия:

- получение информации о противостоящей стороне и его намерениях;
- невозможность выработки управляющих воздействий;
- временное лишение возможности выработки управляющих воздействий;
- снижение качества выработки управляющих воздействий;
- психологические воздействия на противника и даже физическое уничтожение людей.

Средства информационного воздействия могут быть внедрены в информационно-вычислительную сеть корпорации тремя основными способами:

- включением противником средств информационного воздействия в импортируемые ПЭВМ и их периферийное оборудование;
- агентурным путем (внесение средств информационного воздействия завербованными противником агентами из числа абонентов СОИ);
- использованием противником способов удаленных информационных атак на СОИ.

Учитывая то, что в настоящее время практически любая корпорация стремится получить выход во всемирную сеть *Internet* для своей информационно-вычислительной сети, наиболее опасным является реализация противником удаленных атак с использованием информационного оружия.

1.3. Особенности информационного оружия

Существует ряд особенностей, определяющих качественное отличие информационного оружия от других видов вооружений. К ним относятся:

- универсальность;
- скрытность;
- экономическая эффективность;
- возможность применения для решения широкого круга задач;
- масштабность применения;
- обладание эффектом “цепной реакции”;
- сложность осуществления международного контроля за разработкой и применением.

Универсальность.

Необходимым условием применения информационного оружия (осуществления удаленных атак на СОИ противостоящей стороны) является подключение атакуемой СОИ к глобальным информационно-телекоммуникационным системам типа *Internet*. В то же время виды информационного оружия и способы его применения не зависят от климатических и географических особенностей возможных театров военных действий, состояния инфраструктуры (например, состояния дорог и мостов) и т.п.

С точки зрения универсальности информационное оружие сравнимо с ракетно-ядерным (для применения ракетно-ядерного оружия отсутствует ограничение, связанное с

необходимостью выхода СОИ атакуемой стороны во внешние системы). Вместе с тем, в настоящее время существует ряд особенностей, затрудняющих возможность применения ракетно-ядерного оружия. Основные из них следующие:

- наличие международных соглашений, ограничивающих возможность применения ракетно-ядерного оружия;
- возможность гарантированной идентификации агрессора;
- нежелательные политические последствия для государства-агрессора (международная изоляция, возможность распространения зоны поражения на государства, против которых агрессия не планировалась, в том числе на государства-союзники агрессора);
- низкая экономическая эффективность ("победитель" не будет иметь возможность использования "побежденных" территорий, производственные мощности и инфраструктуру "побежденного" государства).

Наконец, массовое применение ракетно-ядерного оружия может привести к гибели человечества.

Учитывая вышесказанное, свойство универсальности информационного оружия приобретает особое значение.

Скрытность.

При совершении агрессии одного государства против другого, с которым имеется общая граница, необходимо:

- иметь мощный экономический потенциал;
- провести мобилизацию;
- создать группировку войск, способную выполнить поставленные задачи;
- нанести внезапный сокрушительный удар по противнику.

Нетрудно заметить, что перечисленные условия противоречат друг другу: возможности современной разведки таковы, что практически невозможно скрыть от предполагаемого противника процессы мобилизации и концентрации войск в приграничных районах. Следовательно, противник в ответ также начнет мобилизацию и концентрацию войск, и внезапного удара не получится. В то же время, для нанесения внезапного сокрушительного удара армии мирного времени недостаточно, а содержание в мирное время армии военного времени приведет к разорению государства, подрыву его экономики.

Применение информационного оружия позволяет снять указанные противоречия.

Схему вступления в войну государства, обладающего необходимым количеством высококачественного информационного оружия, можно обобщенно представить следующим образом:

- скрытая подготовка и внезапное проведение широкомасштабных долговременных информационных воздействий на СОИ противника. Атаки осуществляются на СОИ различного назначения (государственные, военные, СОИ транспортом, СОИ финансовых учреждений, СОИ систем энергообеспечении и т.п.) с целью лишения противостоящей стороны возможности использования этих СОИ для принятия управленческих решений. Одновременно ведется психологическая война с использованием информационных воздействий на СОИ;
- армия мирного времени сосредотачивается на границе, переформировывается в первый стратегический эшелон и начинает захват территории противника. Одновременно объявляется мобилизация, действующая армия доукомплектовывается, формируются второй и последующие стратегические эшелоны, которые завершают захват территории противника.

При отсутствии общей границы между противоборствующими сторонами захват территории противника может осуществляться подразделениями быстрого реагирования.

В данной схеме можно обеспечить *абсолютную скрытность* подготовки к войне, подготовки и осуществления информационного нападения.

Отметим еще две особенности представленной схемы вступления государства в войну:

а) сложность оперативного определения истинного агрессора для принятия контрмер;

б) отсутствие, в общем случае, необходимости наличия общей границы между противоборствующими сторонами.

Экономическая эффективность.

По предварительным оценкам, разработка и применение информационного оружия требуют, по сравнению с другими видами вооружений, существенно меньших затрат. Очевидно, что разработка и размножение средств информационного воздействия значительно дешевле создания и серийного производства традиционных видов вооружения (бронетанковой техники, ракетно-космической техники и др.). Однако, этот тезис в настоящее время требует детального экономического обоснования.

Возможность применения информационного оружия для решения широкого круга задач.

Информационное оружие может быть использовано не только в первом внезапном ударе, но и для решения стратегических, оперативно-тактических и тактических задач на поле боя. Например, информационное оружие может применяться для осуществления воздействий на средства связи и управления батальоном, ротой и взводом, психологических и физических воздействий на отдельных субъектов.

Масштабность применения.

С использованием информационного оружия возможно осуществление воздействий на стационарные и мобильные элементы СОИ наземного, морского, воздушного и космического базирования.

Обладание эффектом "цепной реакции".

Воздействия на отдельный элемент СОИ могут привести к выводу из строя других элементов СОИ, сегментов СОИ и системы целиком.

Сложность осуществления международного контроля.

Факты производства и испытаний информационного оружия, а также используемые для этого технические и программные средства могут быть надежно скрыты от разведок других государств, различных международных организаций, их контролирующих органов.

Главным недостатком информационного оружия является зависимость внутренней структуры средств информационного воздействия от применяемой в СОИ сетевой операционной системы, а также от состава и внутренней структуры механизмов, образующих систему защиты информации.

Таким образом, можно говорить об информационном оружии, как о качественно новом виде вооружений. Такой подход, безусловно, требует пересмотра вопроса о защите от воздействия оружия данного вида. Если мы говорим уже не о проделках хакеров, а о целенаправленном воздействии на системы телекоммуникаций государства-мишени, то и защита от такого воздействия не должна ограничиваться только пассивным разграничением прав доступа, но и бороться с нарушителем, вплоть до его активного подавления с применением все тех же средств информационного воздействия.

1.4. Взгляды министерства обороны США на проблемы разработки и применения информационного оружия

В настоящее время США являются мировым лидером в области ведения информационной войны и разработки информационного оружия.

Деятельность США в области создания и применения информационного оружия осуществляется по следующим основным направлениям:

- формирование организационной структуры информационных войск Министерства обороны США;
- разработка средств информационного воздействия и способов их применения;
- подготовка высококвалифицированных кадров для ведения информационной войны.

1.4.1. Формирование организационной структуры информационных войск США

С начала 90-х годов в США осуществляется формирование организационной структуры информационных войск Министерства обороны. В настоящее время каждый вид Вооруженных Сил США имеет свой собственный Центр информационной войны [37].

Первый Центр информационной войны AFIWC (Air Force Information Warfare Center) был создан в сентябре 1993 года в Военно-воздушных силах США. Центр территориально расположен на авиабазе Kelly (штат Техас, г. Сан-Антонио). В Положении о Центре его задачи определены так: “Центр AFIWC разрабатывает, поддерживает и развертывает средства борьбы с системами командования и оперативного управления противника, планирует их оперативное применение, а также обеспечивает их приобретение и испытания”.

В Военно-морских силах США в августе 1994 года создан Центр информационной войны NIWA (Navy Information Warfare Activity). Основной задачей этого Центра является исследование наиболее важных аспектов информационной войны, способствующих разрешению будущих конфликтов с участием Военно-морских сил США.

В Сухопутных войсках США проблемы информационной войны решает Служба наземных информационных боевых действий ВС США LIWA (Land Information Warfare Activity). Место дислокации Службы: штат Вирджиния, форт Белвуар.

Руководство всей деятельностью, связанной с проблемами информационной войны, в США осуществляется Министерством Обороны.

Ответственность за решение всех вопросов, связанных с информационной войной, возложена на Помощника Министра обороны по вопросам управления войсками, связи и разведки. Непосредственное руководство решением задач информационной войны осуществляет заместитель Помощника Министра обороны по вопросам информационной войны.

Для обеспечения применения информационных войск в соответствии с их предназначением предполагается создать специальные органы [37]:

1) Центр планирования и координации по вопросам информационной войны (разработка системы планирования деятельности по всем вопросам, связанным с информационной войной);

2) Центр стратегического уровня по отслеживанию признаков начала информационной войны (сбор и анализ разведывательной информации, определение признаков начала информационных атак);

3) Центр по проведению операций в области защиты от информационного оружия (предупреждение об информационных атаках тактического уровня, ликвидация последствий информационного нападения);

4) Отдел по разработке конструкций и архитектуры АСУ (разработка единых архитектур и технических стандартов в области средств и систем защиты от информационного оружия);

5) Группу независимых экспертиз - “Красную команду” (анализ уязвимости АСУ, в том числе, посредством осуществления экспериментальных атак на АСУ и их отдельные элементы).

Организационная структура информационных войск постоянно изменяется и совершенствуется. Военно-политическое руководство США часто действует методом “проб и ошибок”: создается структура, и затем проверяется ее эффективность. В случае, если структура не отвечает предъявляемым к ней требованиям, она расформировывается или изменяется.

Представленные организационная структура информационных войск и состав задач, решаемых отдельными подразделениями информационных войск, показывают, по мнению военно-политического руководства США, их постоянно возрастающее значение, в том числе, и при ведении боевых действий.

1.4.2. Разработка средств информационного воздействия и способов их применения

С появлением первых деструктивных программных средств руководство США стремилось использовать такие средства для достижения политических и военных целей.

В 1990 году Министерство обороны США открыто объявило конкурс на разработку боевого деструктивного программного средства, выполняющего следующие функции [74]:

- нарушение нормального режима функционирования линий связи и абонентских комплексов АСУ военного назначения;
- ввод в АСУ ложной информации (дезинформация противника);
- модификация программного обеспечения (в том числе программного обеспечения бортовых ЭВМ спутников связи).

Предполагалось, что внедрение деструктивного программного средства в АСУ должно осуществляться через системы радиосвязи противника.

В настоящее время разработка таких средств информационного воздействия в США продолжается ускоренными темпами.

Новым при создании компьютерных вирусов является разработка так называемых информационных организмов. Под *информационным организмом* понимается относительно короткий набор команд (до 30 команд), которым передается управление компьютером, в результате чего осуществляется захват информационно-вычислительных ресурсов АСУ (оперативной памяти, пространства на магнитных носителях информации, процессорного времени).

Внедрение троянских программ в АСУ противника осуществляется следующими способами:

- использованием механизмов удаленных атак;
- внедрением программных закладок в операционные системы и программное обеспечение, поставляемые на экспорт;
- агентурным путем.

Военно-политическое руководство США использует три представленных способа параллельно во взаимосвязи друг с другом и практически с одинаковыми приоритетами. Например, во время войны в Персидском заливе система противовоздушной обороны (ПВО) Ирака была выведена из строя с помощью деструктивных программных средств, заблаговременно внедренных в память принтеров, закупленных для АСУ ПВО Ирака. Специальной командой эти закладки были удаленно инициализированы и загружены в память ПЭВМ, входящих в состав АСУ ПВО Ирака.

Принципиальным моментом является внедрение в поставляемые на экспорт ПЭВМ и их периферийное оборудование *аппаратных закладок*, которые маскируются под обычные устройства микроэлектроники. Аппаратные закладки применяются для сбора, обработки и передачи конфиденциальной информации.

Для отработки способов применения информационного оружия военно-политическое руководство США периодически проводит военные игры, в ходе которых осуществляется моделирование атак на собственную информационную инфраструктуру. Известны случаи выполнения в ходе учений информационных атак на реально функционирующие системы. Основные результаты атак: дезорганизация функционирования АСУ и снижение доступности информации. Соответственно, в ходе учений осуществляется поиск слабых мест в системе защиты информации АСУ различного назначения.

1.4.3. Подготовка высококвалифицированных кадров для ведения информационной войны

Для реализации планов информационной войны, в том числе применения информационного оружия, требуются высококвалифицированные кадры. Их подготовка в США осуществляется в Университете национальной обороны (г. Вашингтон), где создана соответст-

вующая специальность [37]. По данной специальности обучаются представители всех видов и родов Вооруженных Сил США.

Кроме этого, планируется ввести в учебных заведениях Министерства обороны США (а также в отдельных гражданских высших учебных заведениях) курсы по обучению вопросам ведения информационной войны и информационных боевых действий (выполнения функциональных обязанностей) в условиях применения противником информационного оружия. Также планируется организовать подготовку по специальностям “защита от информационного оружия” и “администратор систем и сетей”.

Представленный комплекс мероприятий по созданию и применению информационного оружия показывает особую роль, которую руководство США отводит информационному оружию как средству достижения политических и военных целей в современной войне.

1.5. Классификация удаленных атак на объекты корпоративной сети обмена информацией

Дадим толкование некоторых понятий, используемых в работе.

Угроза безопасности СОИ – это потенциально возможное происшествие, которое может оказать нежелательное воздействие на СОИ, а также на хранимую, обрабатываемую и передаваемую в ней информацию.

Исследователи обычно выделяют три основных вида угроз безопасности СОИ [58, 7, 11]:

угрозы ***раскрытия*** конфиденциальной информации;

угрозы ***целостности***, которые заключаются в умышленном изменении данных;

угрозы ***отказа в обслуживании***, которые заключаются в блокировке доступа к неко-

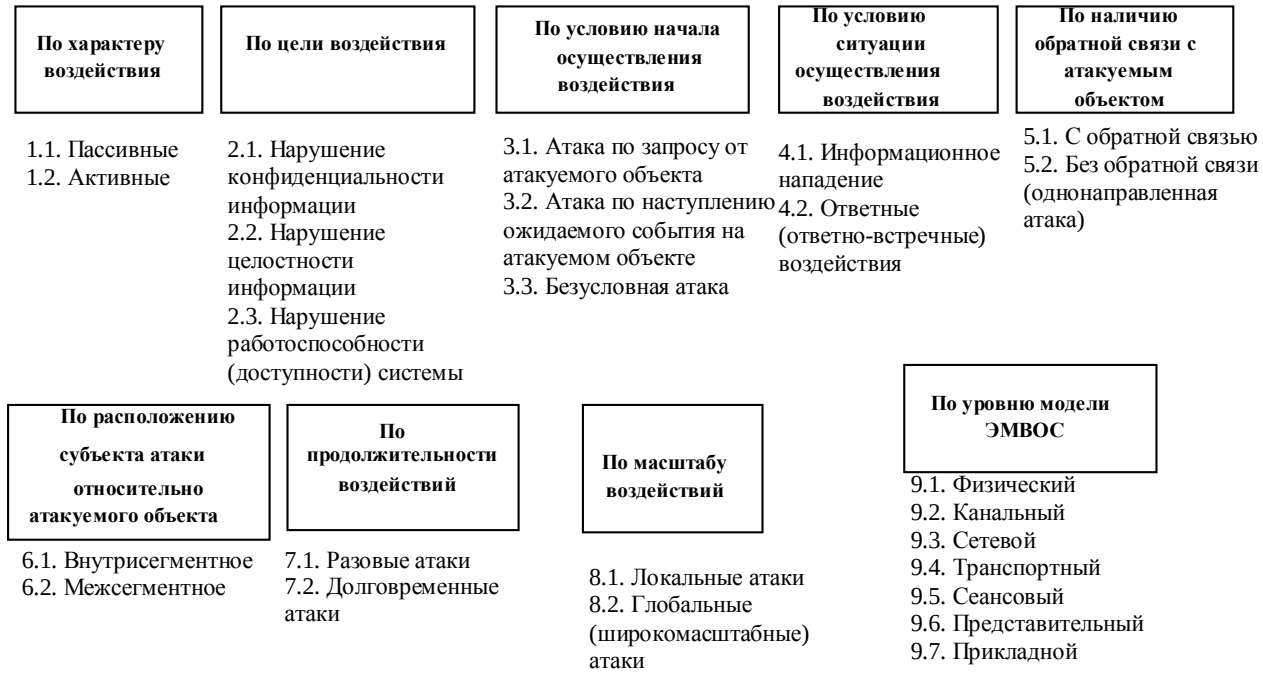


Рис. 1.3. Классификация удаленных атак на СОИ

торому ресурсу вычислительной системы или СОИ.

Уязвимость СОИ – это некоторая ее характеристика, которая обуславливает возможность возникновения угрозы.

Атака на СОИ – это реализация угрозы противником, которая заключается в поиске и использовании той или иной уязвимости.

Под *удаленной атакой* на СОИ будем понимать процесс неэнергетического (информационного) воздействия на хранимую, обрабатываемую и передаваемую в СОИ информа-

цию с целью нанесения ущерба и (или) обеспечения условий для нанесения ущерба атакуемой СОИ и(или) ее пользователям, осуществляемой по каналам связи.

Поэтому можно выделить два вида удаленных атак – удаленные атаки на инфраструктуру СОИ и протоколы сети, и удаленные атаки на телекоммуникационные службы.

При этом под ***инфраструктурой*** СОИ понимается сложившаяся система организации отношений между объектами СОИ и используемые в сети сервисные службы [24, 68].

Далее под *субъектом атаки* будем понимать атакующую сторону, а под *объектом атаки* – атакуемую сторону, которые могут быть представлены отдельным компьютером (с хранимой и обрабатываемой в нем информацией), сегментом СОИ или СОИ в целом.

Удаленные атаки можно классифицировать по следующим признакам (рис.1.3.).

По характеру воздействия:

пассивные (класс 1.1);

активные (класс 1.2);

условно-пассивные (класс 1.3).

Пассивные воздействия не оказывают непосредственного влияния на работу СОИ, но могут нарушать ее политику безопасности.

Активные воздействия имеют целью нанесение прямого ущерба СОИ, заключающегося в нарушении конфиденциальности, целостности и доступности информации, а также в осуществлении психологических воздействий на пользователей СОИ.

Очевидной особенностью активного воздействия по сравнению с пассивным является принципиальная возможность его обнаружения.

Условно-пассивные воздействия имеют целью подготовку к активной информационной атаке и включают в себя ведение компьютерной разведки преодоление СЗИ СОИ.

По цели воздействия

нарушение конфиденциальности информации либо ресурсов СОИ (класс 2.1.)

нарушение целостности информации (класс 2.2)

нарушение работоспособности (доступности) объекта СОИ (класс 2.3)

Этот классификационный признак является прямой проекцией трех основных типов угроз – *раскрытия*, *целостности* и *отказа в обслуживании*.

Основная цель практически любой атаки – получить несанкционированный доступ к информации. Существуют две принципиальные возможности доступа к информации: перехват и искажение. Возможность перехвата информации означает получение к ней доступа, но невозможность ее модификации. Следовательно, перехват информации ведет к нарушению ее *конфиденциальности.*

Возможность искажения информации означает либо полный контроль над информационным потоком между объектами системы, либо возможность передачи сообщений от имени другого объекта. Таким образом, очевидно, что искажение информации ведет к нарушению ее *целостности*.

Принципиально другой целью атаки является *нарушение работоспособно*сти объекта СОИ. В этом случае не предполагается получение атакующим несанкционированного доступа к информации. Его основная цель - добиться, чтобы операционная система на атакуемом объекте вышла из строя и для всех остальных объектов СОИ доступ к ресурсам атакованного объекта был бы невозможен.

По условию начала осуществления воздействия

Атака по запросу от атакуемого объекта (класс 3.1);

Атака по наступлению ожидаемого события на атакуемом объекте (класс 3.2);

Безусловная атака (класс 3.3).

В первом случае атакующий ожидает передачи от потенциальной цели атаки запроса определенного типа, который и будет условием начала осуществления воздействия.

Во втором случае атакующий сервер осуществляет постоянное наблюдение за состоянием операционной системы удаленной цели атаки и при возникновении определенного со-

бытия в этой системе начинает воздействие. Как и в предыдущем случае, инициатором осуществления начала атаки выступает сам атакуемый объект.

В случае *безусловной атаки* начало осуществления атаки безусловно по отношению к цели атаки, то есть атака осуществляется немедленно и безотносительно к состоянию системы и атакуемого объекта. Следовательно, в этом случае атакующий сервер является инициатором начала осуществления атаки.

По условию ситуации осуществления воздействия атаки делятся на:

информационное нападение (класс 4.1);

ответные (ответно-встречные) воздействия (класс 4.2).

Информационным нападением называется внезапное применение информационного оружия для осуществления воздействий на СОИ противостоящей стороны.

Эффективность информационного нападения достигается в том случае, если обеспечены его широкомасштабность, долговременность и скрытность.

Ответные воздействия на информационное нападение осуществляются после установления факта информационного нападения на объекты СОИ и идентификации противника. В случае правильно спланированного противником информационного нападения эффективность ответных воздействий существенно снижается.

Этот вид атак часто называют противодействием (интеллектуальным противодействием). Его отличительной особенностью является то, что субъект и объект атаки меняются местами. Атаки такого рода могут быть организованы путем подмены субъекту атаки объекта на ложный объект и использования специальных программ противодействия, имитирующих работу подлинного объекта атаки.

По наличию обратной связи с атакуемым объектом

с обратной связью (класс 5.1)

без обратной связи (однонаправленная атака) (класс 5.2)

Удаленная атака, осуществляемая при наличии обратной связи с атакуемым объектом, характеризуется тем, что на некоторые запросы, переданные на атакуемый объект, атакующему требуется получить ответ, а, следовательно, между атакующим и целью атаки существует обратная связь, которая позволяет атакующему адекватно реагировать на все изменения, происходящие на атакуемом объекте.

Данная удаленная атака может выполняться следующим образом [72]:

1) установление субъектом атаки контроля за объектом атаки (наблюдение за объектом атаки);

2) ожидание субъектом атаки установленного запроса (“доклада”) от системы информационного воздействия, функционирующего на объекте атаки;

3) выдача субъектом атаки команды на выполнение определенных операций;

4) выполнение заданных операций;

5) сообщение субъекту атаки о выполнении операций.

Далее следует переход к п.2 (или к п.3).

Таким образом, при наличии обратной связи субъект атаки имеет возможность управлять удаленной атакой (в идеальном случае - в реальном масштабе времени).

В то же время, прерывание обратной связи может привести к потере управления атакой, а, следовательно, и к прекращению атаки.

Атаки без обратной связи обычно осуществляются передачей на атакуемый объект одиночных запросов, ответы на которые атакующему серверу не нужны.

По раположению субъекта атаки относительно атакуемого объекта

внутрисегментные (класс 6.1);

межсегментные (класс 6.2).

В случае *внутрисегментной* атаки субъект и объект атаки находятся в одном сегменте.

На практике *межсегментную* атаку осуществить значительно труднее, чем внутрисегментную, но при этом, межсегментная удаленная атака представляет гораздо большую опасность, чем внутрисегментная. Это связано с тем, что в случае межсегментной атаки ее

объект и субъект могут находиться на расстоянии многих тысяч километров друг от друга, что может существенно воспрепятствовать мерам по отражению атаки.

По продолжительности воздействий

разовые атаки (класс 7.1);

долговременные атаки (класс 7.2).

Разовые атаки заключаются в ограниченных во времени целенаправленных воздействиях на объекты СОИ.

При осуществлении *долговременных атак* предусматривается проведение продолжительных по срокам многоразовых атак на объекты СОИ, как правило, с использованием различных видов ИО.

По масштабу воздействий

локальные атаки (класс 8.1);

глобальные (широкомасштабные) атаки (класс 8.2).

Локальные атаки направлены на отдельный сегмент СОИ, а в частном случае, на отдельный элемент СОИ (ПЭВМ, канал связи).

Если же атакам подвергаются несколько сегментов СОИ, то такие атаки будут *глобальными* (широкомасштабными).

По уровню эталонной модели взаимодействия открытых систем (ЭМВОС), на котором осуществляется воздействие

физический (класс 9.1);
канальный (класс 9.2);
сетевой (класс 9.3);
транспортный (класс 9.4);
сеансовый (класс 9.5);
представительный (9.6);
прикладной (класс 9.7).

Любой сетевой протокол обмена, как и любую сетевую программу, можно с той или иной степенью точности спроецировать на семиуровневую ЭМВОС. Такая многоуровневая проекция позволит описать в терминах ЭМВОС функции, заложенные в сетевой протокол или программу. Удаленная атака также является сетевой программой. В связи с этим представляется логичным рассматривать удаленные атаки на объекты СОИ, проецируя их на эталонную модель ВОС.

1.6. Анализ некоторых возможных удаленных атак в сети обмена информацией

В СОИ связь между двумя удаленными хостами осуществляется путем передачи по сети сообщений, которые заключены в пакеты обмена. Пакет, передаваемый по сети, состоит из заголовка и поля данных. в заголовок пакета заносится служебная информация, определяемая используемым протоколом обмена и необходимая для адресации пакета, его идентификации, преобразования и т. п. Причем в поле данных помещаются либо непосредственно данные, либо другой пакет более высокого уровня ВОС.

В настоящее время в СОИ наиболее широко применяются сетевые операционные системы, использующие протоколы TCP/IP или им подобные. При этом пакет обмена выглядит следующим образом.

Ethernet-заголовок	IP-заголовок	TCP-заголовок	Данные

Базовый сетевой протокол – IP (Internet Protocol). Для передачи IP-пакета на хост необходимо указать в IP-заголовке в поле Destination Address 32-разрядный IP-адрес данного хоста. IP-пакет находится внутри аппаратного пакета (например, Ethernet), поэтому он в конечном счете адресуется на аппаратный адрес сетевого адаптера.

Таким образом, для адресации пакетов в сети, кроме IP-адреса хоста, необходим еще либо IP-адрес его сетевого адаптера (в случае адресации внутри одной подсети), либо Ethernet-адрес маршрутизатора (в случае межсетевой адресации).

Задачу удаленного поиска информации о Ethernet-адресах других хостов (хостов находящихся в одном сегменте и маршрутизатора) хост решает, используя протокол ARP (Address Resolution Protocol). Протокол ARP позволяет получить взаимно однозначное соответствие IP- и Ethernet-адресов для хостов, находящихся внутри одного сегмента. Это достигается следующим образом: при первом обращении к сетевым ресурсам хост отправляет широковещательный ARP-запрос на Ethernet-адрес FFFFFFFFFFFFh, в котором указывается IP-адрес маршрутизатора и просит сообщить его Ethernet-адрес. Этот широковещательный запрос получат все станции в данном сегменте сети, в том числе и маршрутизатор. получив данный запрос, маршрутизатор внесет запись о запросившем хосте в свою ARP-таблицу , а затем отправит на запросивший хост ARP-ответ, в котором сообщит свой Ethernet-адрес. Полученный в ARP-ответе Ethernet-адрес будет занесен в ARP-таблицу, находящуюся в памяти операционной системы на запросившем хосте и содержащую записи соответствия IP- и Ethernet-адресов для хостов внутри одного сегмента.

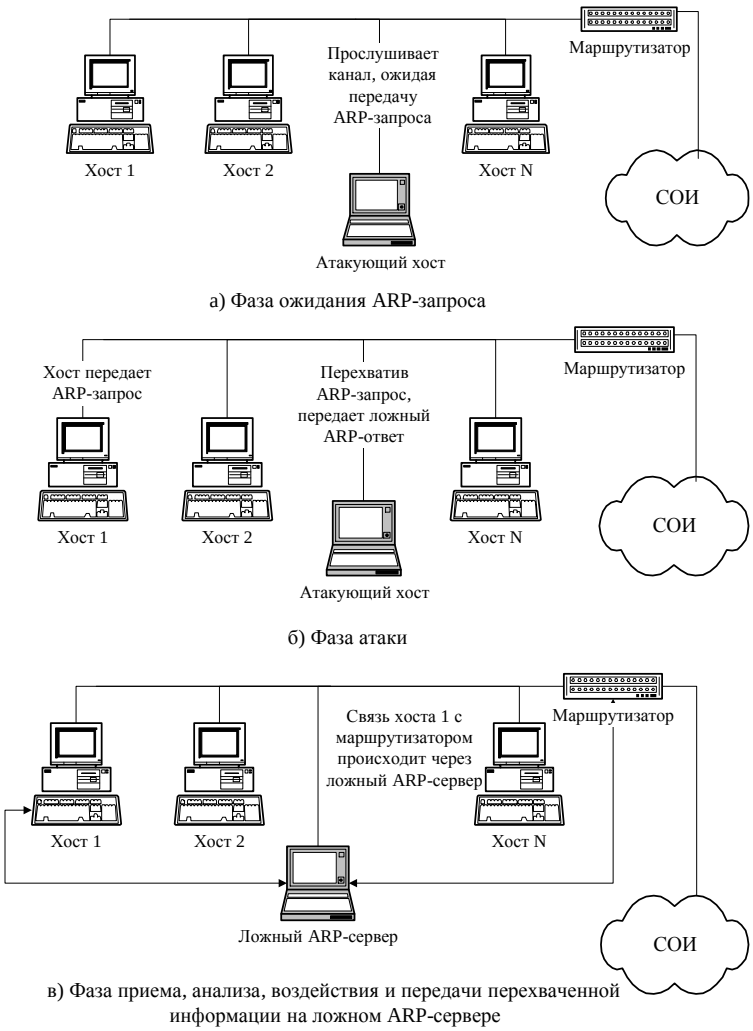


Рис. 1.4. Действие ложного ARP-сервера

Аналогично описанной процедуре хост узнает Ethernet-адрес любого другого хоста, расположенного в рассматриваемой подсети.

Таким образом, появляется возможность осуществить удаленную атаку по принципу ложного объекта. В качестве ложного объекта (ложного ARP-сервера) выступает атакующий хост.

1.6.1. Атака на основе ложного сервера адресов сетевых адаптеров объектов СОИ

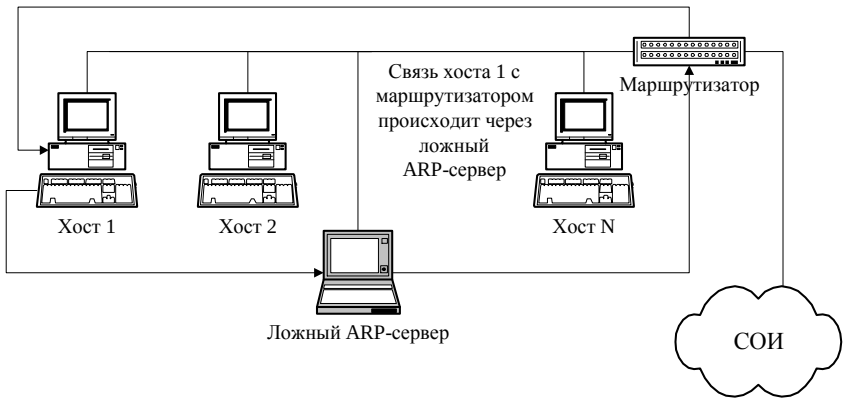


Рис. 1.5. Петлевая схема перехвата ARP-запроса

Обобщенная функциональная схема ложного ARP-сервера выглядит сле-

дующим образом (рис. 1.4.):

ожидание ARP-запроса;

при получении ARP-запроса передача по сети на запросивший хост ложного ARP ответа, в котором указывается адрес сетевого адаптера атакующей станции;

прием, анализ, воздействие и передача пакетов обмена между взаимодействующими хостами.

При этом схема передачи пакетов будет следующей:

- атакованный хост передает пакеты на ложный ARP-сервер;
- ложный ARP-сервер передает принятый от атакованного хоста пакет на маршрутизатор;
- маршрутизатор, в случае получения ответа на переданный запрос, передает его непосредственно на атакованный хост, минуя ложный ARP-сервер.

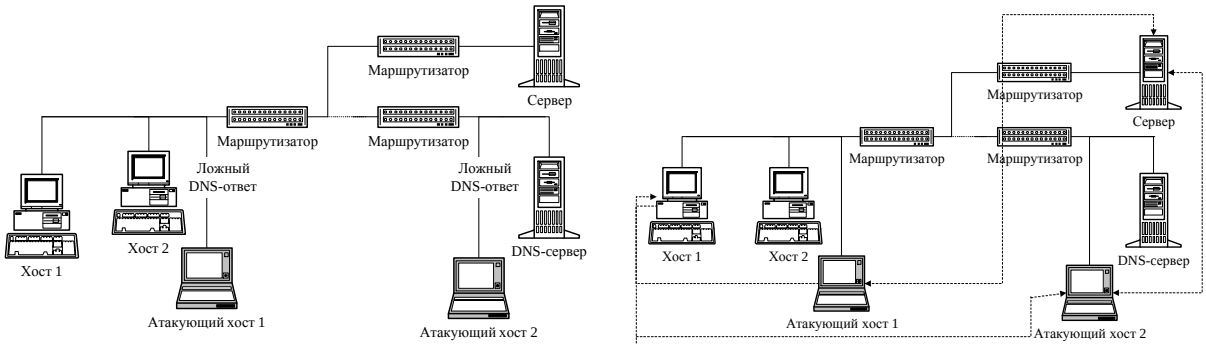
Таким образом, ложный ARP-сервер действует в режиме «полуперехвата», при этом маршрут пакетов образует петлю (рис. 1.5.).

Возможны несколько способов, позволяющих функционировать ложному ARP-серверу по мостовой (полной) схеме перехвата.

Ложный ARP-сервер, получив ARP-запрос, посылает такой же запрос, присваивая себе данный IP-адрес. Недостаток этого подхода заключается в том, что ложный ARP-сервер легко обнаруживается (некоторые сетевые ОС, например, Windows'95 и Sun OS 5.3, перехватив запрос ложного ARP-сервера, выдадут предупреждение об использовании их IP-адреса).

Ложный ARP-сервер, получив ARP-запрос, посылает ARP-запрос, указав в качестве своего IP-адреса любой свободный в данном сегменте IP-адрес. В дальнейшем ложный ARP-сервер ведет работу с данного IP-адреса как с маршрутизатором, так и с «обманутыми» хостами (proxy-схема).

Вывод. Описанный способ воздействия на СОИ относится к внутрисегментным атакам и представляет угрозу в том случае, если атакующий находится внутри данного сегмента.

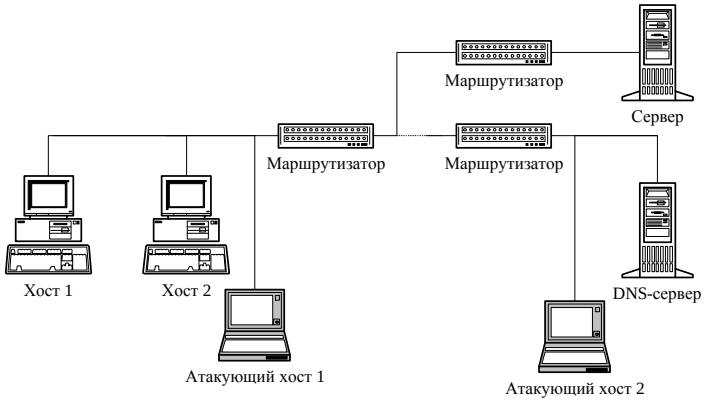


б) Фаза передачи атакующим хостом ложного DNS-запроса

в) Фаза приема, анализа, воздействия и передачи перехваченной информации на ложном DNS-сервере

Рис. 1.6. Ложный DNS-сервер (фаза передачи и приема)

Таким образом, исполнителем данной атаки может быть доверенное лицо, обладающее пусть даже незначительными полномочиями в интересующей противника подсети СОИ.



а) Фаза ожидания атакующим хостом DNS-запроса

Рис. 1.7. Ложный DNS-сервер (фаза ожидания)

1.6.2. Атака на основе ложного сервера сетевых имен узлов СОИ

Для обращения к хостам в СОИ используются 32-разрядные IP-адреса, уникально

идентифицирующие каждый сетевой. Пользователи оперируют не с самими 32-разрядными IP-адресами, а с их мнемоническими представлениями.

Использование мнемонически понятных для пользователя имен породило проблему преобразования имен в IP-адреса. Такое преобразование необходимо, так как на сетевом уровне адресация пакетов идет не по именам, а по IP-адресам.

Для решения задачи преобразования имен используется доменная система имен – DNS (Domain Name System).

Эта система позволяет пользователю в случае отсутствия у него информации о соответствии имен и IP-адресов получить необходимые сведения от ближайшего информационно-поискового сервера (DNS-сервера).

Таким образом, перед хостом возникает стандартная проблема удаленного поиска: по имени удаленного хоста найти его IP-адрес; решением этой проблемы и занимается служба DNS на базе протокола DNS.

Рассмотрим DNS-алгоритм удаленного поиска IP-адреса по имени хоста:

- хост посылает на IP-адрес ближайшего DNS-сервера (он устанавливается при настройке сетевой ОС) DNS-запрос, в котором указывает имя хоста (сервера), IP-адрес которого необходимо найти;

- DNS-сервер, получив запрос, просматривает свою базу имен на наличие в ней указанного в запросе имени. В случае, если имя найдено, а, следовательно, найден и соответствующий ему IP-адрес, то на запросивший хост DNS-сервер отправляет DNS-ответ, в котором указывает искомый IP-адрес. В случае, если указанное в запросе имя DNS-сервер не обнаружил в своей базе имен, то DNS-запрос отсылается DNS-сервером на один из корневых DNS-серверов, адреса которых содержатся в файле настроек DNS-сервера, и описанная процедура повторяется, пока имя не будет найдено (или не найдено).

Очевидно, что такая схема DNS уязвима для типовой удаленной атаки «Ложный объект сети».

При реализации атаки на службу DNS необходимо учитывать следующие особенности.

Служба DNS функционирует на базе протокола UDP, что делает ее менее защищенной, так как протокол UDP в отличие от TCP не предусматривает средств идентификации сообщений.

Значение поля «порт отправителя» в UDP-пакете принимает начальное значение ≥ 1023 и увеличивается с каждым переданным DNS-запросом.

Значение идентификатора (ID) DNS-запроса ведет себя следующим образом (в случае передачи DNS-запроса с хоста это значение зависит от конкретного сетевого приложения, вырабатывающего DNS-запрос):

- ID=1 (при передаче запроса из оболочки командного интерпретатора операционных систем Linux и Windows'95);

- $ID_{n+1}=ID_n+1$ (при передаче запроса из Netscape Navigator);

- $ID_{n+1}=ID_n+1$ (при передаче запроса непосредственно DNS-сервером).

Для реализации удаленной атаки на DNS-службу атакующий хост функционирует как ложный DNS-сервер.

Имеется три основных варианта удаленной атаки на службу DNS:

- внедрение в СОИ ложного DNS-сервера путем перехвата DNS-запроса;

- внедрение в СОИ ложного сервера путем создания направленного «шторма» ложных DNS-ответов на атакуемый хост;

- комбинированный вариант атаки.

Рассмотрим лишь первый вариант атаки - внедрение в СОИ ложного DNS-сервера путем перехвата DNS-запроса.

Обобщенная схема атаки следующая (рис. 1.6. и рис. 1.7.):

- ожидание DNS-запроса;

- извлечение из полученного запроса необходимых сведений (номер UDP-порта отправителя запроса, двухбайтовое значение ID идентификатора DNS-запроса и искомое мне-

моническое имя) и передача по сети на запросивший хост (на извлеченный из DNS-запроса UDP-порт) ложного DNS-ответа, от имени (с IP-адреса) настоящего DNS-сервера, в котором указывается IP-адрес ложного DNS-сервера;

- в случае получения пакета от сервера (запрашиваемого хоста), изменение в IP-заголовке пакета его IP-адреса на IP-адрес ложного DNS-сервера и передача пакета на хост (для хоста ложный DNS-сервер и есть настоящий сервер).

Условия осуществления данной удаленной атаки:

- перехват DNS-запроса (необходимое условие);
- атакующий находится либо на пути основного трафика, либо в сегменте настоящего DNS-сервера.

Два других варианта удаленной атаки на службу DNS достаточно подробно описаны в работах [74, 77].

1.6.3. Навязывание сетевому узлу ложного маршрута с целью создания в СОИ ложного маршрутизатор

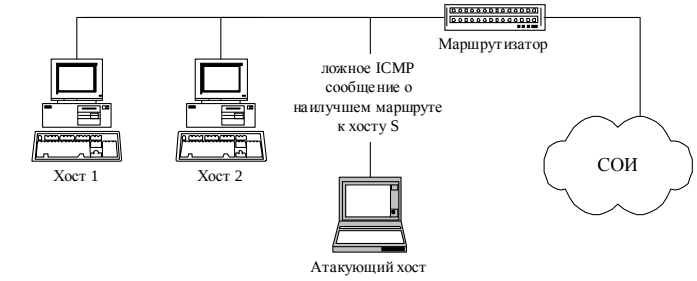
Маршрутизация в СОИ осуществляется на сетевом уровне (IP-уровень). Каждый сегмент сети подключается к базовой СОИ как минимум через один маршрутизатор, а, следовательно, все хосты в этом сегменте и маршрутизатор должны физически размещаться в одном сегменте. Поэтому все сообщения, адресованные в другие сегменты сети, направляются на маршрутизатор, который, в свою очередь, перенаправляет их далее по указанному в пакете IP-адресу, выбирая при этом оптимальный маршрут.

Для обеспечения маршрутизации в памяти сетевой ОС каждого хоста существуют таблицы маршрутизации, содержащие данные о возможных маршрутах. В каждой строке таблицы маршрутизации хоста содержится описание соответствующего маршрута.

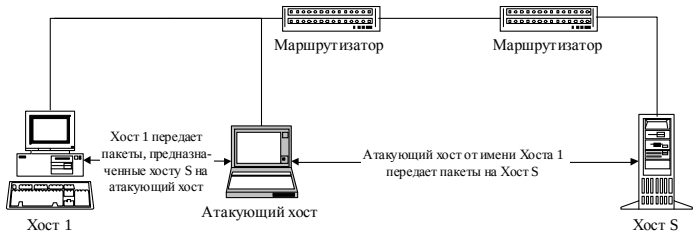
Это описание включает:

- IP-адрес конечной точки маршрута (Destination);
- IP-адрес соответствующего маршрутизатора (Gateway);
- ряд других параметров, характеризующих маршрут.

Обычно в системе существует так называемый маршрут по умолчанию (поле Destination содержит значение 0.0.0.0, то есть default, а поле Gateway – IP-адрес маршрутизатора). Этот маршрут означает, что все пакеты, адресуемые на IP-адрес вне пределов данной подсети, будут направляться по указанному default-маршруту, то есть на маршрутизатор (это реализуется установкой в поле адреса назначения в Ethernet-пакете аппаратного адреса маршрутизатора).



а) Фаза передачи ложного ICMP Redirect сообщения от имени маршрутизатора



б) Фаза приема, анализа, воздействия и передачи перехваченной информации на атакующем хосте

Рис. 1.8. Внутрисегментное навязывание хосту ложного маршрута при использовании протокола ICMP

Функция удаленного управления маршрутизацией на хостах внутри сегмента сети возлагается на протокол ICMP.

Удаленное управление маршрутизацией необходимо для предотвращения возможной передачи сообщений по неоптимальному маршруту.

Удаленное управление маршрутизацией реализуется в виде передачи с маршрутизатора на хост управляющего ICMP-сообщения: Redirect Message. Со-

общение Redirect бывает двух типов: Redirect Net и Redirect Host.

Сообщение типа Redirect Net уведомляет хост о необходимости смены адреса маршрутизатора, то есть default-маршрута; сообщение типа Redirect Host информирует хост о необходимости создания нового маршрута к указанной в сообщении системе и внесения ее в таблицу маршрутизации. Для этого в сообщении указывается IP-адрес хоста, для которого необходима смена маршрута (адрес будет занесен в поле Destination), и новый IP-адрес маршрутизатора, на который необходимо направлять пакеты, адресованные данному хосту (этот адрес заносится в поле Gateway).

Адрес нового маршрутизатора должен быть в пределах адресов данной подсети.

Большинство сетевых ОС игнорируют ICMP-сообщение Redirect Net.

Единственным параметром, идентифицирующим управляющее сообщение ICMP Redirect Host, является IP-адрес отправителя, который должен совпадать с IP-адресом маршрутизатора, так как это сообщение может передаваться только маршрутизатором.

Особенность протокола ICMP состоит в том, что он не предусматривает никакой дополнительной аутентификации источников сообщений и при этом реализован на базе протокола UDP. Таким образом, ICMP-сообщения передаются на хост маршрутизатором однонаправленно, без создания виртуального соединения.

Следовательно, существует реальная возможность послать с атакующего хоста ложное ICMP-соединение о смене маршрута от имени маршрутизатора, то есть осуществить типовую удаленную атаку «Внедрение в ИВС ложного объекта путем навязывания ложного маршрута».

Для осуществления этой удаленной атаки необходимо подготовить ложное ICMP Redirect Host сообщение, в котором указать конечный IP-адрес маршрута (адрес хоста, маршрут к которому будет изменен) и IP-адрес ложного маршрутизатора. Затем это сообщение передается на атакуемый хост от имени маршрутизатора. Для этого в IP-заголовке в поле адреса отправителя указывается IP-адрес маршрутизатора.

Существует два варианта реализации данной атаки:

- внутрисегментная атака;
- межсегментная атака.

В первом случае атакующий хост находится в том же сегменте сети, что и цель атаки. Тогда, послав ложное ICMP-сообщение, он в качестве IP-маршрутизатора может указать либо свой IP-адрес, либо любой из адресов данной подсети.

Функциональная схема осуществления первого варианта атаки выглядит следующим образом (рис. 1.8.):

- передача на атакуемый хост ложного ICMP Redirect Host сообщения;
- отправление ARP-ответа в случае, если пришел ARP-запрос от атакуемого хоста;
- перенаправление пакетов от атакуемого хоста на настоящий маршрутизатор;
- перенаправление пакетов от маршрутизатора на атакуемый хост;
- при приеме пакета возможно воздействие на информацию.

В случае осуществления второго варианта удаленной атаки атакующий находится в другом сегменте относительно цели атаки. Поэтому, даже передав ложное ICMP Redirect сообщение, атакующий не сможет получить контроль над трафиком, так как адрес нового маршрутизатора должен находиться в пределах подсети атакуемого хоста.

В этом случае достигается иная цель - нарушение работоспособности хоста, так как в случае успешной атаки все пакеты, направляемые хостом на определенный сервер, будут отправлены на IP-адрес несуществующего маршрутизатора.

Вывод. Атака, использующая «слабости» протокола ICMP, осуществляется намного легче, чем рассмотренные выше атаки протокола ARP и службы DNS.

Очевидно, что рассмотренные атаки – это лишь иллюстрация возможностей удаленных атак на СОИ. Существует множество других возможностей осуществления удаленных атак на объекты СОИ. При этом успешно проводимые атаки на нижних уровнях ЭМВОС обуславливают успех атак на верхних уровнях ЭМВОС. Например, перехватив передавае-

мую в сети парольную и ключевую информацию, можно осуществлять доступ к информации в базах данных или в информационных хранилищах.

1.7. Оборонительная составляющая информационной войны в СОИ

Как отмечалось выше, основными аспектами оборонительной составляющей информационной войны в СОИ являются защита информационных ресурсов, обнаружение НСД и реагирование (противодействие противнику).

Защита информационных ресурсов обеспечивается информационной безопасностью и рассматривается в теории обеспечения безопасности информации (ОБИ)[25].

Теория обеспечения безопасности информации объединяет основные научные положения прикладной теории алгоритмов, теории передачи информации, теории кодирования, криптологии и с единых системных позиций изучает методы предотвращения случайного или преднамеренного раскрытия, искажения или уничтожения хранимой, обрабатываемой или передаваемой информации в системах управления, функционирующих на базе средств вычислительной техники и СОИ [5, 41, 42, 59, 92].

Приведем ряд определений, используемых в теории ОБИ [25,75].

Безопасность данных – такое состояние хранимых, обрабатываемых и передаваемых данных, при котором невозможно их случайное или преднамеренное получение, изменение или уничтожение [82].

Защита данных – совокупность целенаправленных действий и мероприятий по обеспечению безопасности данных.

Метод (способ) защиты данных – совокупность приемов и операций, реализующих функции защиты данных.

На основе методов (способов) защиты создаются ***средства защиты***.

Механизм защиты – совокупность средств защиты, функционирующих совместно для выполнения определенной задачи по защите данных.

Система обеспечения безопасности данных – совокупность средств и механизмов защиты данных.

Необходимо отметить, что большинство научных и исследовательских работ посвящено именно ОБИ и касаются, по сути, разработки предупредительных методов и средств защиты информации [74, 10], которые совершенно не приспособлены к ведению динамичной информационной борьбы в СОИ, учитывающей изменения стратегий противника, развития ИО, текущих задач СОИ и СУ в целом.

Два же других аспекта оборонительной составляющей – обнаружение несанкционированных действий в СОИ и информационное противодействие в СОИ – практически не исследованы, хотя наиболее перспективны.

Если сравнивать информационную «оборону» в СОИ с обороной боевого подразделения в обычной войне, то ОБИ в СОИ можно поставить в соответствие инженерному оборудованию опорного пункта подразделения, а обнаружению НСД и противодействию – действия командиров (их интеллект) по управлению подразделением в ходе оборонительного боя. Такое сравнение ярко иллюстрирует необходимость разработки методов интеллектуального противодействия в СОИ.

Нерешенной до конца остается и задача обнаружения попыток НСД. При этом перспективным видится обнаружение НСД и СИВ в СОИ с применением нейронных сетей и других математических моделей биологических объектов [67, 38]. Этот подход может быть развит методами построения программных “приманок”, позволяющих быстрее распознавать СИВ в СОИ.

Интересен подход к обнаружению РПС, предложенный авторами [30]. Он основан на предположении подобия кода программы, представленного в шестнадцатеричном виде, с белковыми структурами, элементами которых являются шестнадцать белков.

При этом, на основе корреляции с существующими РПС можно выявлять ранее неизвестные РПС, причем корреляция осуществляется не на уровне программного кода как последовательности чисел, а на уровне структуры РПС, определяющей ее функции.

За рубежом также уделяют большое внимание оборонительным информационным действиям. Министерство обороны США рассматривает оборонительные информационные действия как нечто большее, но одновременно и использующее традиционные подходы типа COMPSEC, COMSEC, OPSEC и INFOSEC [74, 84]. Способность противостоять нападению подразумевает применение всех этих подходов, но что более важно, включает в себя интегрированные средства защиты, обнаружения и реагирования.

Защита от вторжений и злоупотреблений требует эффективных механизмов идентификации и аутентификации, высоконадежных брандмауэров, равно как и методов проверки и отслеживания.

В США считают, что для предотвращения или нейтрализации последствий применения информационного оружия необходимо принять следующие меры [84, 104]:

- защита материально-технических объектов, составляющих физическую основу информационных ресурсов;
- обеспечение нормального и бесперебойного функционирования баз и банков данных;
- защита информации от несанкционированного доступа, искажения или уничтожения;
- сохранение качества информации (своевременности, точности, полноты и необходимой доступности).

Создание технологий обнаружения воздействий на информацию, в том числе в телекоммуникационных сетях, - это естественная защитная реакция на появление нового оружия.

Защита сетевого оборудования на территории России от проникновения в него скрытых элементов информационного оружия особенно важна сегодня, когда осуществляются массовые закупки зарубежных информационных технологий.

Запретить разработку и использование информационного оружия вряд ли возможно, как это сделано, например, для химического или бактериологического оружия. Ограничить усилия многих стран по формированию единого глобального информационного пространства также невозможно.

Поэтому, необходимо решать проблему защиты СОИ от информационного нападения и информационного оружия путем совершенствования разработки соответствующих эффективных методов и средств противодействия, учитывающих эволюцию информационного оружия и способов его применения.

1.8. Противодействие информационному нападению

Как отмечалось выше, основными аспектами оборонительной составляющей информационной войны являются защита информационных ресурсов, обнаружение НСД и реагирование (противодействие).

Как отмечалось выше, защита информационных ресурсов обеспечивается информационной безопасностью. Вместе с тем, обнаружение попыток НСД является до настоящего времени до конца нерешенной проблемой. При этом реагирование невозможно без определенной осмотрительности и осторожности.

Выше мы уже говорили о том, что Министерство обороны США рассматривает оборонительные информационные действия как нечто большее, но одновременно и использующее традиционные подходы. Считается, что способность противостоять нападению подразумевает применение всех этих подходов, но что более важно, включает в себя интегрированные средства защиты, обнаружения и реагирования.

Общепринято, что защита от вторжений и злоупотреблений требует эффективных механизмов идентификации и аутентификации, высоконадежных брандмауэров, равно как и методов проверки и отслеживания сомнительных ситуаций.

Считается, что для предотвращения или нейтрализации последствий применения информационного оружия необходимо принять следующие меры:

- защита материально-технических объектов, составляющих физическую основу информационных ресурсов;
- обеспечение нормального и бесперебойного функционирования баз и банков данных;
- защита информации от несанкционированного доступа, искажения или уничтожения;
- сохранение качества информации (своевременности, точности, полноты и необходимой доступности).

Создание технологий обнаружения воздействий на информацию, в том числе в телекоммуникационных сетях, - это естественная защитная реакция на появление нового оружия. Защита сетевого оборудования от проникновения в него скрытых элементов информационного оружия особенно важна сегодня, когда осуществляются массовые закупки зарубежных информационных технологий.

Запретить разработку и использование информационного оружия вряд ли возможно, как это сделано, например, для химического или бактериологического оружия. Ограничить усилия многих стран по формированию единого глобального информационного пространства также невозможно.

Поэтому необходимо решать проблему защиты СОИ от информационного оружия путем разработки соответствующих эффективных методов и средств, *учитывающих эволюцию информационного оружия*.

В дальнейшем наряду с термином “информационное оружие” будем использовать уже устоявшийся за долгие годы термин “средство осуществления НСД”.

1.9. Концептуальная модель информационной борьбы в корпоративной сети обмена информацией

Разработка эффективных методов информационной борьбы в СОИ должна, так или иначе, базироваться на некоторой модели такой борьбы в сети.

Поэтому ниже мы предлагаем концептуальную модель информационной борьбы (МИБ) в СОИ, которая, по нашему мнению, в наибольшей степени отвечает задачам системного подхода к этой проблеме [66,76].

Основные компоненты модели представлены в виде блоков (рис. 1.9.), содержащих определенные атрибуты и структурно-функциональные связи, порождаемые компонентами.

Ядром этой модели выступает система блоков «Противник», «Система защиты информации», «Пользователь». Ядро МИБ погружено в среду, в качестве которой в работе выступает корпоративная сеть обмена информацией.

При разработке концептуальной модели ИБ были приняты следующие исходные положения.

Все виды угроз субъективны, так как они связаны с действиями субъектов (противника, персонала защиты, пользователя). Внешние факторы, действующие на данную систему, могут приводить к инициированию угроз, исходящих от субъектов, усиливать или ослаблять их проявление.

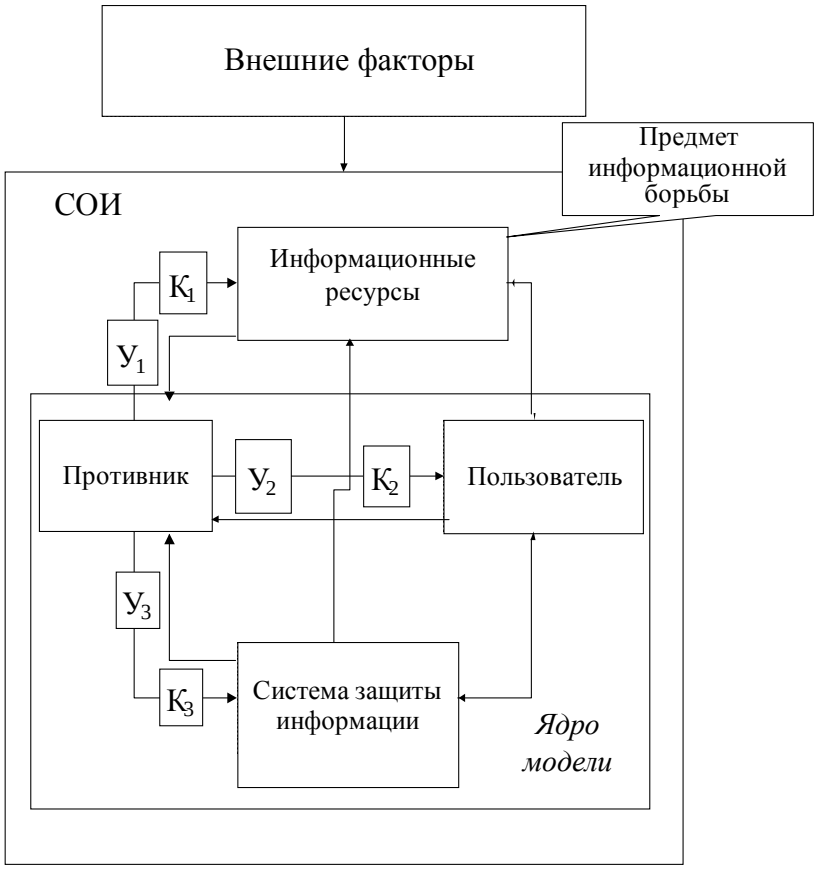
Угрозы и каналы их реализации, используемые нарушителем, могут быть модифицированы им при неэффективности их прежнего варианта.

Методы и средства защиты от определенной угрозы должны быть модифицированы службой защиты при неэффективности их прежнего варианта [67].

Изменение среды (СОИ) может приводить к появлению новых СИВ, способов их использования и средств защиты.

Под *угрозой* будем понимать потенциальную возможность нежелательного преднамеренного или непреднамеренного действия [74]. Угроза может быть связана только с субъек-

том и поэтому, в определенной степени, детерминирована. Внешние факторы (ВФ), действующие на систему, представляют, как правило, стохастические процессы. Их действие можно рассматривать как некоторый “шум”, накладывающийся на результаты проявления



У₁ , У₂ , У₃ - угрозы; **К₁ , К₂ , К₃** - каналы реализации

Рис. 1.9. Концептуальная модель информационной борьбы

угроз, инициируемых Противником.

Рассмотрим последовательно блоки предлагаемой модели, их характеристики и взаимосвязь.

Блок ***“Противник”.*** Стратегия действия противника $s_{нар}$ направлена на получение несанкционированного доступа (НСД) к защищаемой информации **I** в заданном временном интервале $\Delta \mathbf{t_{зад}}$ с высокой вероятностью результата $\mathbf{p_1}$ при минимизации затрат $\mathbf{R_н}$ на такое получение и последующей манипуляцией добытой информацией (уничтожение, модификация, присвоение, распространение и т.п.).

Формально это выглядит следующим образом:

$$s_{нар}: \begin{cases} p_1 = \max\limits_{s \in S} p(acsess(I)) \\ p_2 = \min\limits_{s \in S} p(acsess(I')) \\ p_3 = \min\limits_{s \in S} p(noacsess) \\ R_H = \min\limits_{s \in S} \sum\limits_{i=1}^{n} (a_i r_i) \\ F_{H=} \min\limits_{s \in S} \sum\limits_{i=1}^{m} b_i f_i . \end{cases} \qquad (1.1)$$

где

$S_{нар}$ - множество стратегий противника;

acsess - событие, заключающееся в успешном доступе противника к информации;

no acsess - событие, заключающееся в отказе в доступе противника;

p_1 - вероятность получения противником доступа к требуемой информации I;

p_2 - вероятность получения противником доступа к информации I', выдаваемой за I;

p_3 - вероятность неполучения противником доступа к информации;

$R_н$ - числовое значение суммы приведенных затрат противника;

$F_н$ - числовое значение суммы приведенных видов обнаружения действий противника;

R - множество всех учитываемых затрат противника;

$r_i \in R$ - значение затрат вида i;

a_i - весовой коэффициент, отражающий существенность i-го вида затрат для противника;

F - множество всех учитываемых видов возможностей обнаружения действий противника;

$f_i \in F$ - i-й вид обнаружения действий противника, причем

$$f_i = \begin{cases} 0, \text{ если данный вид обнаружения имеет место} \\ 1, \text{ в противном случае.} \end{cases}$$

Очевидно, что

$$\mathbf{p_1+p_2+p_3=1} \qquad (1.2)$$

В частном случае, целью атаки может быть получение несанкционированного доступа к защищаемой информации (информационным ресурсам СОИ) и последующая манипуляция полученной информацией (уничтожение, модификация, блокировка, распространение) [67].

В соответствии с указанной стратегией Противник может порождать угрозы трех видов [74].

Угрозы первого вида ($У_1$) направлены против "Пользователя" как субъекта, владеющего защищаемой информацией. Угрозы первого вида могут быть реализованы через каналы первого вида ($К_1$).

Угрозы второго вида ($У_2$) направлены непосредственно против информационных ресурсов и технических средств (ТС) их обеспечения и реализуются через каналы второго вида ($К_2$).

Угрозы третьего вида ($У_3$) направлены против СЗИ, то есть против структуры, технических средств (ТС), ПО и персонала службы защиты и реализуются через каналы реализации угроз третьего вида ($К_3$).

Можно также предположить наличие универсальных каналов, через которые могут реализовываться угрозы различных видов.

Для того, чтобы повысить вероятность $\mathbf{p_1}$, в $\Delta t_{зад}$ затраты $R_н$ нарушителя растут, но он их стремится минимизировать. Поэтому они растут не прямо пропорционально росту p_1, а, например, подобно тому, как это представлено на рис. 1.10.

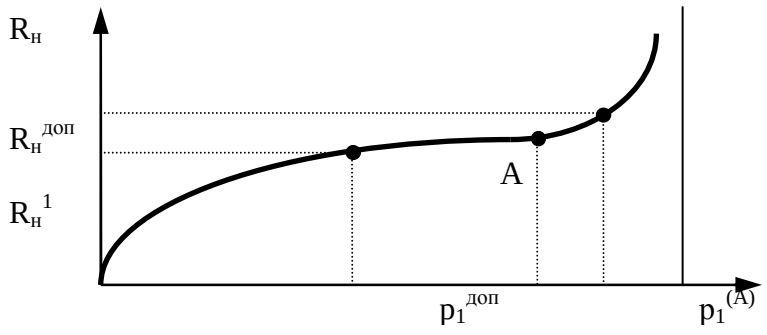


Рис.1.10. Зависимость затрат $R_н$ от вероятности p_1

Путем совершенствования средств и механизмов нападения нарушитель стремится изменить положение точки перегиба **A**, точнее - переместить ее вправо с тем, чтобы повысить значение p_1^{max} (максимально возможное значение p_1 при ограничении на затраты $\max R_н = R_н^{доп}$).

Зависимость затрат $R_н$ от вероятности p_2 приведена на рис.1.11.

Аналогично, как и в случае с p_1, путем совершенствования средств и механизмов нападения нарушитель стремится изменить положение точки **B**, а именно - переместить ее вле-

во с тем, чтобы снизить значение p_2^{min} (минимально возможное значение p_2 при ограничении на затраты max $R_н = R_н^{доп}$).

Как указывалось выше $p_1+p_2+p_3=1$. В современных системах защиты от НСД $p_2=0$ изначально. Это облегчает нарушителю решение задачи максимизации p_1, так как в этом случае

$$\mathbf{p_1+p_2=1} \qquad (1.3)$$

Таким образом меняется смысловое значение, вкладываемое в p_1. Если в случае (1.2) p_1 - это вероятность осуществления НСД и того, что полученная в результате НСД информация подлинна, то в случае (1.3) это просто вероятность успешного осуществления НСД.

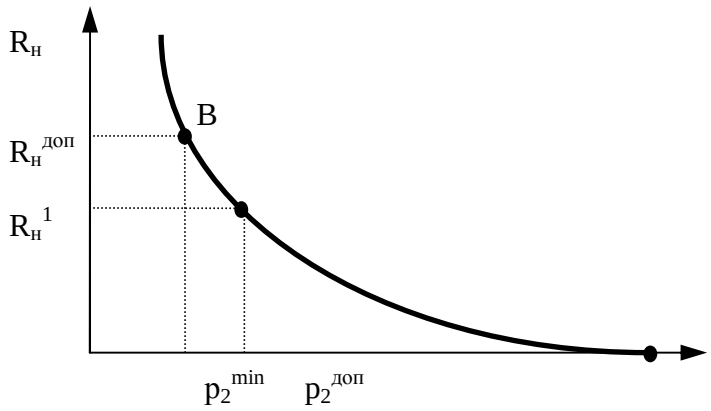


Рис.1.11. Зависимость затрат $R_н$ от вероятности p_2

То есть в случае (1.3) если нарушителю удалось осуществить НСД, то он с вероятностью **"1"** может считать, что полученная информация подлинна. При этом нарушитель направляет все затраты $R_н \le R_н^{доп}$ на преодоление средств защиты от НСД.

Поэтому вероятность p_2 выступает в качестве резерва защиты. И именно в этом направлении необходимо совершенствование современных систем защиты информации (в нашем случае СИП), так как, зная, что $p_2 \neq 0$, противник, получив даже требуемую достоверную информацию I, вынужден будет прибегнуть к ее тщательной проверке, что также приведет к росту $R_н$.

Основными атрибутами блока "Противник" являются: субъект-противник, угроза, модификация угрозы, канал, модификация канала, результат проявления угрозы.

Для блока "Противник" прямое действие функции связи идет по цепи: Противник - Угроза - Канал - Объект атаки (Пользователь, информационные ресурсы или СЗИ). Цель связи - достигаемый результат, ради которого и формулируется стратегия действия противника.

Цепь обратной связи, вносящая соответствующие коррективы в действия противника: Объект атаки (защиты) – Результат проявления угрозы – Модификация канала.

Очевидно, что для правильной организации действий СЗИ по осуществлению информационного противодействия противнику необходимо классифицировать его несанкционированные действия в СОИ. Такая классификация может быть проведена на основе предварительного анализа внешнего проявления (следов) атаки Противника. Внешние проявления атаки в дальнейшем будем называть спецификацией НСД противника.

Выбор того или иного варианта атаки (стратегии) Противником может зависеть от содержания и характера некоторой начальной утечки информации. В качестве начальной утечки информации можно рассматривать, например, соответствующие ключевые слова, позволяющие ассоциативно судить о защищаемой информации, и частоту их повторения.

Блок ***«Система защиты информации».*** Стратегия СЗИ – ведение информационной борьбы для достижения целей оборонительных действий в СОИ.

Примерами частных стратегий СЗИ могут быть обеспечение требуемой защищенности информации при поддержании заданного уровня параметров, характеризующих установленный статус ее хранения, обработки и использования в заданном временном интервале; снижение времени бесконтрольного присутствия противника в СОИ; дезинформация противника и т.д.

В [98,74] приведены семь функций, которые должна выполнять система защиты информации. А именно:

- предупреждение условий, порождающих дестабилизирующие факторы (ДФ) (угрозы информации);

- предупреждение проявления ДФ;
- обнаружение проявления ДФ;
- предупреждение воздействия ДФ на информацию;
- обнаружение воздействия ДФ на информацию;
- локализация воздействия ДФ на информацию;
- ликвидация последствий воздействия ДФ.

Ранее было отмечено, что в данной работе термин «СЗИ» остается традиционным, но вкладываемый в него смысл более широкий – система информационной борьбы.

В этом случае СЗИ свойственны также (кроме вышеназванных) функции по решению задач противодействия. Речь о них пойдет в следующем пункте работы.

В соответствии с указанными функциями СЗИ должна создавать и поддерживать активное функционирование средств и методов, реализующих: прогнозирование, обнаружение и подавление каналов атак на ИР, анализ и накопление статистики по угрозам и каналам, инициируемых Противником, интеллектуальное противодействие.

Основными атрибутами блока СЗИ являются: администратор сети и персонал службы защиты, технические средства защиты, программные средства защиты и противодействия, организационные средства защиты, морально-правовые средства защиты. Классификация методов и соответствующих им средств защиты приводится в работах [74, 89,104, 93]

Структурно-функциональные связи:

- «СЗИ – Средства и методы защиты и противодействия – Противник»;
- «СЗИ – Система активного поиска (обнаружения) – Противник»;
- «СЗИ – Система контроля доступа – Противник»;
- «СЗИ - Система контроля доступа – Пользователь».

Важным концептуальным положением для СЗИ является требование к ее адаптивности или способности к целенаправленному приспособлению при изменении характера внешних дестабилизирующих факторов и угроз, которое в МИБ реализуется путем своевременной модификации средств и методов информационной борьбы, выполняемой по прогнозу действий Противника или внешним проявлениям результатов его атаки. Причем указанное прогнозирование действий “Противника”, порождаемых им угроз и каналов, должно носить превентивный характер, обеспечивающий своевременную адаптацию и необходимый уровень защиты [67].

По внешним проявлениям результатов атаки в случае необходимости осуществляется классификация НСД и реализуется выработанная в оперативном режиме стратегия интеллектуального противодействия.

Таким образом, реализуется два контура функционирования СЗИ:

- классификация НСД;
- выбор стратегии противодействия.

Блок **“Пользователь”.** Стратегия действия “Пользователя” направлена на получение максимального эффекта от владения информацией при поддержании необходимого уровня ее защищенности в заданном временном интервале.

Основные атрибуты этого блока: субъект-пользователь, ТС хранения, обработки и использования информационных ресурсов, система контроля эффективности СЗИ. Наличие системы контроля эффективности СЗИ призвано гарантировать высокий уровень функционирования СЗИ.

Выбор того или иного пути атаки “Противником” зависит от реальной ситуации, в которой могут находиться: “Пользователь”, информационные ресурсы, СЗИ. Обычно ситуация определяется некоторым состоянием каждой из указанных подсистем в данном временном интервале. Например, характеристиками состояния “Пользователя” могут быть: квалификация, темперамент, привычки и т.п. Эти характеристики состояния “Пользователя” должны анализироваться СЗИ при контроле действий “Пользователя”, но, совершенно очевидно, что эти же характеристики будут находиться и в поле зрения “Нарушителя” при подготовке им атаки [67].

Блок **“Информационные ресурсы”.** Это информация, хранимая, обрабатываемая и передаваемая в СОИ. Субъекты ИБ в СОИ (противник и пользователи) взаимодействуют с ней в соответствии со стратегиями, сформулированными для них.

Блок **“Внешние факторы”**. Внешние факторы могут рассматриваться в двух аспектах [67]: природные факторы (типа природных катастроф), проявления которых неблагоприятны для всех субъектов данной модели и СОИ; факторы, связанные с надежностью функционирования СОИ (отказы, износ ТС и т.п.) и прогнозируемые в соответствии с теорией надежности.

В предлагаемой концептуальной модели очевидны следующие конфликтные взаимодействия: “Противник - Пользователь” и “Противник - СЗИ”. В обоих случаях - это состязательные задачи в условиях эволюции среды, применяемых субъектами и СЗИ средств и методов информационной борьбы.

Поэтому для построения и последующего анализа модели информационной борьбы можно воспользоваться методами теории игр или, что более перспективно – методами в рамках концепции искусственной жизни [53].

1.10. Основные цели и задачи информационной борьбы в корпоративной сети обмена информацией

Проанализировав реалии и перспективы информационных войн, отметив важность задач, стоящих перед сетью обмена информацией корпоративного масштаба, можно уверенно говорить о том, что информационная борьба с противником в СОИ может быть эффективна лишь в том случае, если в СОИ будут реализованы ***все*** три аспекта оборонительных инфор-

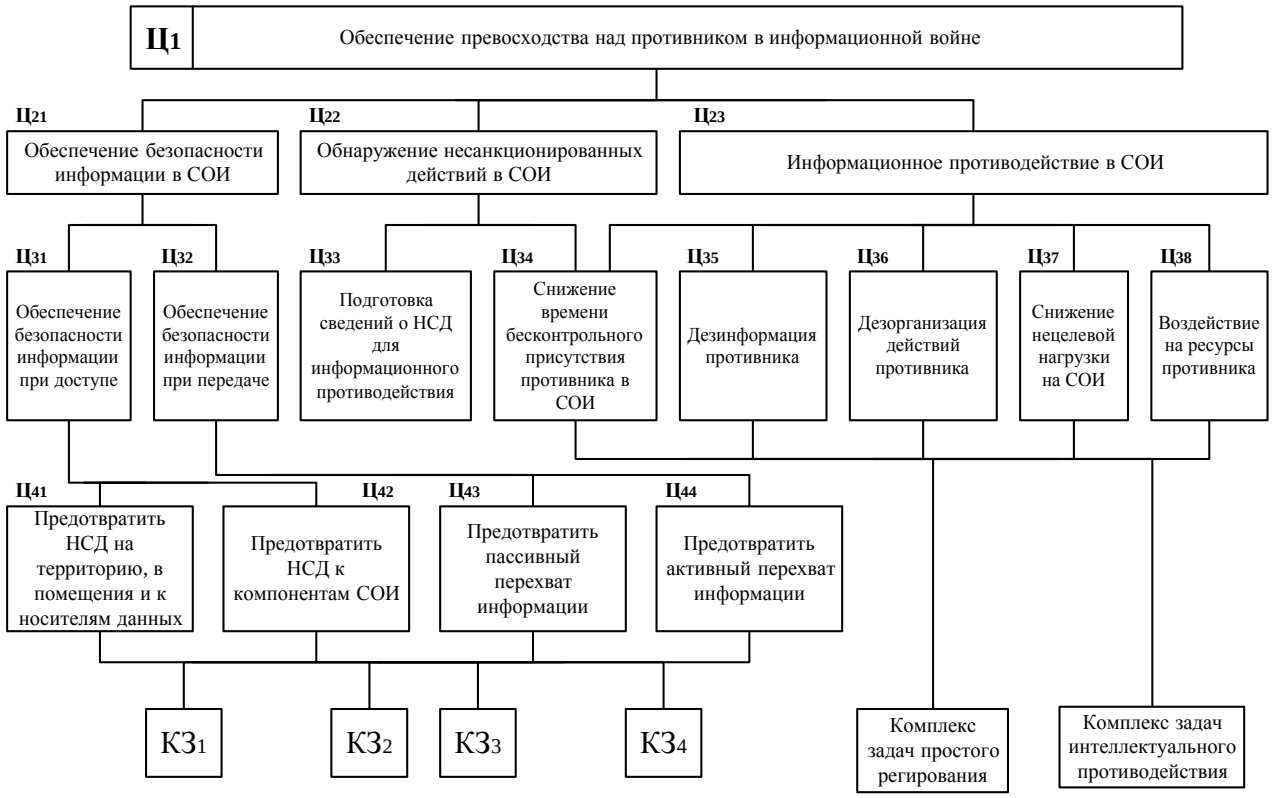


Рис. 1.12. Дерево целей и задач информационной борьбы

мационных боевых действий:

- ***обеспечение безопасности информации в СОИ;***
- ***обнаружение несанкционированных действий в СОИ;***
- ***информационное противодействие противнику в СОИ.***

Анализ информационных боевых действий, проведенный в данной главе, позволяет сформулировать главную цель $Ц_1$ информационной борьбы с противником в информационной войне.

Цель $Ц_1$ подчинена собственно цели функционирования СОИ $Ц_0$.

Используя методы системного анализа [66, 40], можно произвести декомпозицию цели $Ц_1$ на подцели.

Такая декомпозиция позволит сформулировать основные задачи (комплексы задач) ИБ в СОИ.

Представленное на (рис.1.12.) дерево целей и задач отражает результат декомпозиции цели $Ц_1$.

Задачи, решаемые в ходе информационной борьбы в СОИ, являются, безусловно, взаимосвязанными. Однако, исходя из полученных подцелей и содержания каждой из этих задач, все задачи можно объединить в отдельные комплексы задач (КЗ), которые связаны с разработкой и применением средств и механизмов ИБ с противником в СОИ.

Построив дерево целей и задач ИБ, становится очевидным, что недостаточно создать в СОИ систему защиты информации (СЗИ) с традиционным набором функций по обеспечению безопасности. В СОИ должна функционировать, так называемая, система информационной борьбы (СИБ), решающая все задачи безопасности, обнаружения НСД и информационного противодействия противнику в СОИ. В дальнейшем, используя термин СЗИ наряду с СИБ, мы будем вкладывать в него смысл именно полнофункциональной системы информационной борьбы.

На настоящий момент неисследованным остается комплекс задач *интеллектуального*

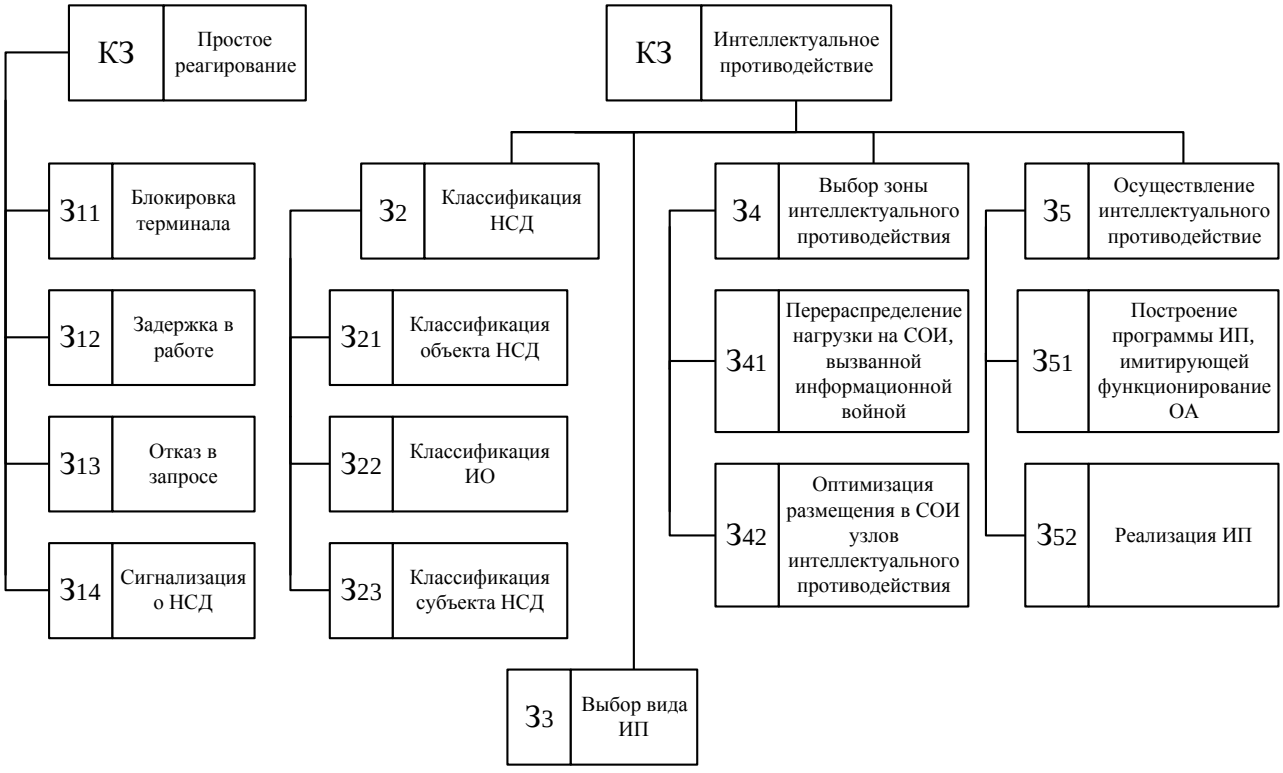


Рис. 1.13. Дерево целей и задач информационной борьбы (продолжение)

противодействия (рис. 1.13), что объясняет особое внимание к нему в данной работе. Мы выбрали этот комплекс задач в качестве основной области исследований в работе.

1.11. Постановка задачи исследования

Проведенный в настоящей главе анализ объекта исследований позволяет сделать вывод о том, что на настоящий момент информационная борьба в любой СОИ не может быть

эффективна ввиду отсутствия соответствующих методов подготовки и осуществления интеллектуального противодействия, учитывающих динамику информационной борьбы и цели функционирования СОИ.

Исходя из этого, сформулируем задачу исследования.

Исходные данные выбраны следующие.

Структура базовой СОИ, представленная в виде взвешенного неориентированного без петель графа **$G(N_1, N_2)$**, где N_1 – множество узлов (центров коммутации), N_2 – множество ветвей связи.

Матрица пропускных способностей ветвей связи $M_{N\times N} = \left\| \mu_{ij} \right\|$, где μ_{ij} – пропускная способность связи между узлами i и j.

В сети используется адаптивный алгоритм маршрутизации **V**.

Требования к качеству обслуживания заданы: требуемым временем доставки информации $\mathbf{t_{TP}}$, допустимой вероятностью потерь пакетов $P_{ПИ}^{доп}$; требуемой вероятностью своевременной доставки пакетов $P_{ПИ}^{TP}$.

Матрица интенсивностей потоков пакетов разных видов информации $\Lambda^{(k)}_{[N\times N]} = \left\| \lambda_{ij}^{(k)} \right\|$. Элемент $\lambda_{ij}^{(k)}$ является элементом матрицы $\Lambda^{(k)}_{<N\times N>}$ и представляет собой поток пакетов $x_{ij}^{(k)}$ от узла i к узлу j в дискретный момент времени $k \in \tau$ ($\tau = [1, n)$).

Матрица интенсивностей потоков пакетов типа Противник-Объект атаки (Пр-ОА) $\Lambda_{N\times N}^{ПрОА\,(k)} = \left\| \lambda_{ij}^{(k)ПрОА} \right\|$. Элемент $\lambda_{ij}^{(k)ПрОА}$ является элементом матрицы $\Lambda_{N\times N}^{ПрОА}$ и представляет собой поток пакетов типа Пр-ОА $x_{ij}^{(k)ПрОА}$ от узла i к узлу j в $k \in \tau$ момент времени.

Множество **M**={1,… i_M,…n_M} – множество типов стратегий противника, где n_M – число рассматриваемых типов стратегий противника.

Множество **L**={1,… i_L,…n_L} – множество реализуемых в СОИ типов стратегий ИП, где n_L – число типов стратегий ИП.

Результат работы системы контроля доступа в СОИ представлен кортежем спецификации НСД $z_i \in Z$, где Z={z_i} – множество возможных спецификаций НСД.

Множество R={R_j} – множество спецификаций протоколов ИП в СОИ.

Процесс информационной борьбы в СОИ можно охарактеризовать временем бесконтрольного присутствия противника в СОИ $\mathbf{T_{БП}}$.

Бесконтрольность присутствия противника означает, что системе, отвечающей за

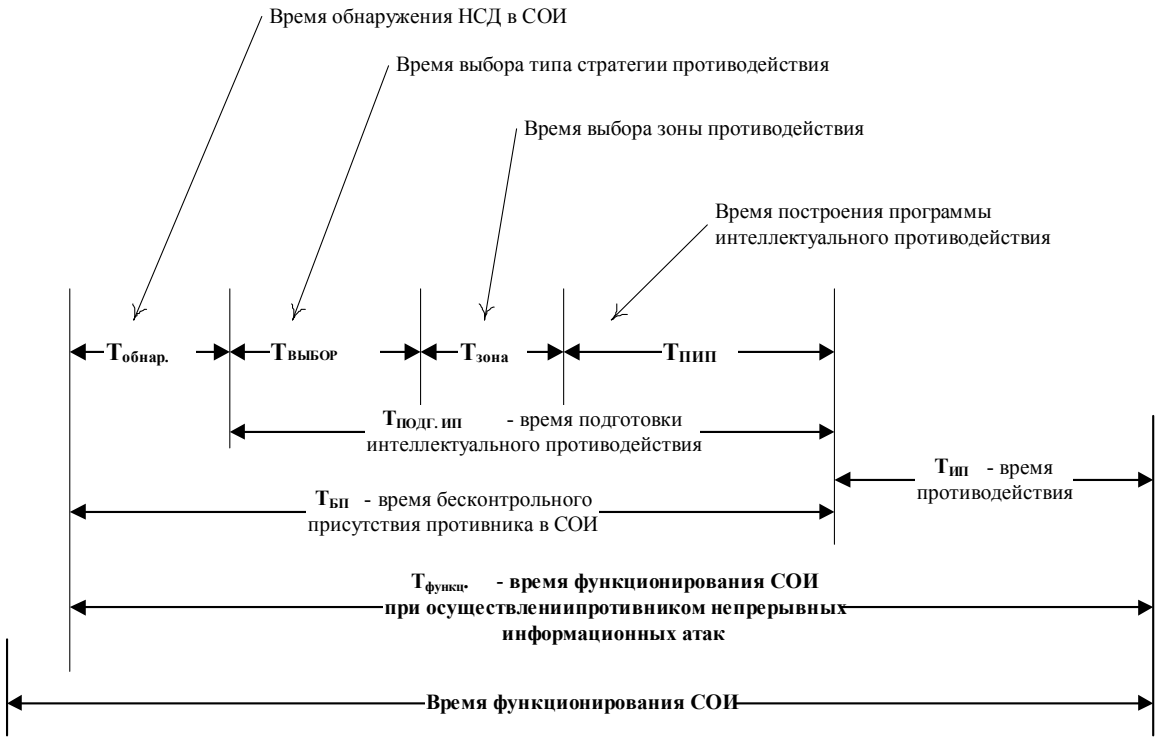


Рис. 1.14. Временная диаграмма этапов подготовки и осуществления интеллектуального противодействия в СОИ

информационную борьбу в СОИ (это может быть СЗИ, СИП), неизвестна стратегия действий противника или на действия противника не представляется возможным влиять в данный момент времени. С другой стороны, информационная борьба характеризуется нагрузкой на СОИ, связанной с действиями противника и описываемой матрицей интенсивностей потоков Противник-Объект атаки (Пр-ОА) $\Lambda_{[N\times N]}^{ПрОА}$.

Сделаем допущение, что в течение некоторого времени функционирования СОИ $\mathbf{T_{ФУНКЦ}}$ противник осуществляет непрерывные атаки на СОИ. При неудачной атаке противник использует другую стратегию или другие каналы реализации атаки.

Тогда время функционирования СОИ численно будет равно сумме времени бесконтрольного присутствия противника в СОИ $\mathbf{T_{БП}}$ и времени противодействия $\mathbf{T_{ПРОТИВОД}}$.

$$\mathbf{T_{ФУНКЦ\,СОИ} = T_{БП}+T_{ПРОТИВОД}}, \qquad (1.4)$$

где

$\mathbf{T_{ПРОТИВОД}}$ – время противодействия (реагирования), когда стратегия противника известна с той или иной степенью детализации и осуществляются выбранные виды реагирования.

Задача информационной борьбы состоит в том, чтобы уменьшить время бесконтрольного присутствия противника в СОИ $\mathbf{T_{БП}}$.

Временная диаграмма организации интеллектуального противодействия противнику в СОИ показана на (рис. 1.14.)

При этом время противодействия определяется следующим образом: в случае простого реагирования – временем задержки в работе средств контроля доступа $\mathbf{T_{ЗАДЕРЖ}}$; в случае интеллектуального противодействия – временем удержания противника на ложном объекте атаки (временем осуществления ИП $\mathbf{T_{ИП}}$).

В силу того, что в данной главе уже были показаны преимущества ИП по отношению к простому реагированию, а также того, что в большинстве случаев $\mathbf{T_{ИП}}$>>$\mathbf{T_{ЗАДЕРЖ}}$ в дальнейшем мы будем в качестве противодействия рассматривать именно ИП и полагать, что $\mathbf{T_{ПРОТИВОД}}$=$\mathbf{T_{ИП}}$.

Время бесконтрольного присутствия противника в СОИ выразим следующей формулой:

$$\mathbf{T_{БП}=T_{ОБНАР.}+T_{ПОДГ.\,ИП}}, \quad (1.5)$$

где

$\mathbf{T_{ОБНАР.}}$ – время обнаружения действий противника;

$\mathbf{T_{ПОДГ.\,ИП}}$ – время подготовки интеллектуального противодействия.

В свою очередь, время обнаружения действий противника запишется в виде:

$$\mathbf{T_{ОБНАР.}=t_{ОБН.}-t_{ДПр}}, \quad (1.6)$$

где

$\mathbf{t_{ДПр}}$- момент времени начала действий противника в СОИ;

$\mathbf{t_{ОБН.}}$ – момент времени обнаружения системой контроля доступа НСД в СОИ.

Тогда

$$\mathbf{T_{ФУНКЦ\,СОИ} = T_{ОБНАР.}+T_{ПОДГ.\,ИП}+T_{ИП}}, \quad (1.7)$$

Задача ИБ состоит в том, чтобы минимизировать $\mathbf{T_{БП}}$, то есть при $\mathbf{T_{ФУНКЦ.СОИ}}$=const минимизировать значения $\mathbf{T_{ОБНАР.}}$ и $\mathbf{T_{ПОДГ.\,ИП}}$ и максимизировать значение $\mathbf{T_{ПРОТИВОД.}}$.

При этом, как отмечалось выше, долговременное реагирование может быть реализовано за счет ИП, осуществляемого на базе подставляемого (ложного) объекта атаки. При этом решается задача максимизации значения $\mathbf{T_{ИП}}$ ($\mathbf{T_{ИП}}$ – время интеллектуального противодействия).

Решение задачи минимизации значения $\mathbf{T_{ОБНАР.}}$ сводится к задаче совершенствования систем контроля доступа в СОИ и в данной работе не рассматривается.

Решение задачи максимизации времени интеллектуального противодействия $\mathbf{T_{ИП}}$ осуществляется каждый раз при реализации программы интеллектуального противодействия, может иметь разные решения при одном и том же типе стратегии противника и одной и той же программе противодействия. Поэтому в данной работе эффективность интеллектуального противодействия оценивается через частные показатели эффективности подготовки противодействия противнику в СОИ.

В качестве одного такого показателя эффективности в работе рассматривается *время подготовки интеллектуального противодействия* – $\mathbf{T_{ПОДГ.\,ИП}}$.

А, ввиду того, что задачей интеллектуального противодействия является также снижение нагрузки на базовую сеть СОИ, связанной с действиями противника в ходе информационной борьбы и описываемой матрицей $\Lambda^{ПрОА}$, а также, учитывая то, что в нашей СОИ мы используем алгоритм адаптивной маршрутизации **V**, в качестве второго показателя эффективности интеллектуального противодействия выбрана *сумма коэффициентов недоиспользования пропускной способности СОИ* – D, которая определяется следующим образом:

$$D = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(1-\frac{\lambda_{ij}^{\sum}}{K \bullet \mu_{ij}}\right), \quad (1.8)$$

где

$\lambda_{ij}^{\sum}$ - суммарная интенсивность пакетов в ветви (i, j);

K – коэффициент, учитывающий степень готовности ветви (i, j) и вероятность воздействия противника на линию связи неинформационным путем.

Подготовка интеллектуального противодействия включает в себя три основные этапа:

- выбор вида ИП (выбор типа стратегии ИП);
- перераспределение нагрузки на СОИ (на этом этапе осуществляется выбор зон противодействия);
- построение программы ИП.

Тогда время подготовки интеллектуального противодействия запишем следующим образом:

$$\mathbf{T_{ПОДГ.\,ИП} = T_{ВЫБОР}+T_{D}+T_{ПИП}}, \quad (1.9)$$

где

$\mathbf{T_{ВЫБОР}}$ – время выбора вида ИП;

$\mathbf{T_D}$ – время решения задачи перераспределения нагрузки;

$\mathbf{T_{ПИП}}$ – время построения программы интеллектуального противодействия.

Так как $\mathbf{T_D} << \mathbf{T_{ВЫБОР}}$ и $\mathbf{T_D} << \mathbf{T_{ПИП}}$, то примем *допущение*, что показатели эффективности интеллектуального противодействия $\mathbf{T_{ПОДГ.\,ИП}}$ и D являются независимыми.

Исходя из сделанных допущений и выбранных показателей эффективности интеллектуального противодействия в СОИ, задача исследования состоит в следующем.

Необходимо разработать методику, повышающую оперативность подготовки ИП в СОИ

$$T_{ПОДГ.\,ИП} = \min_{i \in I}(F_i\ (Z_C, R_C)) \tag{1.10}$$

и осуществляющую перераспределение нагрузки $\Lambda^{ПрОА}$ на СОИ исходя из следующего критерия оптимальности

$$D = \max_{j \in J}(F_{D\,j}(G,\ \Lambda, \Lambda^{ПрОА}, i_L)) \tag{1.11}$$

при ограничениях

$$\left.\begin{array}{l} P_{ПИ\,ij}^{(k)} \le P_{ПИ}^{доп} \\ P_{СД\,ij}^{(k)} \ge P_{СД}^{TP} \end{array}\right\} \tag{1.12}$$

где

F_i – функция, определяемая способом подготовки ИП;

$z_C \in Z$ – спецификация НСД;

R_C – спецификация протокола ИП;

i_L – тип выбранной стратегии ИП;

$F_{D\,j}$ – функция, определяемая способом перераспределения нагрузки на СОИ.

1.12. Эффективность защиты от информационного нападения

Под эффективностью защиты информационных ресурсов от НСД будем понимать комплексное свойство процесса защиты информационных ресурсов от НСД, характеризующее его пригодность к решению стоящей перед ним задачи.

В процессе осуществления защиты от НСД в сети задействуются соответствующие средства и механизмы, совокупность которых образует систему защиты от НСД в СОИ (или система интеллектуального противодействия).

Эффективность системы интеллектуального противодействия определяется следующими свойствами:

1) результативность, характеризующаяся ее способностью давать целевой результат;

2) оперативность, характеризующаяся расходом времени, используемого на обнаружение НСД;

3) ресурсоемкость, характеризующаяся расходом сетевых ресурсов (времени занятия памяти специализированными программами, времени передачи служебных сообщений и т.п.) для обнаружения НСД и осуществления интеллектуального противодействия.

Свойство результативности количественно характеризуется вероятностью защиты информационных ресурсов в течение времени $t_з$ в условиях воздействия противника

$$W_1 = p(t > t_з), \tag{1.13}$$

$t_з$ - определяется временем старения информации.

Очевидно, что в период ведения боевых действий информация устаревает значительно быстрее, чем в мирное время.

Поэтому в современных условиях мирного времени, когда наибольшую угрозу представляют информационные войны, а не вооруженные конфликты, значение $t_з$ должно быть достаточно большим.

Другим показателем результативности процесса защиты информационных ресурсов от НСД является показатель безопасности $W_Б$:

Таким образом, результативность системы интеллектуального противодействия оценивается двумя показателями

$$W_Р = \|W_Б, W_1\|^T. \quad (1.14)$$

Оперативность такой системы оценивается промежутком времени

$$W_2 = t_2 - t_1 \quad (1.15)$$

t_1 - момент времени начала проверки прав доступа к информационным ресурсам;

t_2 - момент принятия решения о санкционировании доступа.

Ресурсоемкость системы интеллектуального противодействия W_3 характеризуется величиной суммарного времени выполнения всех операций по обнаружению НСД и осуществлению противодействия нарушителю (противнику).

Таким образом, эффективность системы интеллектуального противодействия оценивается векторным показателем эффективности W, который содержит четыре компоненты:

$$W = \|W_Б, W_1, W_2, W_3\|^T. \quad (1.16)$$

При построении системы интеллектуального противодействия, пригодной для практического использования, необходимо, прежде всего, обеспечить ее результативность, причем безопасность самой системы является определяющей. На показатель оперативности можно наложить ограничение, исходя из требований по оперативности доступа к данным в сети, а показатель ресурсоемкости, вследствие исключительной важности системы интеллектуального противодействия для нормального функционирования СОИ, можно рассматривать в последнюю очередь.

Следовательно, можно осуществить строгое ранжирование по важности элементов системы интеллектуального противодействия:

$$W_Б \succ W_1 \succ W_2 \succ W_3 \quad (1.17)$$

и решать задачу построения системы интеллектуального противодействия как задачу лексикографической оптимизации.

Проведенный анализ позволил нам сделать вывод о том, что существующие методы защиты информационных ресурсов от НСД в полной мере не отвечают современным требованиям к управлению защитой информации в компьютерных системах, не учитывают их масштабы, особенности построения, условия функционирования и, самое главное, не учитывают ***эволюцию*** информационного оружия.

Учитывая это, а также приведенные выше замечания по оцениванию эффективности системы защиты информации, нами была поставлена задача:

- разработать адекватную динамическую модель информационной борьбы в СОИ, способную к ***адаптации, самоорганизации и развитию с учетом особенностей и условий функционирования сети обмена информацией***;

- разработать эффективные методы защиты СОИ от НСД и методы противодействия

$$W_Б = \begin{cases} 1, \text{ если при выявленных ограничений и допущений нарушитель не в состоянии незаметно выявить и изменить параметры и алгоритм функционирования системы} \\ 0, \text{ в противном случае} \end{cases} \quad (1.18)$$

злоумышленнику в СОИ, учитывающие эволюцию методов и средств НСД к информационным ресурсам.

Вследствие того, что эффективность защиты информационных ресурсов СОИ от НСД определяется эффективностью функционирования системы интеллектуального противодействия и с учетом введенных в подразделе элементов системы интеллектуального противодействия и отношения предпочтения, задача построения такой системы для СОИ имеет следующий вид.

Пусть $U=\{u_1, u_2, ..., u_м\}$ - множество возможных и исследуемых систем интеллектуального противодействия (множество стратегий защиты информационных ресурсов СОИ от НСД). Каждая стратегия u_i характеризуется векторным показателем эффективности:

$$W(u_i) = \|W_Б(u_i), W_1(u_i), W_2(u_i), W_3(u_i)\|^T, \quad (1.19)$$

причем $W_Б(u_i)$ f $W_1(u_i)$ f $W_2(u_i)$ f $W_3(u_i)$, i = 1, M.

Необходимо определить стратегию $u^* \in U_д$, которая является решением лексикографической задачи оптимизации:

$$W(u^*) = \underset{u_j \in U_д}{\text{lexmax}}\, W(u_j), \quad (1.20)$$

при ограничениях $U_д$:

$$U_Б: \begin{cases} W_з(u_j)=1; \\ W_1(u_j) \geq W_1^{тр}; \\ W_2(u_j) \leq W_2^{тр}; \\ W_3(uj) \leq W_3^{тр}, j = \overline{1,M}, \end{cases} \quad (1.21)$$

где

$W_1^{тр}$ - требуемое значение вероятности защиты информационных ресурсов в течение заданного времени при осуществлении противником попыток НСД;

$W_2^{тр}$ - требуемое значение оперативности системы интеллектуального противодействия;

$W_2^{тр}$ - требуемое значение ресурсоемкости системы интеллектуального противодействия.

Подведем краткие итоги первой главы. Мы установили, что в настоящее время информационный ресурс играет первостепенную роль в развитии общества.

Наряду с этим, усиление информационной зависимости различных предприятий, многообразие информационного оружия, способов его применения делают реальной возможность ведения в СОИ любой корпорации необъявленной информационной войны. Наряду с этим показано, что большинство существующих СОИ не приспособлены к ведению информационной борьбы с противником.

Для дальнейшего изложения материала кратко представлена классификация информационного оружия, удаленных атак на сеть обмена информацией, которые являются наиболее вероятными при ведении информационных войн в сетях. Представлена концептуальная модель информационной борьбы в СОИ.

Построено дерево целей и задач информационной борьбы в сети обмена информацией, показана необходимость осуществления в СОИ интеллектуального противодействия противнику.

Описан комплекс задач интеллектуального противодействия противнику в СОИ. Предложен подход к осуществлению интеллектуального противодействия, основанный на использовании объекта СОИ (ложного объекта атаки), имитирующего вид и/или функционирование объекта СОИ, выбранного противником для атаки (объекта атаки).

Глава 2. Анализ способов и методов информационной борьбы в корпоративной сети обмена информацией

2.1. Предполагаемая архитектура корпоративной сети телекоммуникаций

В своей работе по разработке методов интеллектуального противодействия в информационной войне мы опирались на то, что некая корпорация к две тысяче какому-нибудь году пытается создать всеобъемлющую телекоммуникационную сеть, связав массу своих филиалов по всему свету в единый кибернетический организм.

Нам показалось, что телекоммуникационная сеть может быть представлена примерно

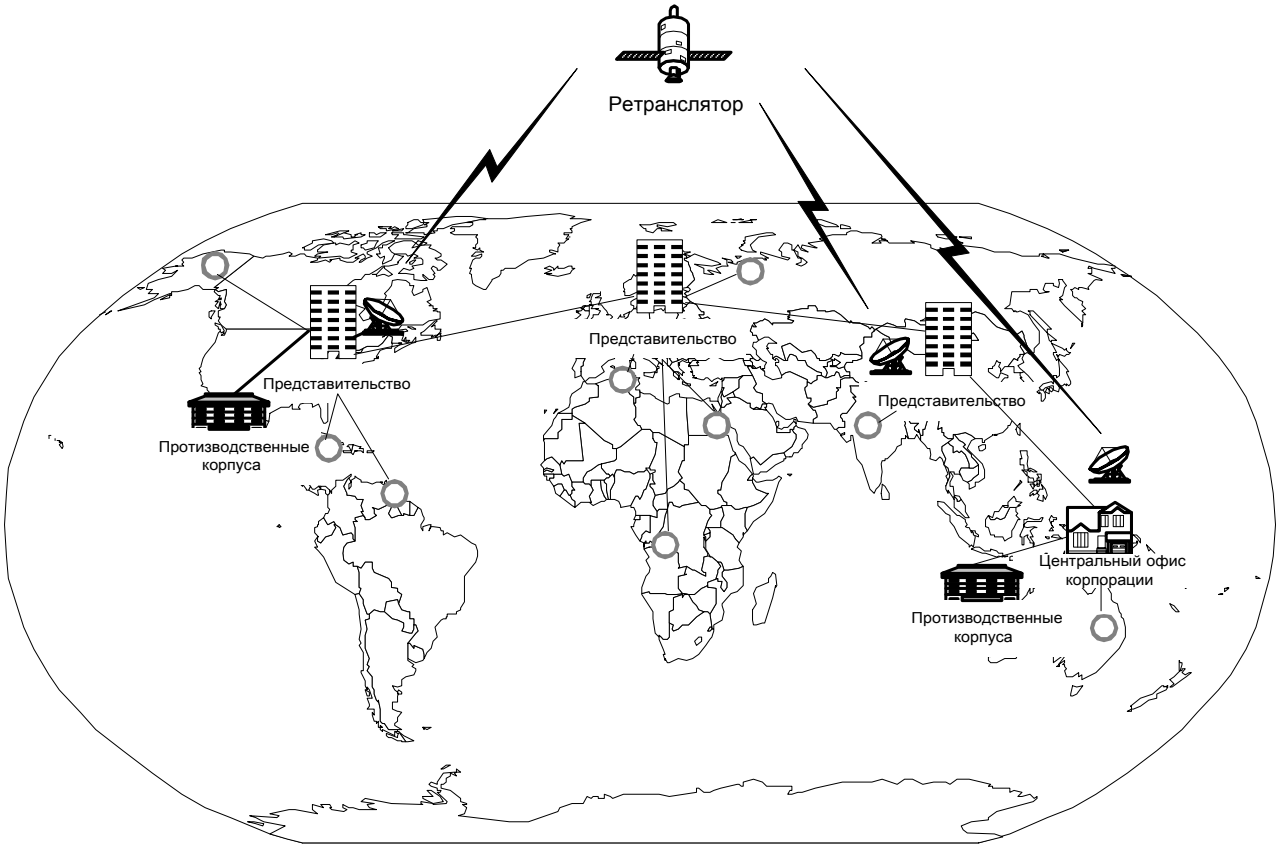


Рис. 2.1. Структурная схема телекоммуникационной сети корпорации

в следующем виде (рис. 2.1.).

Наша корпорация занимается производством какого-то очень важного продукта, который к тому же пользуется бешеным спросом по всему миру.

Главный офис корпорации находится на одном из тысяч маленьких островков Океании. Он должен быть связан с представительствами корпорации во всех частях света при помощи оптоволоконных линий связи через Internet. Могут использоваться и космические спутники-ретрансляторы.

Корпорация весьма прогрессивная. Планируется, что практически все программное обеспечение в информационной инфраструктуре корпорации будет создано по трехуровневой технологии «клиент сервер». Базы данных корпорации значительно децентрализованы по региональным представительствам.

Сотрудники в отделах продаж и на складах готовой продукции будут активно использовать персональные электронные менеджеры типа Palm Pilot.

Так как наша корпорация развивается в 21 веке, а бумажные деньги к этому времени практически вышли из употребления, то маркетинговая стратегия уже давно строится на организации продаж в Internet-магазинах.

Основные активы корпорации размещены в виртуальных Internet-банках, а ее представители - интеллектуальные программы-агенты, принимают активное участие в торгах на виртуальных биржах [12, 48, 49, 73].

Фрагмент сети корпорации в одном из ее региональных представительств мог бы выглядеть примерно так (рис. 2.2.).

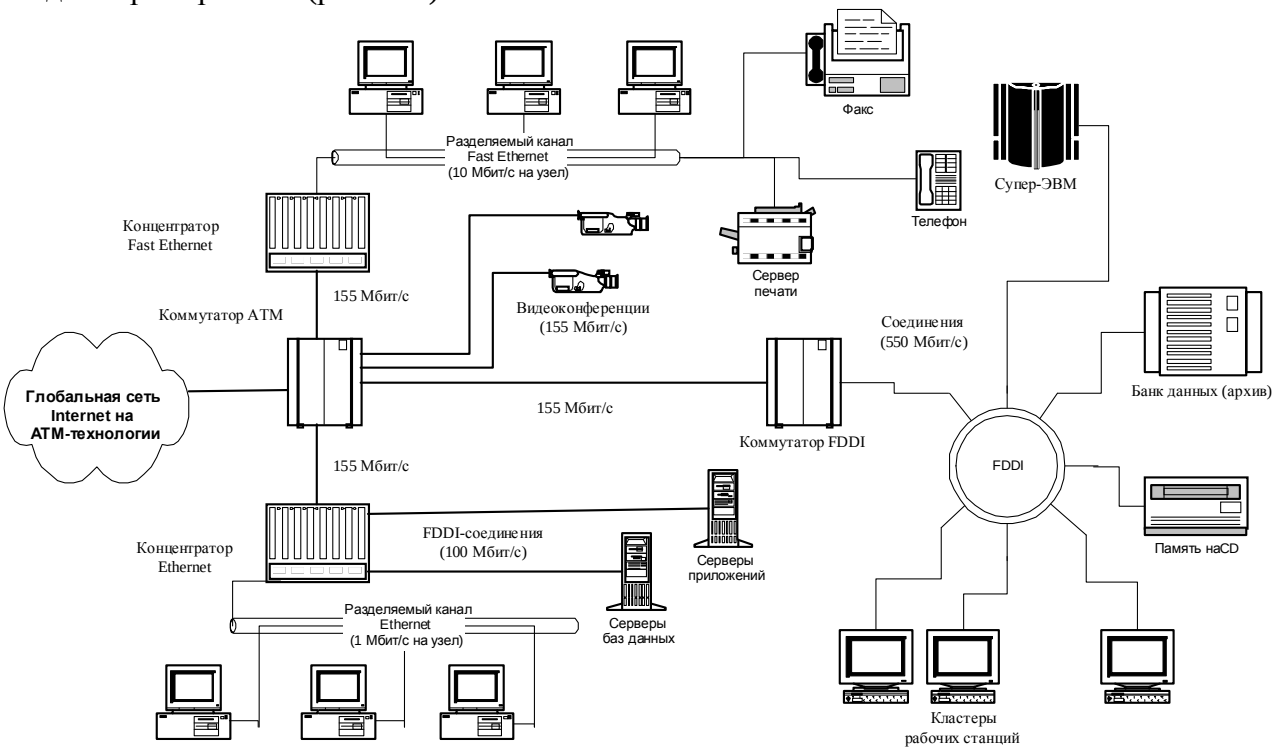


Рис. 2.2. Фрагмент сети корпорации в одном из региональных представительств

Вместе с тем, пока наша корпорация процветала и умножала свой капитал за счет рационального управления бизнес-процессами посредством развитой телекоммуникационной инфраструктуры, ее конкуренты не спали.

Они, видя головокружительный успех корпорации, решили, во что бы то ни стало помешать ей. Их коварный план строился на разрушающем (и, возможно, тайном) воздействии в одном из наиболее узких мест бизнеса корпорации – на широкой и развитой сети обмена данными.

К сожалению, методы защиты информации в корпоративных сетях к тому времени не получили столь бурного развития, как технологии электронной коммерции (по нашим представлениям). Все ждали появления методов интеллектуального противодействия в информационной борьбе...

Остановимся подробнее на информационных ресурсах корпорации и попытаемся выяснить: на что в первую очередь будет направлен удар противника?

2.2. Состав, структура и особенности функционирования корпоративной информационно-вычислительной сети

Как правило, телекоммуникационная инфраструктура любой корпорации предназначена для управления рядом бизнес-процессов в своей области деятельности. В состав нашей корпорации входят региональные представительства, заводские и складские корпуса, огромная сеть клиентов по всему миру. Для упрощения организации управления всей массой подчиненных подразделений в головных отделениях корпорации созданы пункты управления.

В соответствии с целевым назначением и местом в иерархической системе на пунктах управления размещаются комплексы средств автоматизации, узлы связи, специальные и тех-

нические системы, обеспечивающие деятельность органов управления в лице совета директоров региональных представительств по решению возлагаемых на них задач.

На пунктах управления органами управления осуществляется:

- прием, обработка, документирование приказов, поступивших с пунктов управления высших уровней, доведение их до подчиненных;
- сбор и обработка информации о состоянии рынка сбыта, состоянии подчиненных структурных подразделений, состоянии и инициативах конкурентов;
- оценка обстановки на мировом и региональном финансовом рынках;
- планирование инвестиций и развития производства;
- формирование и выдача приказов подчиненным силам и средствам; контроль исполнения приказов.

Состав решаемых задач практически единый для всех региональных пунктов управления. Отличия определяются характером задач, спецификой местного рынка и политической ситуации.

Автоматизированная система управления (АСУ) бизнес-процессами корпорации представляет собой взаимосвязанную общими целями и алгоритмами функционирования совокупность комплексов средств автоматизации (КСА) пунктов управления региональными отделениями корпорации, объединенных средствами связи и передачи данных. АСУ корпорации предназначена для обеспечения автоматизированного управления отделениями корпорации на местах.

Как отмечалось ранее, одним из основных элементов системы управления нашей корпорацией, обеспечивающим динамичность и оперативность процессов управления, является ИВС, объединяющая следующие элементы:

- сеть обмена данными;
- локальные вычислительные сети (ЛВС) и КСА региональных представительств, подключенных к базовой сети;
- подсети и КСА предприятий и складов готовой продукции;
- подсети рабочих групп в рамках подразделений;

С ростом требований по управлению растут требования к качественным и количественным характеристикам информационного обмена. Поэтому перспективным видится переход от прямых трактов информационного обмена между подразделениями корпорации к созданию сети обмена данными с коммутацией пакетов сообщений и использование при построении ИВС современных высокоскоростных технологий, таких как FDDI, FDDI II, 100-BASE X Ethernet, 100-BASE VG AnyLAN, ATM, Fiber Channel и Gigabit Ethernet [47, 65, 70].

К особенностям функционирования перспективной ИВС корпорации (той, что мы описали чуть выше) можно отнести:

- *разнородность технических средств и программного обеспечения;*
- *высокая доля разнородных сетевых информационных технологий;*
- *расширение спектра услуг и, как следствие, увеличение объема и качественного разнообразия информации, хранимой, обрабатываемой и передаваемой в ИВС;*
- *увеличение числа пользователей ИВС.*

Для перспективной ИВС любой корпорации может стать актуальным следующее:

- *сопряжение с глобальными сетями типа Internet;*
- *сопряжение с сетями других корпораций в результате расширения международного сотрудничества.*

Перечисленные особенности корпоративной ИВС, обуславливают высокую степень уязвимости информации, хранимой, обрабатываемой и передаваемой в такой сети.

Очевидно, что при создании единой ИВС корпорации в ее системе защиты информации сразу не удастся учесть все способы и возможные каналы информационного воздействия на сеть. Учитывая то, что во многих иностранных государствах и, прежде всего, в США глубоко осознали важность информационных ресурсов при ведении боевых действий, и значительное внимание уделяется вопросам ведения информационных войн, разрабатываются новые виды информационного оружия, в ИВС должны функционировать эффективные меха-

низмы противодействия информационным воздействиям со стороны противника на сеть. В противном случае, ИВС будет не в состоянии обеспечивать выполнение системой управления корпорации своих функций и последняя просто потерпит крах.

Понимая невозможность перекрытия всех атак противника, прежде всего ввиду значительных размеров нашей ИВС противодействие должно быть построено таким образом, чтобы максимально долго контролировать противника при положительном результате обнаружения его атаки. То есть, зная стратегию действий противника, необходимо прибегнуть к имитации ожидаемых противником результатов атаки. Чем успешнее будет построена имитация, тем меньше вероятность того, что противник прибегнет к другой стратегии информационного воздействия, которую система защиты информации ИВС может и не выявить.

Таким образом, создание эффективных методов и алгоритмов противодействия противнику в ИВС является актуальной задачей, ориентированной на ближайшую перспективу информационных войн.

2.3. Анализ особенностей информационной борьбы в сети обмена информацией. Роль информационных ресурсов в корпоративной ИВС

Для проведения комплекса мероприятий, направленных на создание, развертывание и эффективное функционирование упомянутой выше системы управления корпорацией, необходимы соответствующие затраты, называемые ресурсами управления. Одним из видов ресурсов управления являются информационные ресурсы (ИР) ИВС, без использования которых управление просто невозможно [90,26].

Применительно к рассматриваемой нами ИВС основными информационными ресурсами являются поля данных, записи в базах данных (БД), файлы, программы, пакеты прикладных программ (ППП) автоматизированных рабочих мест (АРМ) сотрудников, магнитные и оптические носители информации и др.[26]. Одним из ключевых видов ИР в ИВС являются базы данных (БД) различного назначения.

Самыми важными информационными ресурсами ИВС являются парольная информация, ключи шифрования и вся ключевая информация в целом. Доступ противника к этому виду ИР делает беспрепятственным его доступ ко всем другими, перечисленным выше ИР ИВС.

Важной информацией, хранимой, обрабатываемой и передаваемой в ИВС является информация о клиентах корпорации.

Особо следует выделить *специальную информацию* (разведывательная информация о конкурентах, информация о некоторых научных экспериментах в исследовательских подразделениях корпорации и т.п.).

Кроме этого в ИВС циркулирует большое количество информации, связанной с управлением подчиненными подразделениями. В перспективе при объединении многочисленных локальных сегментов ИВС в единую сеть эти потоки могут значительно возрасти.

Информация в ИВС подлежит защите от угроз информационного воздействия противника (угрозы конфиденциальности, целостности и доступности информации). При этом заметим, что защите подлежит не всякая информация, а только та, которая имеет цену.

Для оценки требуется распределение информации на категории не только в соответствии с ее *секретностью*, но и *важностью*.

По *уровню важности* информацию, хранимую, обрабатываемую и передаваемую в ИВС, можно разделить следующим образом.

Жизненно важная информация, наличие которой необходимо для функционирования системы управления корпорации.

Важная информация, то есть информация, которая может быть заменена или восстановлена, но процесс восстановления очень труден и связан с большими затратами.

Полезная информация - информация, которую трудно восстановить, однако система управления может эффективно функционировать и без нее.

Несущественная информация - информация, которая больше не нужна.

На практике отнесение информации управления к одной из этих категорий может представлять собой очень трудную задачу, так как одна и та же информация может быть использована многими подразделениями, каждое из которых может отнести эту информацию к различным категориям важности. Категория важности информации управления обычно изменяется со временем. Причем в обычное время информация «стареет» гораздо медленнее, чем во время различных кризисов, и, тем самым, требуется ее более надежная защита.

Отнесение информации управления к той или иной категории важности может и не быть согласованным с существующим принципом деления информации по уровням секретности.

Уровень секретности - это административная и законодательная мера, соответствующая мере ответственности лица за утечку или потерю конкретной секретной информации, регламентируемой специальным документом, с учетом государственных, военно-стратегических и служебных интересов.

Практика показала, что защищать необходимо не только секретную информацию. Несекретная информация, подвергнутая несанкционированным изменениям (например, модификация команд управления), может привести к утечке или потере связанной с ней секретной информации.

Суммарное количество, или статистика несекретных данных, в итоге может оказаться секретной. Аналогично сводные данные одного уровня секретности в целом могут являться информацией более высокого уровня секретности.

Традиционный путь, по которому идут разработчики систем защиты ИВС состоит из нескольких этапов:

- анализ возможных угроз безопасности ИР в ИВС;
- построение дерева целей и задач обеспечения безопасности ИР в ИВС;
- разработка методов, средств и механизмов обеспечения безопасности ИР в ИВС.
- разработка механизмов реагирования на несанкционированные действия.

Недостаток такого подхода очевиден. Если на ранней стадии разработки СЗИ не были учтены те или иные угрозы безопасности ИР или каналы их реализации, то велика вероятность того, что противник со временем обнаружит эти уязвимости СЗИ и проведет успешную информационную атаку на ИР. Этот вывод обуславливается тем, что в настоящее время используются лишь достаточно простые методы реагирования на НСД (блокировка терминалов, задержка в работе, отказ в запросе и т.п.), позволяющие противнику однозначно определить результат своей атаки. Эти методы реагирования лишь немного уменьшают время, в течение которого противник осуществляет подбор стратегии, которая закончится успешной информационной атакой. А так как именно время бесконтрольного присутствия противника в ИВС прямо влияет на вероятность успешной атаки с его стороны на ИР сети, то необходимы методы, которые позволят сократить это бесконтрольное время. Таких методов на настоящий момент не существует, при этом перспективным видится подход, заключающийся в имитации ожидаемого противником результата информационного воздействия на ИВС. Очевидно, что такая имитация должна быть дифференцированной, то есть учитывать:

- субъект атаки;
- объект атаки (важность, секретность ИР и другие его характеристики);
- цели информационной борьбы;
- цели управления бизнес-процессами.

Подобные методы противодействия могут осуществляться либо только высокопрофессиональным специалистом-экспертом, либо специалистом, использующим соответствующие средства искусственного интеллекта. Вовлечение в процесс реагирования на НСД человека и средств искусственного интеллекта делает этот подход противодействия противнику интеллектуальным.

Проанализировав имеющуюся в ИВС нашей корпорации информацию, единицы которой мы условились называть информационными ресурсами, становится очевидно, что в ИВС должно предусматриваться интеллектуальное противодействие противнику, которое бы за-

ключалось в нетривиальном реагировании на его действия с учетом множества факторов, возникающих в сети на конкретный момент времени.

Отойдем на время от вопроса создания систем интеллектуального противоборства в ИВС и попытаемся выяснить: а какие собственно методы используются для защиты информации в корпоративных сетях в настоящее время? Мы уже не раз говорили о том, что информационная война представляет собой новый вид боевых действий в киберпространстве с применением информационного оружия. Чтобы не плестись в хвосте и не латать дыры в системе безопасности информации уже ***после*** информационных ударов (от которых не всякая ИВС сможет оправиться) необходимо играть на опережение, необходимо, хотя бы приблизительно, знать какой удар ждать тебя в следующий момент.

2.4. Традиционные модели обеспечения безопасности

2.4.1. Модели разграничения доступа, построенные по принципу предоставления прав

В моделях разграничения доступа, построенных по принципу предоставления прав неформально право доступа может быть описано как “билет”, в том смысле, что владение “билетом” разрешает доступ к некоторому объекту, описанному в “билете” [86, 74].

Основными типами моделей, построенных на предоставлении прав, являются модели дискретного и мандатного доступа, которые используются в большинстве реальных систем, синтезированных в настоящее время.

В основе моделей дискретного доступа лежит представление системы защиты информации в виде некоторого декартова произведения множеств, составными частями которых являются соответствующие составные части системы защиты информации, например: субъекты, объекты, уровни доступа, операции. В качестве математического аппарата используется аппарат теории множеств. Представителем моделей дискретного доступа является одна из первых моделей безопасности - модель АДЕПТ-50 [86].

В модели АДЕПТ-50 представлено четыре типа объектов, относящихся к безопасности: пользователи (u), задания (j), терминалы (t) и файлы (f`), причем каждый объект описывается четырехмерным кортежем (A, C, F, M), включающим параметры безопасности.

Уровень безопасности (A) - скалярная величина - элемент из набора иерархически упорядоченных положений о безопасности, таких как: НЕСЕКРЕТНО, СЕКРЕТНО, СОВЕШЕННО СЕКРЕТНО.

Категория (C) - дискретный набор рубрик. Категории не зависят от уровня компетенции. В качестве примера можно привести следующий набор: {ОГРАНИЧЕННО, ТАЙНО, ТОЛЬКО ДЛЯ ПРОСМОТРА, КОММЕРЧЕСКИЙ, ПОЛИТИЧЕСКИЙ}.

Полномочия (F) - группа пользователей, имеющих право на доступ к определенному объекту.

Режим (M) - набор видов доступа, разрешенных к определенному объекту или осуществляемых объектом. Например: ЧИТАТЬ ДАННЫЕ, ПРИСОЕДИНЯТЬ ДАННЫЕ, ИСПОЛНИТЬ ПРОГРАММУ. Если U={u} обозначает набор всех пользователей, известных системе, а F(i) набор пользователей, имеющих право использовать объект i, то для модели, формулируются следующие правила:

- пользователь u получает доступ к системе тогда и только тогда, когда $u \in U$, где U - набор всех известных системе пользователей, имеющих на это право;

-пользователь u получает доступ к терминалу t тогда и только тогда, когда $u \in F(t)$, то есть в том и только в том случае, когда пользователь u имеет право использовать терминал t.

- пользователь u получает доступ к файлу f, тогда и только тогда, когда $A(j) \geq A(f)$, $C(j) \supseteq C(f)$, $M(j) \supseteq M(f)$ и $u \in F(f)$ то есть тогда и только тогда, когда:

- привилегии выполняемого задания шире привилегий файла или равны им;

- пользователь является членом F(f), то есть принадлежит к тем пользователям, которые имеют право использовать файл f.

Четырехмерный кортеж безопасности, полученный на основе прав задания, а не прав пользователя, используется в модели для управления доступом. Такой подход обеспечивает однородный контроль права на доступ над неоднородным множеством программ и данных, файлов, пользователей и терминалов. Например, наивысшим полномочием доступа к файлу для пользователя “СОВ. СЕКРЕТНО”, выполняющего задание с терминала “СЕКРЕТНО”, будет “СЕКРЕТНО”.

Еще одним типичным представителем моделей дискретного доступа является модель, называемая пятимерным пространством Хартсона [74]. В этой модели использует пятимерное пространство безопасности для моделирования процессов установления полномочий и организации доступа по ним.

Модель имеет пять основных наборов: A- набор установленных полномочий; U – набор пользователей; E – набор операций; R – набор ресурсов; S – набор состояний.

Область безопасности выглядит как декартово произведение:

$$A \times U \times E \times R \times S. \qquad (2.1)$$

Доступ рассматривается как ряд запросов, осуществляемых отдельными пользователями **u** для осуществления операции **e** над ресурсами **R** в то время, когда система находится в состоянии **s**.

Например, запрос на доступ представляется четырехмерным кортежем q=(u, e, R, s), где $u \in U$, $e \in E$, $s \in S$, $r \subseteq R$. Величины **u** и **s** задаются системой в фиксированном виде. Таким образом, запрос на доступ - подпространство четырехмерной проекции пространства безопасности. Запросы получают право на доступ в том случае, когда они полностью заключены в соответствующие подпространства.

Процесс организации доступа описывается алгоритмически следующим образом.

Для запроса q, где q(u, e, R, s), набора $\tilde{U}$ вполне определенных групп пользователей, набора $\tilde{R}$ вполне определенных единиц ресурсов и набора P правильных (установленных) полномочий, процесс организации доступа будет состоять из следующих процедур.

1. Вызвать все вспомогательные программы, необходимые для “предварительного принятия решений”.

2. Определить из $\tilde{U}$ те группы пользователей, к которым принадлежит u, затем выбрать из P те спецификации полномочий, которым соответствуют выделенные группы пользователей. Этот набор полномочий F(u) определяет привилегию пользователя u.

3. Определить из P набор F(e) полномочий, которые устанавливают e как основную операцию. Этот набор называется привилегией операции e.

4. Определить из P набор F(R) (привилегию единичного ресурса R) - полномочия, которые определяют поднабор ресурсов из $\tilde{R}$, имеющего общие элементы с запрашиваемой единицей ресурса R.

Полномочия, которые являются общими для трех привилегий в процедурах 2, 3, 4, образуют домен D(q), так называемый домен полномочий для запроса

$$q: D(q) = F(u)\ I\ F(e)\ I\ F(R).$$

5. Удостовериться, что запрашиваемый ресурс R полностью включается в D(q), то есть каждый элемент из R должен содержаться в некоторой единице ресурса, которая определена в домене полномочий D(q).

6. Осуществить разбиение набора D(q) на эквивалентные классы так, чтобы два полномочия попадали в эквивалентный класс тогда и только тогда, когда они специфицируют одну и ту же единицу ресурса. Для каждого такого класса логическая операция ИЛИ (или И) выполняется с условиями доступа элементов каждого класса. (Эффект обеспечивается тем, что это ИЛИ (или И) выполняется над всеми группами пользователей, в которые входит И).

Новый набор полномочий - один на каждую единицу ресурса, указанную в D(q), есть F(u,q) - фактическая привилегия пользователя u по отношению к запросу q.

7. Вычислить EAC - условие фактического доступа, соответствующего запросу q, осуществляя логическое И (или ИЛИ) над условиями доступа членов F(u, q). Это И (или

ИЛИ) выполняется над всеми единицами ресурсов, которые перекрывают единицу запрошенного ресурса.

8. Оценить условие фактического доступа и принять решение о доступе:

- разрешить доступ к R, если R перекрывается;
- отказать в доступе в противном случае.

9. Произвести запись необходимых событий. (Включение этого механизма в модель описано в работе [104]).

10. Вызвать все программы, необходимые для организации доступа после “принятия решения”.

11. Если решение о доступе, вытекающее из п. 8, есть “разрешить”, то выполнить все программы, “вытекающие из условия разрешить”. Если на шаге 8 принимается решение “отказать”, то выполнить все вспомогательные программы, “вытекающие из условия отказать”.

12. Если решение о доступе было “разрешить”, то завершить физическую обработку.

Реализации этой модели могут различаться, и приведенная здесь последовательность шагов не всегда является необходимой в полном объеме. Например, в большинстве реализаций выполнение шагов 2 и 6 осуществляется во время регистрации пользователя, обращающегося с терминала, в результате чего фактическая привилегия пользователя определяется на этом этапе.

К *достоинствам* моделей дискретного доступа можно отнести относительно простую реализацию. В качестве примера реализаций данного типа моделей можно привести так называемую матрицу доступа, строки которой соответствуют субъектам системы, столбцы - объектам, а элементы матрицы характеризуют права доступа.

К *недостаткам* относится “статичности” этих моделей. Это означает, что модель не учитывает динамику изменений состояний компьютерной системы и ИВС, не накладывает ограничений на состояния системы. В рассмотренной выше модели Хартстона одним из наборов пятимерного пространства является набор состояний S. При построении модели исходят из того, что S задано в фиксированном виде. Понятно, что число состояний в любой ИВС достаточно велико и полный набор S заранее сформировать невозможно. То есть это говорит о том, что хотя модель Хартстона и имеет теоретическую значимость, на практике ее использовать нецелесообразно. Такая модель может быть приемлема для отдельной компьютерной системы, но не для сети государственного масштаба.

Следствие указанного недостатка моделей, построенных на основе дискретной защиты, выражается теоремой.

Теорема 2.1. Не существует алгоритма, который может решить для произвольной системы дискретной защиты и общего права $r \in R$, является или нет заданная исходная конфигурация безопасной.

Доказательство этой теоремы приведено в работе [104].

Еще одним существенным недостатком моделей с дискретным доступом является нерешенная в них проблема троянских программ - одного из основных видов информационного оружия.

В отличие от дискретного доступа, который позволяет передавать права от одного пользователя другому без всяких ограничений, мандатный доступ накладывает ограничения на передачу прав доступа от одного пользователя другому. Это позволяет разрешить проблему троянских коней.

Классической моделью, лежащей в основе построения многих систем мандатного доступа и породившей большинство моделей этой группы, является модель Белла-Лападула.

Модель Белла-Лападула (БЛМ) до сих пор оказывает огромное влияние на исследования и разработки в области компьютерной безопасности.

Первое правило БЛМ, известное как правило “нет чтения вверх” (NRU), гласит: субъект с уровнем безопасности x_s может читать информацию из объекта с уровнем безопасности x_o, только если x_s преобладает над x_o. Это означает, что если в системе, удовлетворяющей правилам модели БЛМ субъект с уровнем доступа “секретный” попытается прочитать ин-

формацию из объекта, классифицированного как “совершенно секретный”, то такой доступ не будет разрешен.

Второе правило БЛМ, известное как правило “нет записи вниз”(NDR), гласит, что субъект безопасности x_s может писать информацию в объект с уровнем безопасности x_o только если x_o преобладает над x_s. Это означает, что если в системе, удовлетворяющей правилам модели БЛМ субъект с уровнем доступа «совершенно секретный» попытается записать информацию в неклассифицированный объект, то такой доступ не будет разрешен.

Правила запрета по записи и чтению БЛМ отвечают интуитивным понятиям того, как предотвратить утечку информации к неуполномоченным источникам.

Формализация БЛМ выглядит следующим образом:

S - множество субъектов;

O - множество объектов;

L - решетка уровней безопасности;

$F:S \cup O \rightarrow L$ - функция, применяемая к субъектам и объектам; определяет уровни безопасности своих аргументов в данном состоянии;

V - множество состояний - множество упорядоченных пар (F, M), где M - матрица доступа субъектов системы к объектам.

Система представляется начальным состоянием v_0, определенным множеством запросов R и функцией переходов $T:(V \times R) \rightarrow V$ такой, что система переходит из состояния в состояние после исполнения запроса. Сформулируем определения, необходимые для доказательства основной теоремы безопасности (ОТБ), доказанной для БЛМ.

Определение. Состояние (F, M) безопасно по чтению (RS) тогда и только тогда, когда для $\forall s \in S$ и для $\forall o \in O$, чтение$\in M[s, o] \rightarrow F(s) \geq F(o)$

Определение. Состояние (F, M) безопасно по записи (WS) тогда и только тогда, когда для $\forall s \in S$ и для $\forall o \in O$, запись$\in M[s, o] \rightarrow F(o) \geq F(s)$

Определение. Состояние безопасно тогда и только тогда, когда оно безопасно по чтению и по записи.

Определение. Система (v_0, R, T) безопасна тогда и только тогда когда состояние v0 безопасно и любое состояние, достижимое из v0 после исполнения конечной последовательности запросов из R безопасны в смысле предыдущего определения.

Теорема 2.2.2 (Общая теорема безопасности) Система (v_0, R, T) безопасна тогда и только тогда, когда состояние v_0 безопасно и T таково, что для любого состояния v, достижимого из v_0 после исполнения конечной последовательности запросов R безопасны, если $T(v, c)=v^*$, где $v=(F, M)$ и $v^*=(F^*, M^*)$ такие что для $s \in S$ и для $o \in O$

если чтение $\in M^*[s, o]$ и чтение $\notin M[s, o]$, то $F^*(s) \geq F^*(o)$;

если чтение $\in M[s, o]$ и $F^*(s)<F^*(o)$, то чтение$\notin M^*[s, o]$;

если запись $\in M^*[s, o]$ и запись $\notin M[s, o]$, то $F^*(o) \geq F^*(s)$;

если запись M[s, o] и $F^*(o)<F^*(s)$, то запись$\notin M^*[s, o]$.

Несмотря на все достоинства при использовании БЛМ на практике возникает ряд технических вопросов. Данные вопросы являются логическим следствием достоинства БЛМ - ее простоты. Проблемы возникают при рассмотрении вопросов построения политик безопасности для конкретных типов систем, то есть на менее абстрактном уровне рассмотрения. Как следствие, в мире компьютерной безопасности ведется широкая полемика по поводу применимости БЛМ для построения безопасных систем.

Из недостатков БЛМ, на наш взгляд, можно отметить то, что:

-на практике удаленное чтение в распределенных системах может произойти, только, если ему предшествует операция записи “вниз”, что является нарушением правил БЛМ;

- невозможно применить БЛМ для доверенных субъектов, которые могут функционировать в интересах администратора или представлять собой процессы, обеспечивающие критические службы, такие как драйвер устройства или подсистема управления памятью.

- проблема так называемой системы Z, подробно описанная в работе [11].

Отсутствие в модели БЛМ поддержки многоуровневых объектов (например: наличие несекретного параграфа в секретном файле данных).

Таким образом, недостатки БЛМ, являющиеся логическим следствием ее простоты, свидетельствуют о высокой степени абстрактности этой модели и неприменимости ее в ИВС для организации информационной борьбы.

С целью устранения ряда недостатков БЛМ при проектировании системы передачи военных сообщений Лендвером и МакЛином была разработана модель MMS [54].

В модели MMS используются следующие определения.

Классификация - обозначение, накладываемое на информацию, которое отражает ущерб, который может быть причинен неавторизированным доступом; включает уровни: СОВ. СЕКРЕТНО, СЕКРЕТНО и т.д. и множество разграничений ("ТАЙНО", "ЯДЕРНЫЙ" и т.д.). Множество классификаций и отношение между ними образуют решетку.

Степень доверия пользователю - уровень благонадежности персоны. Каждый пользователь имеет степень доверия, и операции, производимые системой для данного пользователя, могут проверить степень доверия пользователю и классификацию объектов, с которым он оперирует.

Пользовательский идентификатор - строка символов, используемая для того, чтобы отметить пользователя системы. Для использования системы пользователь должен предъявить ей пользовательский идентификатор, и система должна провести аутентификацию пользователя. Данная процедура называется *login*. Каждый пользователь иметь уникальный идентификатор.

Пользователь - персона, уполномоченная для использования системы.

Роль - работа, исполняемая пользователем (например: пользователь, имеющий право удалять, распространять или понижать классификацию объектов). Пользователь всегда ассоциирован как минимум с одной ролью из нескольких в данный момент, и он может менять роль в течение сессии. Для действия в данной роли пользователь должен быть уполномочен. Некоторые роли могут быть связаны только с одним пользователем в данный момент времени (например, распространитель). С любой ролью связана способность выполнения определенных операций.

Объект - одноуровневый блок информации. Это минимальный блок информации в системе, который имеет классификацию. Объект не содержит других объектов; он не многоуровневый.

Контейнер - многоуровневая информационная структура. Имеет классификацию и может содержать объекты (каждый со своей классификацией) и (или) другие контейнеры. Файл - это контейнер. Некоторые структуры файла могут быть контейнерами. Различие между объектом и контейнером базируется на типе, а не на текущем содержимом: если один из файлов данного типа является контейнером, то все остальные файлы данного типа являются контейнерами, даже если некоторые из них содержат только объекты или пусты. Устройства, такие как диски, принтеры, ленты, сетевые интерфейсы и пользовательские терминалы - контейнеры.

Сущность - объект или контейнер.

Требование Степени Доверия Контейнеров - атрибут некоторых контейнеров. Для некоторых контейнеров важно требовать минимум степени доверия, то есть пользователь, не имеющий соответствующего уровня благонадежности, не может просматривать содержимое контейнера. Такие контейнеры помечаются соответствующим атрибутом (CCR). Например, пользователь, имеющий степень доверия СЕКРЕТНО не может просматривать параграф СЕКРЕТНО сообщения, помеченного СОВ. СЕКРЕТНО, если оно содержится в CCR контейнере. Если пользователь должен иметь возможность просматривать данное сообщение, контейнер не должен быть помечен как CCR.

Идентификатор (ID) - имя сущности без ссылки на другие сущности. Например, имя файла есть идентификатор этого файла. Обычно все сущности имеют идентификатор.

Ссылка на сущность *Прямая*, если это идентификатор Сущности.

Ссылка на сущность *Косвенная*, если это последовательность двух или более имен Сущностей (из которых только первая - идентификатор). Пример: "текущее сообщение, первый абзац, вторая строка".

Операция - функция, которая может быть применена к сущности. Она может позволять просматривать или модифицировать сущность. Некоторые операции могут использовать более одной сущности (пример - операция копирования).

Множество Доступа - множество троек (Пользовательский идентификатор или Роль, Операция Индекс операнда), которое связано с сущностью. Операция, которая может быть специфицирована для особых сущностей зависит от типа данной сущности. Если операция требует более одного операнда, индекс операнда специфицирует позицию, на которой ссылка на данный операнд может появиться в операции.

Сообщение - особый тип, реализуемый в MMS. Сообщение является контейнером. Сообщение включает поля Куда, Откуда, Время, предмет, текст, автор.

Неформальная модель MMS выглядит следующим образом [54].

Пользователь получает доступ к системе только после прохождения процедуры login. Для этого пользователь предоставляет системе Пользовательский идентификатор, и система производит аутентификацию, используя пароли, отпечатки пальцев или другую адекватную технику. После успешного прохождения аутентификации Пользователь запрашивает у системы Операции для исполнения функций системы. Операции, которые Пользователь может запросить у системы, зависят от его ID или Роли, для которой он авторизован: с использованием операций пользователь может просматривать или модифицировать объекты или контейнеры.

Формализация модели MMS и ОТБ для нее приведены в [55].

Основной идеей проведенной в [55] формализации является взгляд на КС как на взаимоотношения между состояниями системы и системой.

И все-таки, модель MMS остается труднореализуемой на практике, статичной, и в значительной степени перегруженной абстрактным формализмом.

2.4.2. Вероятностные модели защиты информации

Модели этого типа исследуют вероятность преодоления систем защиты за определенное время Т. К *достоинствам* моделей данного можно отнести числовую оценку стойкости системы защиты. К *недостаткам* - изначальное допущение того, что система защиты может быть вскрыта. Задача модели - минимизация вероятности преодоления систем защиты.

Наиболее известными моделями, построенными на основе теории вероятностей, являются так называемая *игровая модель* и *модель системы безопасности с полным перекрытием*.

Игровая модель системы защиты строится по следующему принципу. Разработчик создает первоначальный вариант системы защиты. После этого злоумышленник начинает его преодолевать. Если к моменту времени Т, в который злоумышленник преодолел систему защиты, у разработчика нет нового варианта, система защиты преодолена. Если нет - процесс продолжается. Данная модель описывает процесс эволюции системы защиты в течение времени.

Может показаться, что игровая модель - это панацея от всех бед, применимая для обеспечения безопасности в ИВС, так как она учитывает эволюцию информационной борьбы. Но реально эта модель не показывает направлений совершенствования СИБ и не вырабатывает эффективные средства информационной борьбы. Она лишь констатирует факт, кто “выиграл”, а кто “проиграл” при имеющемся наборе средств информационной борьбы.

Система, синтезированная на основании модели безопасности с полным перекрытием, должна иметь, по крайней мере, одно средство для обеспечения безопасности на каждом возможном пути проникновения в систему [74].

В модели точно определяется каждая область, требующая защиты, оцениваются средства обеспечения безопасности с точки зрения их эффективности и их вклад в обеспечение безопасности во всей вычислительной системе. Считается, что несанкционированный доступ к каждому из наиболее защищаемых объектов О сопряжен с некоторой “величиной” ущерба для своего владельца, и этот ущерб может (или не может) быть определен количественно.

С каждым объектом, требующим защиты, связывается некоторое множество действий, к которым может прибегнуть злоумышленник для получения несанкционированного доступа к объекту. Можно попытаться перечислить все потенциальные злоумышленные действия по отношению ко всем объектам безопасности для формирования набора угроз T, направленных на нарушение безопасности. Основной характеристикой набора угроз является вероятность проявления каждого из злоумышленных действий. В любой реальной системе эти вероятности можно вычислить с ограниченной степенью точности.

Множество отношений объект-угроза образуют двухдольный граф, в котором ребро $<t_i, o_j>$ существует тогда и только тогда, когда ti ($\forall t_i \in T$) является средством получения доступа к объекту o_i ($\forall o_i \in O$). Связь между объектами и угрозами типа "один ко многим", то есть одна угроза может распространяться на любое число объектов, и объект может быть уязвим со стороны более чем одной угрозы. Цель защиты состоит в том, чтобы перекрыть каждое ребро графа и воздвигнуть барьер для доступа по этому пути.

Завершает модель третий набор, включающий средства безопасности M, которые используются для защиты информации в вычислительной системе. Идеально каждое средство m_k ($\forall m_k \in M$) должно устранять некоторое ребро $<t_i, o_j>$ из графа на рис. Набор M средств обеспечения безопасности преобразует двухдольный граф в трехдольный граф. В защищенной системе все ребра представляются в виде $<t_i\ m_k>$ и $<m_k\ o_j>$. Любое ребро в форме $<t_i\ o_j>$ определяет незащищенный объект. Одно и то же средство обеспечения безопасности может перекрывать более одной угрозы и (или) защищать более одного объекта. Отсутствие ребра $<t_i\ o_j>$ не гарантирует полного обеспечения безопасности (хотя наличие такого ребра дает потенциальную возможность несанкционированного доступа за исключением случая, когда вероятность появления ti равна нулю).

Таким образом, система с полным перекрытием - это система, в которой имеются средства защиты на каждый возможный путь проникновения.

Основное *преимущество* данного типа моделей состоит в возможности получения численной оценки степени надежности системы защиты информации.

Очевидным *недостатком* данного типа моделей является ее идеализированность, которая состоит в том, что считается заранее известным полный список угроз. Из этого недостатка вытекает второй, который заключается в запаздывании введения в системы защиты информации новых средств обеспечения безопасности при выявлении новых путей проникновения в систему.

К моделям, построенным на основе теории информации Шеннона, относятся модели невмешательства и невыводимости. Теория данных моделей бурно развивается в настоящее время.

Эти модели хорошо описаны в [91], частично устраняют недостатки рассмотренных нами ранее моделей, в частности БЛМ, но сами в свою очередь не лишены их.

Так или иначе, вопросы создания комплексных систем защиты информации в компьютерных системах и ИВС рассмотрены в работах [71, 51].

Из проведенного нами анализа этих и ряда других работ, а также описанных выше подходов к построению моделей безопасности нами был сделан вывод о том, что несмотря на обилие предлагаемых теоретических моделей безопасности, большинство из них оперируют достаточно абстрактными понятиями, способны моделировать системы защиты информации небольшой размерности, сложностью адаптации к изменяющейся обстановке в ИВС сравнима со сложностью их первоначальной разработки.

Таким образом, общие недостатки их таковы:

- если модели просты, то чаще всего они оперируют абстрактными понятиями, способствуют пониманию процесса защиты, но не применимы на практике, так как не позволяют учесть все факторы и угрозы;

- если мы имеем дело с серьезными разработками в области моделирования систем защиты информации, то они насыщены громоздким математическим аппаратом, трудны для восприятия и понимания. Это затрудняет и без того бесконечный поиск узких мест.

В качестве следующего общего недостатка можно также указать статичность моделей, их слабую масштабируемость и способность к адаптации в меняющейся обстановке в ИВС.

Совершенствование моделей систем защиты информации, их адаптация, повышение адекватности моделируемым системам носят во многом субъективный характер (непременным условием является участие человека).

Кроме того, в этих моделях заранее жестко определены направления защиты информации на основе анализа известных угроз, а также множества средств, операций и механизмов защиты информации в компьютерной системе и ИВС.

Одним словом, в моделировании системы защиты информации ярко выражена тенденция к формализму и определенному отрыву от практики. Поэтому на практике системы защиты строятся чаще всего интуитивно, а не на основе анализа соответствующих моделей. Это зачастую приводит к избыточности средств защиты на одном направлении и провалу на другом. Возникает проблема бесконечного “латания дыр” в системе защиты информации.

Стоит также отметить, что в условиях угрозы информационной войны недостаточным является создание пассивных систем защиты компьютерных систем и ИВС; понятие системы защиты информации должно трансформироваться в понятие систем интеллектуального противодействия и именно их, на наш взгляд, необходимо создавать. Системы интеллектуального противодействия должны включать в себя элементы и защиты, и интеллектуального противодействия вплоть до активного подавления ПЭВМ-нарушителя. Интеллектуальное противодействие особенно актуально для сетей обмена информацией государственных учреждений и силовых министерств.

Модели подобных систем информационной борьбы должны быть такими, чтобы их можно было внедрить в компьютерные системы с возможностью гибкого управления безопасностью в конкретных условиях функционирования.

В таком случае необходимо прибегнуть к новым нетрадиционным подходам в моделировании процессов обеспечения безопасности информации. Один из возможных подходов - это концепция “искусственной жизни”, представляющая собой так называемую четвертую волну искусственного интеллекта.

2.5. Анализ некоторых существующих систем защиты информации

Как уже отмечалось на протяжении всего предыдущего изложения материала, проникновения в корпоративную сеть извне и изнутри организации становятся, увы, привычным явлением и нередко способны привести к значительным финансовым потерям [74].

Растущий уровень сложности сетевых архитектур, повышение степени открытости сетей и все более тесная их привязка к *Internet* заставляют компании пересматривать свое отношение к обеспечению безопасности собственных сетей. Для предохранения от внешних атак в корпоративных сетях развертываются разнообразные защитные структуры. Самыми популярными среди них являются брандмауэры – системы, реализуемые, как правило, на программном уровне и формирующие барьер между *Internet* и внутренней сетью предприятия в точке их соединения.

Эти продукты образуют переднюю линию обороны и обычно являются первой системой, сопротивление которой пытается преодолеть противник. К сожалению, настройка брандмауэра – задача не из легких даже для опытных специалистов: в конфигурацию необходимо включить множество разнородных списков управления доступом, причем малейшая ошибка может обернуться зияющей брешью в защите. (Это касается, кстати, не только брандмауэров, ведь малейшей неточности в конфигурации маршрутизатора, сервера или коммутатора нередко бывает достаточно для возникновения лазейки.) К тому же брандмауэрные конфигурации необходимо регулярно обновлять, например, чтобы открыть доступ к новым сетевым службам или реализовать новые правила обеспечения безопасности, необходимость в которых возникла после структурной реорганизации.

Да и в правильно настроенном брандмауэре желающий практически всегда найдет

слабые места. Хакеры уже продемонстрировали миру свои способности в борьбе с системами данной категории (некоторые из методов проникновения описаны выше в этой книге). Профессионалы могут взломать большинство из ныне предлагаемых брандмауэров, а в отдельных случаях – просто обойти их [106].

Нередко взломы являются результатом внутренних атак на сеть, которые совершают чем-либо обиженные сотрудники самих организаций (по данным ФБР, примерно в 60% случаев корни компьютерных преступлений следует искать внутри предприятий).

Механизмы защиты часто интегрируются в операционные системы: это могут быть, например, алгоритмы аутентификации пользователей на базе паролей, средства шифрования, схемы многоуровневого управления доступом к данным. Однако и такая защита оказывается уязвимой. Операционные системы довольно трудно настраивать с точки зрения обеспечения защиты. Кроме того, программное обновление версии ОС может привести к появлению слабых мест, неизвестных администратору. Управление доступом на уровне ОС не всегда удается отобразить непосредственно на сетевой уровень, поэтому защитные функции системных средств управления доступом могут оказаться неэффективными в случае атаки на сеть.

Итак, традиционные службы сетевой защиты не всегда справляются с возложенными на них обязанностями; разрыв между политикой организации в области информационной безопасности и реальной практикой функционирования средств защиты становится все шире. В целях устранения этого несоответствия стали создаваться специальные системы, предназначенные для обнаружения попыток несанкционированного доступа в сеть извне и изнутри нее. Подобная система устанавливается на компьютерах, размещенных в стратегически важных контрольных точках сети, например у канала маршрутизатора, ведущего в Internet, или перед локальной сетью, содержащей ключевые корпоративные данные. Система ведет непрерывный мониторинг работы сети и анализирует такие параметры, как характеристики сетевого трафика, показатели использования центрального процессора и подсистемы ввода/вывода, динамика файловых операций, стремясь найти в этих параметрах признаки нарушения средств защиты. Попытки атак выявляются максимально оперативно; обнаружив нарушение, программа автоматически выполняет заранее предусмотренные действия, например ставит в известность об этом факте (по электронной почте или пейджеру) специалистов службы зашиты и сетевых администраторов. Кроме того, подозрительные операции могут регистрироваться в контрольном журнале. Некоторым системам даже удается разорвать TCP-соединение, установленное противником, и заменить его соединением администратора, который сможет выяснить, чем занимался хакер, и предпринять надлежащие меры. К сожалению, в таких системах отсутствует возможность работы с нарушителем в реальном времени посредством ложного объекта атаки.

Для отслеживания попыток взлома системы зашиты средства обнаружения сетевых атак используют различные методы, которые можно разбить на три группы: алгоритмы выявления статистических аномалий, алгоритмы обнаружения с использованием базы правил и алгоритмы, представляющие собой комбинацию первых двух типов [67, 63].

Средства обнаружения статистических аномалий выявляют нарушения системы зашиты путем анализа журналов мониторинга, отыскивая в них следы поведения пользователя и системы, которое отклоняется от некоторого шаблона. Предполагается, что подобное поведение должно указывать на несанкционированный доступ.

Чтобы реализовать такой алгоритм, прежде всего необходимо составить профили нормального режима работы пользователя и системы, которые служат статистической основой процедуры обнаружения. Для этого периодически изучается содержимое контрольных журналов и на его основе строятся модели событий, соответствующие типичным операциям. Любая последовательность событий, отклоняющаяся от сформированных профилей сверх определенной администратором меры, помечается как подозрительная ситуация.

Основное достоинство метода выявления статистических аномалий заключается в том, что в данном случае для эффективного отражения атак не обязательно заранее знать о возможных дефектах службы защиты сети. Кроме того, соответствующее программное обеспечение обладает достаточной адаптивностью, т. е. способно воспринимать новые модели

поведения пользователей и операционной среды.

Однако скрупулезная обработка данных мониторинга чрезмерно загружает системные ресурсы и потому отрицательно сказывается на общесистемной производительности. Кроме того, неаккуратное или не доведенное до конца составление профилей чревато ложными тревогами. Сами профили нуждаются в сопровождении; их следует периодически обновлять в соответствии с изменениями моделей поведения. Поэтому применение профилей оказывается не столь эффективным в динамических средах, например там, где сотрудники работают по свободному графику или часто перераспределяют ресурсы между проектами.

Наконец, не так-то легко установить оптимальный порог отклонения актуального поведения от действующего профиля, превышение которого должно рассматриваться как потенциальная атака. Если задать слишком низкий порог, это приведет к обилию ложных тревог, а слишком высокий барьер может помешать распознаванию настоящего нарушения.

Сейчас на рынке осталось мало систем обнаружения, основанных исключительно на статистическом анализе: они постепенно вытесняются продуктами, использующими базы правил, а также смешанными разработками, в которых выявление статистических аномалий служит лишь одним из механизмов распознавания сетевых атак.

Большую часть известных сегодня типов сетевых атак можно охарактеризовать некоей последовательностью событий. В системах обнаружения, основанных на своде правил, используются составленные в организации модели изменений системного состояния высокого уровня (событий мониторинга), которые имеют место во время атак, Такие модели образуют базы правил.

Системы обнаружения, опирающиеся на базу правил, работают аналогично антивирусным программам: они осуществляют мониторинг системных журналов и ресурсов в поисках модели, соответствующей какому-либо из известных профилей атаки. Имеющуюся базу правил необходимо постоянно обновлять, включая в нее впервые обнаруживаемые методы атаки. Поскольку подобные системы отслеживают только уже известные образцы атак, они почти не выдают ложных тревог. Однако здесь кроется и очевидный недостаток: эти продукты практически не могут распознавать новые методы нарушения служб защиты, которые бывают вполне «по зубам» приложениям, основанным на обнаружении статистических аномалий.

Раньше установка пакетов, использующих базы правил, приводила к снижению общесистемной производительности: сравнение содержимого контрольных журналов с профилями атак требовало множества ресурсов. Однако в последнее время появились технологии, позволяющие уменьшить потребности таких продуктов в системных ресурсах.

Некоторые фирмы предлагают продукты смешанного типа, пытаясь совместить в них достоинства систем двух описанных категорий и при этом избавиться от присущих им недостатков. Так, в гибридных системах с помощью базы правил можно выявлять известные варианты атак, а с помощью алгоритма статистических аномалий – атаки новых видов. Примером подобной системы может служить продукт Computer Misuse Detection System (CMDS) корпорации Science Application International Corporation (SAIC).

Система реального времени RealSecure [106], разработанная компанией Internet Security Systems основана на распределенной сетевой архитектуре и способна защитить корпоративные сети с самой сложной топологией. Данный продукт охватывает средствами безопасности всю организацию, предусматривая установку агентов-мониторов в наиболее критичных точках – в локальных сетях, между подсетями и даже в удаленных сетях, связанных через Internet. В основу системы положена технология интеллектуальных программных агентов [106]. Агенты анализируют проходящие по сети пакеты данных и ищут в них признаки внешней или внутренней атаки известных образцов, сверяясь с имеющейся базой данных. Обнаружив такие признаки, система реагирует одним из заранее выбранных способов.

Система RealSecure способна решать многие проблемы, стоящие перед защитными системами быстрого реагирования». Она может служить «второй линией обороны» сети, расположенной за брандмауэрами и выявляющей запрещенные действия, которые становятся возможными из-за неправильной конфигурации брандмауэров, маршрутизаторов или иных

устройств, Даже всплеск активности системы обнаружения станет индикатором сбоя брандмауэра, нарушения конфигурации или угрозы нападения. Развернутая в масштабах корпоративной сетевой среды, система RealSecure просматривает весь трафик, а не только проходящий сквозь брандмауэр; это позволяет бороться с атаками, инициируемыми изнутри сети, и иными нарушениями внутренних правил безопасности – обращениями к файлам паролей на компьютерах других сотрудников, несанкционированным чтением защищенной информации общего пользования и т. п. RealSecure способна декодировать многие сетевые протоколы и предоставляет пользователям возможность определять свои собственные события для любого соединения, чтобы выявлять даже те нарушения установленных правил, которые не относятся непосредственно к сфере зашиты данных (например, правил, ограничивающих число Web-сеансов в течение дня или резервирующих ресурсы для проведения видеоконференции с участием руководства компании).

Система Rea1Secure имеет распределенную архитектуру, в состав которой входит центральная административная консоль и процессоры (engines).

Процессор – это компонент, устанавливаемый в конкретном сегменте сети; он отслеживает сетевой трафик в режиме реального времени и при обнаружении события атаки реагирует соответствующим образом, Мониторинг ведется с использованием регулярно обновляемой базы данных, содержащей сведения обо всех известных на данный момент схемах атаки. Процессор может выявлять IP-атаки низкого уровня (IP Spoofing, IP Fragmentation, SYN Flood), предпринимаемые в обход брандмауэров с пакетными фильтрами, а также атаки высокого уровня, исходящие из служб Web, FTP, NFS, NIS или сеансов электронной почты. На сегодняшний день база данных RealSecure насчитывает 130 моделей атак.

На каждом процессоре администратор может установить специфические правила организации службы защиты, предусматривающие определенную реакцию на обнаруживаемые соединения в зависимости от типа пакетов, IP-адресов источника и назначения (или диапазона таких адресов), номеров портов, модели атаки и т. п. Столь гибкие возможности позволяют нужным образом настроить процессор на мониторинг конкретных хост-систем и сетей. Поскольку процессор относится к типу пассивных мониторов, он не наносит ущерба сетевой производительности.

В сети предприятия можно разместить любое число процессоров, не пропустив ни одного стратегически важного пункта сетевой топологии. Более того, в одной точке допускается параллельное функционирование нескольких процессоров; они могут обслуживать соединения с широкой полосой пропускания, такие как линии ТЗ, используемые для доступа к сети Intenet.

Канал связи между процессором и консолью защищен механизмом аутентификации, использующим пары «общий ключ/личный ключ», и средствами шифрования от RSA, базирующимися на интерфейсе Microsoft Cryptographic API (MCAPI).

Система RealSecure предлагает широкий выбор способов автоматического реагирования на обнаруженные атаки. Процессор может поступить следующим образом: – известить о факте сетевой атаки обслуживающий персонал через административную консоль, электронную почту или пейджер; – зарегистрировать событие в контрольном журнале с указанием даты, времени, источника, цели атаки и прочей сопутствующей информации. Можно запомнить и последовательность событий целиком, включая все нажатые хакером клавиши, а затем воспроизвести ее с консоли наподобие видеозаписи или напрямую «транслировать» атаку на консоль в режиме реального времени; – инициировать сценарий, составленный пользователем, с передачей контекстно-зависимых данных в качестве аргументов; – сгенерировать ловушку SNMP, информацию из которой можно оперативно передать на такие платформы администрирования, как HP OpenView, IBM NetView и Tivoli TME 10;

– перестроить конфигурацию брандмауэра. Система RealSecure поддерживает протокол Suspicious Activity Monitoring Protocol (SAMP) фирмы CheckPoint, встроенный, в частности, в брандмауэр Firewall-1 упомянутой компании; это позволяет в случае атаки мгновенно (за несколько миллисекунд) переориентировать брандмауэр на отклонение любого трафика, поступающего из атакующего узла, и тем самым предотвратить дальнейшие попытки не-

санкционированного доступа (длительность такого режима определяется пользователем);

– разорвать соединение, на котором зафиксирована атака. Процессор может аннулировать соединение как на стороне нарушителя, так и на стороне атакуемого компьютера; для этого он посылает в компьютер нарушителя пакет сброса (RST), а затем, имитируя IP-адрес хакера, посылает аналогичный пакет в атакуемую систему.

Всеми процессорами, установленными в сети, можно управлять из единого центра – с административной консоли. С ее помощью администратор формирует конфигурацию процессоров; здесь же осуществляется сбор и вывод данных о событиях, отмеченных процессорами. Процессор, обнаружив атаку, посылает сообщение о ней на административную консоль, которая отображает событие в той же последовательности, в какой оно происходило; при желании можно просмотреть порядок нажатых хакером клавиш и изображение на экране его монитора. События, связанные с нарушениями службы защиты, для удобства разделяются на группы по приоритетам (высокий, средний, низкий) и отображаются в соответствующих окнах,

О каждом событии можно получить дополнительную справку, содержащую его описание, возможные последствия и предлагаемые методы борьбы.

В организации можно установить более одной административной консоли; например, локальные консоли будут принимать данные о всех нарушениях для оперативного анализа и принятия ответных мер, а консоли вышестоящего уровня – только сообщения, относящиеся к наиболее серьезным нарушениям. В то же время один процессор имеет право передавать собранную информацию сразу на несколько консолей.

Для настройки конфигурации процессора на административной консоли необходимо задать типы событий, которые он должен распознавать, и способы реагирования на них. В системе RealSecure предусмотрена стандартная конфигурация, которая предлагается по умолчанию; она имеет некоторый уклон в сторону усиления мер безопасности. Кроме того, в комплект поставки входит несколько примерных вариантов конфигурации, обеспечивающих разные степени защиты; базируясь на них, администратор может построить схему защиты, наиболее точно отвечающую корпоративным потребностям. Предусмотрена также возможность маскировки трафика с помощью фильтров, позволяющая сконцентрировать усилия процессора на выявлении атак в наиболее критичном подмножестве входящего сетевого трафика.

Администратору предлагается не только установить автоматический режим реагирования, но и возможность самостоятельно, вручную проанализировать те или иные атаки, обращаясь для этого к возможностям графического пользовательского интерфейса. Например, он может запросить у процессора дополнительную информацию об атаке (источник пакета, его содержимое, включая заголовки сообщений электронной почты), зарегистрировать данное событие в контрольном журнале (если автоматическая регистрация событий этого типа не ведется) или уничтожить событие (если не задано автоматическое прекращение сеанса).

На основе содержимого журнальных файлов система RealSecure способна составлять подробные отчеты, которые отображаются на консоли в виде текста, гистограмм или в ином формате, определяемом администратором. Эти отчеты могут включать такие сведения, как объем данных, обработанных некоторым Wcb-сервером в течение дня, или число уничтоженных за день соединений и адреса их инициаторов. Подобные отчеты помогают оптимизировать функционирование службы зашиты сети.

Другой системой защиты этого же класса является система CMDS. Система выявления нарушений прав доступа к компьютерам (Computer Misuse Detection System, CMDS), предложенная корпорацией SAIC, предназначена для отслеживания случаев неадекватного использования компьютеров. Она ведет обработку неоднородных данных мониторинга поступающих из различных точек операционной среды, генерирует предупреждения (практически в режиме реального времени) и составляет графические отчеты.

Данный продукт способен вести мониторинг брандмауэров, ядра UNIX, СУБД и любых приложений, способных формировать контрольные данные. Программные компоненты CMDS защищают брандмауэры от взломщиков с внешней стороны, а также выявляют случаи

несанкционированной отправки ваших конфиденциальных данных в Internet. При разработке системы особое внимание было уделено возможностям переориентации на новые источники контрольных данных, имеющих различные форматы, и настройки конфигурации в соответствии с требованиями конкретной среды. В настоящее время сервер CMDS поддерживает такие платформы, как Sun C2, BSM, Solaris, Trusted Solaris, HP/UX, InterLock FireWall, Raptor FireWall, Windows NT, Data General DG/UX и др. Система допускает гибкое масштабирование: она может охватить своим контролем свыше тысячи компьютеров.

Программное обеспечение CMDS состоит из компонентов двух типов: серверного модуля CMDS Analyst Toolkit, управляющего мониторингом объектов защиты, и программных агентов CMDS Target, которые устанавливаются на каждой контролируемой станции и занимаются сбором и начальной обработкой данных. Архитектура серверного модуля разделяется на базовую (ядро) и настраиваемую части. Программы ядра реализуют механизмы обнаружения нарушений и транспорта данных, общие для разных платформ и сетевых протоколов; они отвечают за преобразование больших массивов данных мониторинга в содержательную поведенческую статистику, модели атак и графические отчеты о трендах. Настраиваемые функции предназначены для удовлетворения конкретных требований службы защиты в конкретной сети, например для поддержки оригинальных форматов контрольных данных, применения специальных алгоритмов обработки предупреждений, генерации кратких отчетов с уникальным содержанием.

CMDS поддерживает три механизма выявления нарушений: метод статистических отклонений, распознавание моделей атак и генерацию отчетов на основе анализа трендов. Для достижения большей гибкости эти элементы интегрированы параллельно-последовательным способом: параллельное объединение необходимо для того, чтобы каждый из механизмов мог выдать предупреждение независимо от двух остальных и в то же время служил источником информации для любого другого компонента. На пример, статистический процесс может предоставлять свои результаты в подсистему обнаружения моделей атак.

Статистический механизм выдает оператору предупреждение в том случае, когда профиль какого-либо пользователя, формируемый в режиме реального времени, отклоняется от ожидаемого профиля, который был рассчитан в начале сеанса работы данного пользователя. Статистический аппарат базируется на интуитивно формируемых категориях данных, например данных, связанных с сетевыми операциями, операциями выполнения или операциями просмотра. Каждая из этих категорий прослеживается в реальном масштабе времени, по мере обработки данных сервером CMDS. Для всякого Пользователя запоминается суточный профиль, отражающий его активность в каждой категории с часовым шагом. На основе суточных профилей за последние 90 дней вычисляется общий исторический профиль пользователя. Если его профиль за текущий день отклонится от исторического профиля в одной или нескольких категориях данных, на консоль может быть выдано предупреждающее сообщение.

В каждой категории ведется статистика двух типов: итоговое значение и процентное отношение итога по данной категории к общему числу записей мониторинга. Для каждого итогового значения категории вычисляется средняя (ожидаемая) величина и устанавливается пороговое значение, Под отклонением понимается превышение порогового значения категории в ее суточном профиле. Кроме того, для каждой категории вычисляется средний (ожидаемый) коэффициент с доверительным интервалом, который выражается в процентах и определяется отношением числа обработанных записей данной категории к общему числу полученных записей. Отклонением считается выход коэффициента суточного профиля в какой-либо категории за пределы ее доверительного интервала.

Распознавание моделей атак в CMDS основано на экспертной системе-оболочке CLIPS с прямым логическим выводом, разработанной в американском Национальном управлении по аэронавтике и исследованию космического пространства (NASA). База правил CLIPS также подразделяется на ядро и нестандартную часть. В ядро занесены общеизвестные атаки, которым могут подвергнуться любые среды (они классифицируются по нескольким основным категориям – вирусы, «черви», "троянские кони», преодоление парольной за-

щиты и др.). Нестандартная часть базы правил настраивается на последнем этапе выпуска продукта или – оператором CMDS – в процессе его эксплуатации в рамках общей настройки системы.

Фактологический материал вводится в экспертную систему по номерам событий, именам объектов или на основании иных критериев. Попавшие в CLIPS факты могут быть использованы для конструирования составных моделей атак, которые было бы довольно сложно распознать чисто программным способом.

Третий механизм, поддерживаемый сервером CMDS, – это генератор графических отчетов о трендах. Такие отчеты позволяют выявлять нарушения, которые не были зарегистрированы в режиме реального времени, производить оценку

нанесенного ущерба, настраивать производительность и перераспределять нагрузку в сетевой среде. Генератор отчетов может активизироваться автоматически в заранее установленные моменты времени или по требованию администратора.

Описанные механизмы выявления атак рассматривают данные мониторинга под разными углами зрения и потому естественным образом дополняют друг друга, образуя довольно прочную линию сетевой защиты.

Система CMDS снабжена удобным графическим интерфейсом пользователя, который служит для представления результатов мониторинга и выполнения простейших операций системного управления. В окнах интерфейса можно наблюдать как первичные, необработанные данные, относящиеся к отдельным пользователям, так и генерируемые статистические профили, информация может отображаться в реальном масштабе времени или по требованию оператора. При возникновении опасности нарушения службы защиты на экране появляются окна предупреждений, в которых приводится описание угрозы и предлагается на выбор несколько вариантов дальнейших действий, а именно: – оставить без внимания; – усилить наблюдение – для данного пользователя создается специальное окно мониторинга, в котором в режиме реального времени ведется слежение за первичными данными и их обработкой сервером CM0S; – отказать в доступе – на контролируемом компьютере запускается процесс, который уничтожает все процессы, принадлежащие нарушителю (для этого требуются права уровня Super-user); – экстренный останов – принудительное прекращение работы наблюдаемой системы в целом. Данный вариант особенно полезен в том случае, если возникли подозрения на наличие вируса: важно успеть избавиться от него до того, как будут заражены другие системы в сети. Средства настройки позволяют включить в арсенал защитных средств другие варианты реагирования на сетевые атаки.

Статистическая информация представляется в CMDS в графическом виде: так оператор быстрее заметит неадекватное поведение какой-либо системы. Данные отображаются по категориям в форме графиков, показывающих отклонение текущих значений от ожидаемых, или гистограмм с пороговыми значениями; вывод диаграмм осуществляется в реальном времени по мере генерации контрольных записей. Сильное отклонение от расчетных величин либо превышение пороговых значений рассматривается как признак возможного нарушения службы защиты. При этом оператор видит не только текущие значения, но и скорость их изменения, а также ретроспективные данные, которые могут представлять интерес. Кроме того, используется еще один вид графических отчетов – гистограммы трендов, на которых отображается предыстория отслеживаемых данных в форме, позволяющей соотносить их с другой информацией, не связанной напрямую с системой зашиты: например, спады или пики активности пользователя можно сопоставлять с такими сведениями, как расписание отпусков, сроки сдачи проектов, случаи отсутствия на работе и т. п. Так, повышенная частота операций просмотра при аномально низком проценте операций выполнения может означать, что работает автоматическая программа поиска информации, однако если данный пользователь числится в отпуске, а перед началом работы программы были зафиксированы неудачные попытки ввода принадлежащего ему пароля, то налицо повод для тщательного расследования.

2.6. Интеллектуальное противодействие противнику в корпоративной сети обмена информацией

На дереве целей и задач в развернутом виде мы выделили два комплекса задач, относящихся к ответным действиям СЗИ по отношению к противнику в ИВС: простое реагирование (или просто - реагирование) и интеллектуальное противодействие.

Эти два вида ответных действий в ИВС сродни с действиями человека по отношению к неблагоприятным факторам среды. Реагированию в ИВС можно поставить в соответствие рефлекторное сокращение мышц человека и сигнал боли, передаваемый через нервную систему. Само по себе такое реагирование не может предотвратить других разнообразных воздействий среды на человека. Интеллектуальному противодействию в ИВС можно поставить в соответствие сознательные действия человека по управлению средой, исключающие или значительно снижающие проявления ее неблагоприятных факторов.

Таким образом, комплекс задач интеллектуального противодействия в ИВС является определяющим в общем успехе информационной борьбы с противником.

2.6.1. Определение и задачи интеллектуального противодействия в сети обмена информацией

Приведем ряд определений.

Определение. Под *реагированием* на несанкционированные действия в ИВС будем понимать реакцию СЗИ, которая может включать в себя следующие действия:

- сигнализацию о НСД;
- блокировку (отключение терминала, группы терминалов, элементов ИВС и т.п.);
- отказ в запросе.

Реагирование на НСД может осуществляться как автоматически, так и с участием должностного лица, ответственного за информационную безопасность.

Определение. Под *несанкционированными действиями* в ИВС будем понимать информационное воздействие на ИР, персонал, информационные системы ИВС, элементы ИВС и ИВС в целом, а также подготовку этого воздействия.

Определение. *Информационная система* – это организационно-упорядоченная совокупность данных и информационных технологий, в том числе с использованием средств вычислительной техники и ИВС, реализующих информационные процессы [76].

Определение. *Информационные процессы* – это процессы сбора, обработки, накопления, хранения, поиска и распределения информации [76].

Определение. *Информационные технологии* – это совокупность методов, способов, приемов, средств обработки информации и регламентированного порядка их применения, направленные на удовлетворение информационных потребностей [76].

Определение. Под ***интеллектуальным противодействием*** (ИП) – будем понимать комплекс решаемых в ИВС задач по реагированию на несанкционированные действия на основе оперативного анализа стратегии противника и применяемого им информационного оружия, технических возможностей ИВС, текущих задач информационной борьбы и управления корпорацией, в том числе, и с использованием средств искусственного интеллекта.

Интеллектуальное противодействие подчинено следующим целям информационной борьбы в ИВС (рис.13):

- снизить время бесконтрольного присутствия противника в ИВС;
- дезинформировать противника;
- дезорганизовать действия противника по осуществлению информационных атак в ИВС;
- снизить нецелевую нагрузку на ИВС;
- воздействовать на ресурсы противника.

Под *бесконтрольностью присутствия противника в ИВС* понимается то, что в определенный момент времени СЗИ неизвестна стратегия действий противника или на действия противника не представляется возможным повлиять.

Дезинформация противника в ИВС представляет собой санкционированное распространение не соответствующей действительности информации о планах, способах действий и намерениях руководства корпорации.

Дезорганизация действий противника в ИВС – это действия, направленные на дезориентацию противника относительно реального расположения интересующего его объекта атаки (ИР, информационной системы, элемента ИВС), а также действия по рассогласованию технологически взаимосвязанных СИВ, применяемых противником в ИВС.

Снижение нецелевой нагрузки на ИВС подразумевает перераспределение потоков пакетов в ИВС, связанных с информационной войной, и снижение их интенсивностей таким образом, чтобы минимизировать показатель, характеризующий нагрузку в ИВС.

Под *нагрузкой* в ИВС традиционно понимают интенсивность пакетов информации всех видов и назначения.

Показатель, характеризующий нагрузку на ИВС, выбирается с учетом используемого в сети алгоритма маршрутизации. И, например, для ИВС в которой используется адаптивный алгоритм маршрутизации [87], принято в качестве такого показателя выбирать сумму коэффициентов недоиспользования пропускных способностей ветвей связи [87, 63].

Воздействие на ресурсы противника предполагает поглощение этих ресурсов и неэффективное использование их противником.

Ресурсы противника, на которые можно воздействовать в ходе информационной борьбы, следующие:

- время, затрачиваемое противником для достижения цели информационного нападения;
- ресурсы вычислительных средств, затрачиваемые противником в ходе информационной атаки и в процессе верификации полученной информации от атакованного объекта;
- людские ресурсы, затрачиваемые в ходе информационной борьбы;
- морально-психологическая устойчивость лиц, участвующих в информационной борьбе на стороне противника;
- информационные ресурсы противника, формируемые в результате информационных атак на ИВС;
- информационное оружие противника и способы его применения;
- материальные и финансовые затраты противника на ведение информационной войны.

В качестве основного подхода ИП в данной работе используется понятие подставляемого противнику (ложного) объекта атаки, имитирующего процесс или результат работы объекта атаки, выбранного противником.

С помощью этого подхода можно достичь целей информационной борьбы, соответствующих комплексу задач интеллектуального противодействия.

Основные задачи ИП следующие.

1. Классификация НСД в ИВС, которая включает в себя:
   - классификацию объекта НСД;
   - классификацию субъекта НСД;
   - классификацию используемого информационного оружия.
2. Выбор вида ИП, который заключается в предварительном выборе типа стратегии противодействия на основе проведенной классификации НСД.
3. Выбор зоны интеллектуального противодействия, который включает в себя:
   - перераспределение нагрузки на ИВС, вызванной информационной войной;
   - оптимизацию размещения в ИВС узлов интеллектуального противодействия (центров безопасности).
4. Осуществление интеллектуального противодействия, включающее:
   - построение программы ИП, имитирующей функционирование ОА;
   - собственно реализация ИП.

2.6.2. Особенности построения и использования ложного объекта атаки при интеллектуальном противодействии для достижения целей информационной борьбы в корпоративной ИВС

Суть предложенного и описанного в работе подхода, заключается в следующем.

Согласно концептуальной модели информационной борьбы в ИВС противник (субъект атаки) осуществляет информационное воздействие (атаку) на объект атаки посредством тех или иных каналов реализации угроз. СЗИ ИВС призвана обеспечить информационную безопасность объекта атаки. При этом заранее неизвестно, какую стратегию выберет противник, и какую стратегию он выберет в случае неуспешного завершения предыдущей атаки.

Эта неопределенность приводит к увеличению времени бесконтрольных действий противника в ИВС.

Поэтому СЗИ должна «дорожить» тем, что удалось зафиксировать НСД и распознать стратегию противника, и не разглашать факт своей осведомленности противнику.

Но в то же время должна обеспечиваться информационная безопасность объекта атаки.

Это противоречие можно разрешить путем введения в ИВС ложного объекта атаки (ЛОА).

Определение. Под ложным объектом атаки будем понимать подставляемый противнику объект или элемент ИВС, имитирующий процесс или результат работы объекта атаки, выбранного противником.

Ложный ОА предназначен для использования в процессе интеллектуального противодействия и способствует достижению целей информационной борьбы в ИВС.

По сути, ЛОА представляет собой проекцию ОА на плоскости его взаимодействия с участниками информационной войны в ИВС (противника и субъекта информационной борьбы ИВС).

Принципы построения ЛОА следующие:

- визуальное подобие объекту атаки, включая интерфейс взаимодействия с пользователями (а, соответственно, и с противником) и форматы данных ввода-вывода;
- развитый интерфейс с субъектом информационной борьбы (администратором сети или оператором системы безопасности);
- управляемость ЛОА со стороны субъекта информационной борьбы.
- модульность построения ЛОА, обеспечивающая возможность его наращивания и оперативной реконфигурации.

В общем случае ЛОА состоит из следующих модулей:

- модуль взаимодействия с противником.
- основной модуль, отвечающий за преобразование данных и композицию всех других модулей ЛОА.
- модуль взаимодействия с субъектом информационной борьбы.
- модуль взаимодействия с центром безопасности ИВС.

Функционирует ЛОА на основе корректно построенного протокола интеллектуального противодействия, привлекая, когда необходимо, субъекта информационной борьбы для решения нетривиальных ситуаций взаимодействия с противником. Программную реализацию ЛОА будем называть программой интеллектуального противодействия (ПИП).

В целях широкого использования опыта субъектов информационной борьбы методика построения ПИП должна максимально обеспечивать настройку ПИП на конкретного ее пользователя (администратора сети или службу безопасности).

При этом в качестве подставляемого объекта можно рассматривать и подлинный объект атаки, функционирующий в специальном режиме интеллектуального противодействия. Предпочтительно, чтобы работа ОА в этом режиме осуществлялась параллельно с использованием этого объекта при решении целевых задач управления бизнес-процессами.

Решение задачи адаптации, а также сохранение бесценного опыта экспертов информационной борьбы, можно обеспечить именно компонентным способом построения ПИП.

Заметим, что ПИП может функционировать и в автоматическом режиме. В этом случае при возникновении сложных ситуаций необходимо прибегнуть к традиционным методам простого реагирования, чтобы не демаскировать факт использования ЛОА и не допустить нарушение безопасности информации в ИВС.

В ходе интеллектуального противодействия с использование ЛОА может быть окончательно локализовано место подключения противника, уточнена его стратегия и, в некоторых, случаях стратегические цели противника, решаемые в том числе и с использованием результатов информационных боевых действий в ИВС.

В случае незначительности НСД и неэффективности интеллектуального противодействия осуществляется возврат управления ПИП механизмам простого реагирования.

Примечательно, что в сети «не чувствуется» реального расстояния между атакующим и атакуемым узлами. Поэтому ЛОА может быть организован достаточно близко к зоне расположения противника в ИВС. Тем самым достигается оптимальное перераспределение не целевой нагрузки на базовую ИВС.

В качестве ЛОА могут использоваться проекции баз данных, программного обеспечения информационных систем различного назначения, информационных серверов, объектов инфраструктуры сети и телеслужб ИВС, автоматизированных рабочих мест операторов.

2.7. Основы теории искусственной жизни

Сложность поставленной задачи заставляет искать новые подходы к ее решению. Как мы отмечали ранее, к решению столь сложной задачи необходимо подходить с позиций искусственного интеллекта. Их всего многообразия методов современной теории искусственного интеллекта нами было выбрано теоретическое направление известное как «искусственная жизнь».

В июне 1995 года в Испании состоялся крупный международный конгресс ECAL, целиком посвященный проблемам новой науки и исследователи возвестили о пришествии четвертой волны искусственного интеллекта.

Итак, что такое искусственная жизнь. Самое общее определение дает журнал MIDRANGE Systems, трактуя ее как “компьютерное моделирование живых объектов” (MIDRANGE Systems, 1995, v. 8, №13, p. 29). Однако на практике к искусственной жизни принято относить компьютерные модели, обладающие рядом конкретных особенностей [94, 95].

Во-первых, центральная модель системы обладает способностью адаптироваться к условиям внешнего мира, пополняя знания о нем путем взаимодействия с другими объектами и средой.

Во-вторых, компоненты системы, развиваясь в процессе эволюции, способны передавать свои характерные черты по наследству, то есть присутствует механизм порождения новых поколений - путем деления, скрещивания либо дублирования существующих объектов.

В-третьих, окружающий мир достаточно жесток и сводит к минимуму шансы на выживание и появление потомства у слабых и плохо приспособленных особей.

И, наконец, присутствует механизм порождения новых форм (аналог мутаций в реальном мире), обычно содержащий элемент случайности.

Иными словами, чтобы решить указанную задачу методами искусственной жизни (например, разработать эффективную систему защиты информации в сети и т.п.), надо построить динамическую модель среды, в которой предстоит существовать проектируемому объекту, населить ее множеством разновидностей этого объекта и дать им “пожить” несколько поколений. Слабые особи будут погибать, сильные скрещиваться, закрепляя в новых поколениях свои лучшие черты. Через несколько десятков (иногда - сотен и даже тысяч) циклов такая селекция породит “цивилизацию” практически неуязвимых особей, идеально приспособленных к заданной вами модели мира. И можно с уверенностью сказать, что “жизненная сила” прошедшего жесточайший отбор решения будет достаточной, чтобы успешно противостоять любым действиям конкурентов.

Чем обусловлено рождение новой концепции? Почему нельзя решать те же задачи классическими методами теории управления, оптимизации и системного анализа? Дело в том, что любой проектировщик сложных систем сталкивается с одним и тем же комплексом проблем, плохо поддающихся решению традиционными методами. Неполнота знаний о внешнем мире, неизбежная погрешность датчиков, непредсказуемость реальных ситуаций-все это заставляет разработчиков мечтать об интеллектуальных адаптивных системах, способных подстраиваться к изменению “правил игры” и самостоятельно ориентироваться в сложных условиях.

Кроме того, реальным сдерживающим фактором при решении многих (если не большинства) серьезных задач становится фактор размерности. Проектировщик не в состоянии учесть и свести в общую систему уравнений всю совокупность внешних условий - особенно при наличии множества активных противников. Самостоятельная адаптация системы в процессе динамического моделирования “условий, приближенных к боевым” - едва ли не единственный способ решения задачи в таких случаях.

Есть и еще одна причина - философская. Человек пока лишен способности прогнозировать долгосрочные последствия своих - пусть и вызванных самыми благими намерениями - поступков. Так пусть живучесть решений проверяется самой жизнью.

Строго говоря, искусственная жизнь - это обобщающий метод построения динамических моделей, базирующийся на совокупности других наук - генетических алгоритмов, М-сетей, нейронных сетей, теории хаоса, системной динамики и др.

В общем виде концепция искусственной жизни представлена на рис. 2.3.

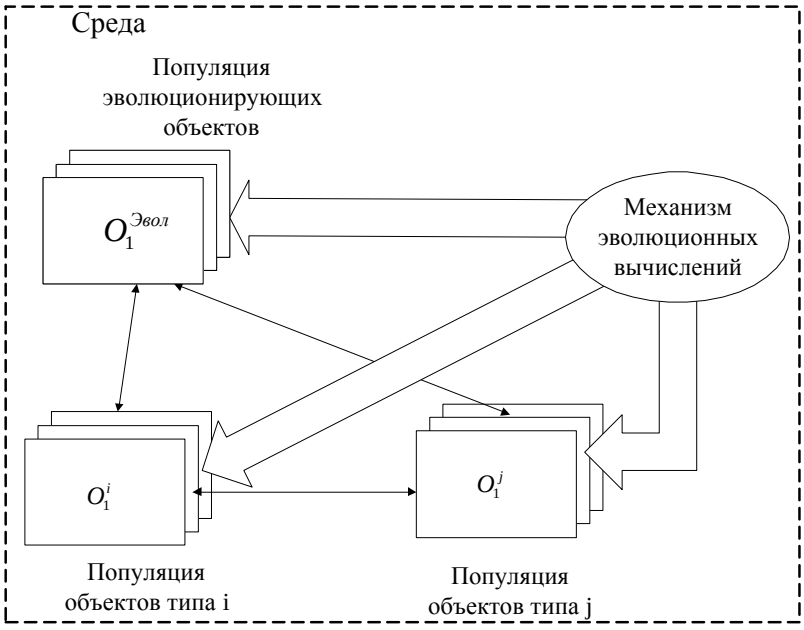


Рис. 2.3. Концепция искусственной жизни

На рис. 2.3. имеют место следующие обозначения:

$O^{эвол}(O_1^{ЭВОЛ}, O_2^{ЭВОЛ}, ... O_{n^{эвол}}^{ЭВОЛ})$ - рассматриваемая популяция (совокупность) объектов;

O^i, O^j – конкурирующие популяции объектов (конкуренция осуществляется за какой-то конкретный ресурс);

Каждая популяция имеет свою целевую функцию, вид которой определяется целью объектов популяции, средой и наличием объектов других популяций.

Достижение нижнего порогового значения этой функции свидетельствует о «гибели», то есть о неэффективности, с данной точки зрения, конкретного объекта.

Например, если $f_{O^{эвол}}(O_k^{эвол})$ - целевая функция популяции $O^{эвол}$, то при $f_{O^{эвол}}(O_k^{эвол}) < f_{O^{эвол}}^{порог}$ объект $O_k^{эвол}$ погибает.

Естественно, что в рамках искусственной жизни каждая популяция стремится максимизировать значения целевой функции своих объектов и осуществляет это путем применения механизма эволюционных вычислений.

Подведем краткие итоги второй главы нашей книги. Мы ввели предполагаемую структуру сети передачи данных и попытались разобраться с информационным ресурсом корпорации, чтобы эффективно его защищать. Кратко рассмотрели основные существующие направления создания систем защиты информации в сетях подобного уровня. В заключении описаны основы концепции искусственной жизни.

Глава 3. Методы интеллектуального противодействия информационному нападению в корпоративной информационно-вычислительной сети

3.1. Метод выбора вида противодействия в корпоративной сети обмена информацией

3.1.1. Постановка задачи

Вид противодействия противнику в терминах поставленной в настоящей работе задачи определяется типом стратегии противодействия противнику в СОИ $i_L \in L$.

Тогда разработка метода противодействия направлена на решение следующих задач интеллектуального противодействия в СОИ (рис. 1. 11):

- классификация несанкционированных действий в СОИ;
- выбор типа стратегии интеллектуального противодействия в СОИ.

Исходя из этого, задача разработки метода интеллектуального противодействия в СОИ может быть поставлена следующим образом.

Исходные данные.

Индикатор несанкционированных действий в СОИ

$$ind = \begin{cases} 1, & \textit{если системой контроля доступа, в соответствии со своими} \\ & \textit{алгоритмами и протоколами работы, зафиксирована попытка} \\ & \textit{НСД в СОИ}; \\ 0, & \textit{в противном случае.} \end{cases}$$

Результат работы системы контроля доступа в СОИ представлен кортежем спецификации НСД $Z_{i<n_Z>} \in Z$, где $Z=\{Z_i\}$ – множество возможных спецификаций НСД (кортежей наблюдаемых проявлений (признаков) НСД).

Множество $\mathbf{M}=\{1,\dots i_M,\dots n_M\}$ – множество типов стратегий противника, где n_M – число рассматриваемых типов стратегий противника.

Множество $\mathbf{L}=\{1,\dots i_L,\dots n_L\}$ – множество реализуемых в СОИ типов стратегий ИП, где n_L – число типов стратегий ИП.

Вектор $Z'_{i<n_{Z'}>} \in Z'$ - вектор управляющих воздействий СУ.

Время выбора $T_{ВЫБОР}$ типа стратегии ИП $i_L \in L$ определяется следующим образом.

$$\mathbf{T_{ВЫБОР}=T_{КЛ.НСД}+T_{ПР}} \qquad (3.1.)$$

где $T_{КЛ.НСД}$ – время отнесения НСД к типу i_m стратегии противника, $T_{ПР}$ - время выбора типа стратегии ИП i_L.

При этом $T_{КЛ.НСД} >> T_{ПР}$, тогда $T_{ВЫБОР}$ определяется функцией F_B построения и реализации системы классификации S_{Cl}

$$\mathbf{T_{ВЫБОР} = F_B(S_{cl}),}$$

где $F_B(S_{cl})$ – функция, определяемая способом построения и реализации S_{cl}.

Необходимо. Разработать метод выбора вида противодействия, который позволяет на основе исходных данных задачи определить тип $\boldsymbol{i_L}$ стратегии противодействия за минимальное время $\mathbf{T_{ВЫБОР}}$.

$$T_{ВЫБОР} = \min_{i \in I} (F_{B\,i}(S_{cl})) \qquad (3.2)$$

Как подчеркивалось в главе I, в настоящее время ИП в СОИ практически не осуществляется, да и в существующих условиях осуществление ИП возможно только с участием квалифицированного специалиста (эксперта) [62].

В этом случае очевидны следующие недостатки:

- эксперт-человек подвержен разного рода воздействиям, во многом определяющим его выбор даже в практически одинаковых ситуациях;
- замена эксперта полностью разрушает сложившуюся систему ИП;
- эксперт склонен к стереотипному мышлению, что не позволяет осуществлять разностороннее ИП;
- неприемлемым является и большое время решения задач классификации НСД и выбора вида ИП.

Все это говорит о необходимости обращения к методам искусственного интеллекта для выбора тех или иных видов ИП в условиях расширяющегося многообразия форм и методов ведения ИБ.

3.1.2. Многоуровневая система классификации несанкционированных действий в сети обмена информацией

Чтобы принять решение о типе стратегии противника необходимо, прежде всего, классифицировать НСД.

В общем случае классификация НСД включает в себя:
- классификацию субъекта НСД;
- классификацию объекта НСД;
- классификацию используемого субъектом информационного оружия.

Рекомендуется строить многоуровневую классификацию, при которой классификационные признаки нижних уровней определяют классификационные признаки верхних уровней.

Семантическое описание классификационных признаков НСД в СОИ подробно представлено в работе [74].

Систему классификации представим следующим образом

$$\mathbf{S_{cl} = <Cl, RCl>,}$$

где $Cl = < Cl^{1}_{<N^1>}, Cl^{2}_{<N^2>}, \dots Cl^{n}_{<N^n>} >$; $RCl = < RCl^{12}_{<N^1 \times N^2>}, \dots RCl^{n-1\,n}_{<N^{n-1} \times N^n>} >$;

Вектор $Cl^{i}_{<N^i>}$ есть вектор, компонентами которого являются классификационные признаки i-го уровня классификации S_{cl}.

Элемент cl^{i}_{k} - элемент вектора $Cl^{i}_{<N^i>}$, представляющий собой значение k-го классификационного признака i-го уровня классификации S_{cl}.

Матрица $RCl^{ij}_{<N^i \times N^j>}$ есть матрица, элементами которой являются связи между классификационными признаками i-го и j-го уровня классификации S_{cl}.

Элемент rcl^{ij}_{kh} - элемент матрицы $RCl^{ij}_{<N^i \times N^j>}$, представляющий собой связь между k-ым классификационным признаком i-го уровня и h-ым классификационным признаком j-го уровня классификации S_{cl}.

Последний (n-ый) уровень классификации определяет тип стратегии противника.

Проблеме классификации состоит в том, что конкретная попытка НСД может нести в себе признаки различных типов стратегии противника. Причем один или несколько признаков могут служить маскировкой истинной цели НСД.

Задача классификации НСД и выбора вида ИП далеко не тривиальная при достаточно большой размерности системы классификации НСД, множеств L и M. Поэтому, как отмечалось выше, целесообразно ее решать с привлечением средств искусственного интеллекта.

В качестве такого средства в работе выбран аппарат М-сетей [4].

3.2. Общие вопросы аппарата М-сетей

Рассмотрев ряд подходов эвристического моделирования [1, 4, 99] выяснилось, что М-сеть эффективнее, полнее отвечает требованиям, предъявляемым к решаемой нами задаче, в

частности, требованию сохранение гибкости, принимаемых решений, в масштабных задачах, где обычно используется интеллект человека.

М-сети относятся к классу сетей с семантикой и являются математическим аппаратом эвристического моделирования. Аппарат М-сетей предложен академиком Н.М. Амосовым и развит его последователями как попытка создания искусственного разума.

Целью создания искусственного разума является разработка методов построения систем, которые, не уступая по эффективности человеческому мозгу, могли бы обеспечивать решение разнообразных сложных задач, не уступающих сложности задач, решаемых человеком.

Именно такой сложной задачей с высокой степенью участия человека является задача классификации НСД и выбора вида интеллектуального противодействия в СОИ.

В связи с этим мы нашли вполне целесообразным обратиться к аппарату М-сетей, так как он разработан именно для моделирования человеческого мышления и, следовательно, с его помощью можно добиться принятия адекватных решений.

Выбранный в работе язык моделирования развит на основе представления о мышлении как о направленном процессе взаимодействия множества информационных моделей объектов внешнего и внутреннего мира человека в коре его головного мозга. Искусственные системы, строящиеся на основе этого представления, реализуются в виде специфических сетей, названных М-сетями [4].

Узлы М-сети есть формальные элементы, которые ставятся в соответствие информационным моделям коры головного мозга (корковым информационным моделям). Будем называть эти узлы i-моделями. Связи между i-моделями отвечают предполагаемым связями между корковыми моделями. С содержательной стороны i-модели могут быть поставлены в соответствие образам и понятиям, которыми оперирует человек. Поэтому с помощью М-сети можно представлять взаимосвязанные системы образов и понятий, предположительно используемые человеком в ходе мышления. М-сеть является, таким образом, сетью с семантикой.

Рассмотрим содержание основных понятий использованного в работе языка моделирования.

3.2.1. Понятие i-моделей и связи между ними

i-Модель есть формальный элемент, которому может быть поставлено в соответствие определенное понятие. В семантическом плане i-модель является знаком понятия. С конструктивной точки зрения i-модель есть элемент некоторой структуры, который может находиться в ряде отличных друг от друга состояний. С функциональной точки зрения i-модель есть набор некоторых операторов или алгоритмов переработки информации.

Опишем i-модель как элемент, обладающий следующими свойствами [4].

Каждая i-модель имеет конечное число входов и один выход.

Каждая i-модель может находиться в состоянии *возбуждения*, степень которого характеризуется числовой величиной **П**, называемой *возбужденностью*. При описании функционирования i-модели в дискретном времени возбужденность j некоторой i-модели в момент времени t будем обозначать $П^t_j$.

i-Модели могут быть соединены направленными связями, по которым возбуждение передается от одних i-моделей к другим.

Приведем основные свойства связи.

Каждая связь может быть направлена от выхода i какой-либо i-модели к одному из входов j другой i-модели.

От выхода i-модели может отходить более чем одна связь, а к одному входу может подходить только одна связь.

Между двумя i-моделями может существовать только одна связь.

Каждая связь характеризуется упорядоченным набором параметров R, называемым проходимостью связи, или, для краткости, просто связью. Проходимость связи, направленной от i-модели j к i-модели i, в момент дискретного времени t будем обозначать R^t_{ij}.

Семантика i-моделей задается двумя путями: во-первых, ее определяет соответствие, установленное между данной i-моделью и и некоторым содержательным понятием, а, во-вторых - совокупность связей, соединяющих данную i-модель с другими.

Характеристики связи. Связь R_{ij} есть вектор $R_{ij}= <r_{ij}, \tilde{r}_{ij}, r_{(0)ij}, \tilde{r}_{(0)ij}>$,
где параметры r_{ij} и $\tilde{r}_{ij}$ - усиливающий и тормозной компоненты проходимости связи, а параметры $r_{(0)ij}$ и $\tilde{r}_{(0)ij}$ - остаточные составляющие этих компонент. Эти параметры могут принимать численные значения, что содержательно означает следующее. Возбуждение, поступающее по связи R_{ij}, может как увеличивать, так и уменьшать, тормозить возбужденность i-модели i. Численной мерой этих воздействий и являются значения r_{ij} и $\tilde{r}_{ij}$. Эти значения могут меняться во времени, так что в случае $r_{ij} >> \tilde{r}_{ij}$ можно говорить об усиливающем характере связи R_{ij}, а в обратном случае - об ее тормозном характере. Остаточные связи всегда удовлетворяют соотношениям

$$r^t_{(0)ij} \le r^t_{ij};\ \tilde{r}^{\,t}_{(0)ij} \le \tilde{r}^{\,t}_{ij} \qquad (3.3)$$

и составляют долговременную память связей.

Будем полагать, что $R_{ij} = 0$ в том случае, если равны нулю все ее компоненты.

Каждая связь описывается следующими характеристиками.

1. *Характеристика проторения* есть функция, описывающая зависимость проходимости связи от возбужденностей соединяемых ею i-моделей:

$$R^t_{ij} = R\ (\Pi^t_i, \Pi^t_j, R^{t-1}_{ij}, \Delta^t), \qquad (3.4)$$

где Δ - интегральная оценка качества функционирования М-сети. Эта оценка формируется с помощью специальных i-моделей сети. Она имеет смысл и может быть определена в тех случаях, когда на основе М-сети уже построена некоторая динамическая модель - М-автомат, заданы цели его функционирования и определены критерии качества его работы. По отношению к этим критериям и формируется оценка Δ. Конкретные механизмы и правила вычисления значений Δ будут, соответственно, определяться при построении той или иной конкретной модели.

Будем называть связь непроторенной в момент t, если для нее $R^t_{ij} = 0$. Если для некоторой связи $R^{t-1}_{ij} =0$ и $R^t_{ij} \neq 0$, то будем говорить, что в момент t произошло установление связи R_{ij}. Таким образом, установление является частным случаем проторения. Для непроторенной связи функция (3.4) может иметь иной вид, чем для проторенной, так что в случае установления

$$R^t_{ij} = R\ (\Pi^t_i, \Pi^t_j, \Delta^t). \qquad (3.5)$$

Функцию (3.5) называют характеристикой установления связи [4].

2. *Характеристика затухания связи* - это набор функций, описывающих уменьшение значений ее параметров во времени. Эта характеристика описывает процесс уменьшения проходимости связи R_{ij} при условии, что в некоторый начальный момент времени t_0 значение $R^{t0}_{ij} \neq 0$ и во все последующие моменты времени $\Pi_i=0$ и $\Pi_j=0$. Для усиливающих и тормозных компонентов связи характеристики затухания имеют вид

$$r^t_{ij}= r_1(r^{t-1}_{ij}, r^t_{(0)ij}), \qquad (3.6а)$$

$$\tilde{r}^{\,t}_{ij}= r_2(\tilde{r}^{\,t-1}_{ij}, \tilde{r}^{\,t}_{(0)ij}). \qquad (3.6б)$$

Функции (3.6а) и (3.6б) описывают такой процесс затухания, при котором значения r_{ij} и $\tilde{r}_{ij}$, уменьшаясь, стремятся к значениям их остаточных составляющих. Для остаточных составляющих характеристики затухания

$$r^t_{(0)ij}= r_3(r^{t-1}_{(0)ij}), \qquad (3.7а)$$

$$\tilde{r}^{\,t}_{(0)ij}= r_4(\tilde{r}^{\,t-1}_{(0)ij}). \qquad (3.7б)$$

таковы, что описывают стремление значений $r_{(0)ij}$ и $r_{(0)ij}$ к нулю. Усиливающие и тормозные компоненты затухают во времени намного быстрее, чем их остаточные составляющие. Поэтому, начиная с момента t_0, проходимость R_{ij} уменьшается сравнительно быстро, то есть имеет место “кратковременная память связей”. Затем значения r_{ij} и $\tilde{r}_{ij}$ достигают значений $r_{(0)ij}$ и $\tilde{r}_{(0)ij}$, и, поскольку условия (3.3) должны всегда сохраняться, дальнейшее уменьшение

R_{ij} происходит в соответствии с характеристиками (3.7а) и (3.7б), то есть существенно медленнее, то есть имеет место "долговременная память связей".

3. *Характеристика передачи связи* определяет значение воздействия $E_i(\tilde{E}_i)$ на входе i-модели i в зависимости от проходимости $r_{ij}(\tilde{r}_{ij})$ связи между i-й и j-й i-моделями и величины возбужденности $П_j$. В общем случае для i-й i-модели будем рассматривать величину входного воздействия по усиливающим (E_i) и тормозным ($\tilde{E}_j$) связям:

$$E^t_i = E(П^t_1, П^t_2, ..., r^t_{i1}, r^t_{i2}, ...), \quad (3.8а)$$

$$\tilde{E}_{ti} = \tilde{E}(Пt1, Пt2, ..., \tilde{r}_{ti1}, \tilde{r}_{ti2}, ...), \quad (3.8б)$$

Характеристики i-модели. В каждый момент времени каждая i-модель обладает определенной возбудимостью, под которой понимается способность i-модели отвечать собственным возбуждением на входное воздействие по усиливающим связям. Чем выше возбудимость i-модели, тем большей возбужденностью ответит она на постоянное входное воздействие. Возбудимость i-модели может изменяться во времени. Будем характеризовать возбудимость i-й i-модели двумя параметрами. Один из них - *порог возбуждения* i-модели $\mathbf{Q^t_i}$, представляющий собой минимальное значение E^t_i, необходимое для возбуждения i-й i-модели в момент времени t. Другой параметр - *условный коэффициент возбудимости* **К**. Будем различать два значения К - текушее (K^t_i), изменяющееся в зависимости от величины тормозных воздействий $\tilde{E}^t_i$, и начальное ($K^t_{нi}$). Значения параметров $K^t_{нi}$ и Q^t_i могут изменяться во времени значительно медленнее, чем K^t_i.

Перечислим характеристики i-моделей.

1. Характеристика торможения есть функция, определяющая изменение возбудимости i-модели в зависимости от величины суммарного воздействия на нее по тормозным связям:

$$K^t_i = Ф(K^t_{нi}, \tilde{E}^t_i). \quad (3.9)$$

2. Характеристика затухания есть функция, определяющая изменение возбужденности i-модели во времени:

$$Пti = T(Пt\text{-}1i, Ktнi). \quad (3.10)$$

Функция (3.10) описывает процесс уменьшения возбужденности при отсутствии воздействий на входах i-модели и характеризует "временную память возбуждений" i-моделей.

3. Характеристика возбуждения есть функция, определяющая значение возбужденности на выходе i-модели в зависимости от возбудимости i-модели и величины суммарного воздействия на нее по усиливающим связям:

$$П^t_i = П(K^t_i, Q^t_i, E^t_i). \quad (3.11)$$

4. Характеристики гипертрофии и адаптации определяют значения $K^t_{нi}$ и Q^t_i в зависимости от возбужденности i-модели:

$$Q^t_i = Q(Q^{t-1}_i, П^t_i), \quad (3.12а)$$

$$K^t_{нi} = K(K^{t-1}_{нi}, П^t_i). \quad (3.12б)$$

Функции (3.12а) и (3.12б) таковы, что определяемые ими изменения параметров невелики в каждый момент дискретного времени. Эти функции задаются таким образом, что для i-моделей, которые в течение длительного времени обладают малой возбужденностью, значение порога увеличивается, а значение коэффициента возбудимости уменьшается ("адаптация"). В результате редко или слабо возбуждающиеся i-модели оказываются трудновозбудимыми и мало влияют на процессы в сети. Если i-модель возбуждается часто и сильно, функции (3.12а) и (3.12б) обеспечивают уменьшение коэффициента возбудимости ("гипертрофия").

3.2.2. Определение М-сети и принципы ее функционирования

М-сеть представляет собой совокупность i-моделей и связей между ними. Фрагмент М-сети на примере восьми i-моделей представлен на рис.3.1.

С точки зрения взаимодействия со средой в М-сети будем различать внутренние и граничные i-модели. Внутренние i-модели являются решающими, граничные i-модели целесообразно разделить на рецепторные i-модели и i-модели действий [ ].

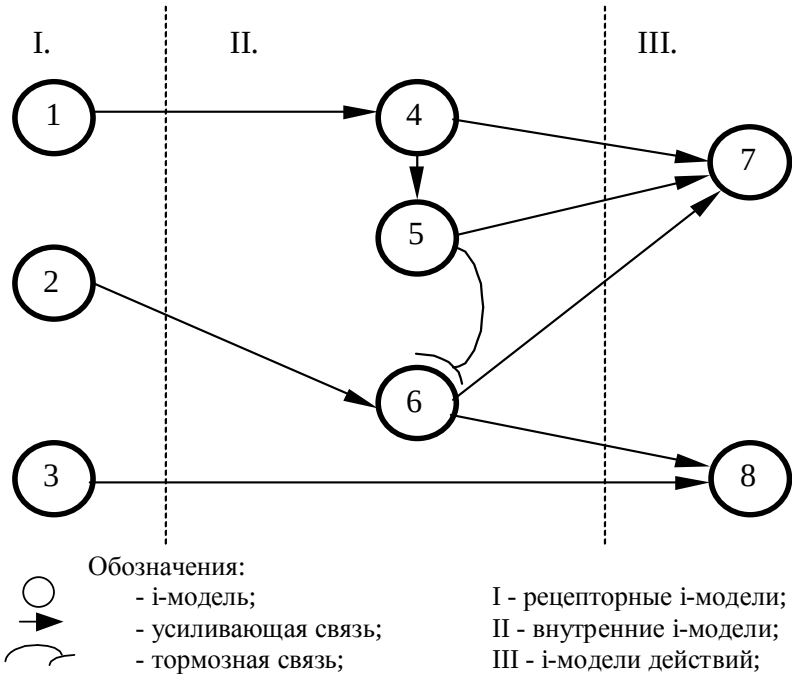


Рис. 3.1. Пример М-сети

Для того чтобы с помощью М-сети построить модель некоторого объекта или системы, необходимо в соответствии со спецификой этого объекта (системы) и целями моделирования задать М-сеть: зафиксировать совокупность i-моделей и их содержательных интерпретаций; задать связи между i-моделями, т.е. заранее определить их начальную конфигурацию в сети и начальное распределение **C** их проходимостей; зафиксировать начальное значение параметров и вид характеристик i-моделей и связей; зафиксировать начальное распределение **J** значений возбужденностей i-моделей. В ходе функционирования М-сети все первоначально заданные элементы могут изменяться. Этап задания сети является по существу этапом предварительной организации разрабатываемой модели.

Одним из основных принципов предварительной организации М-сетей является принцип иерархичности структуры, обеспечивающий возможность многоуровневой обработки информации (рис.3.2).

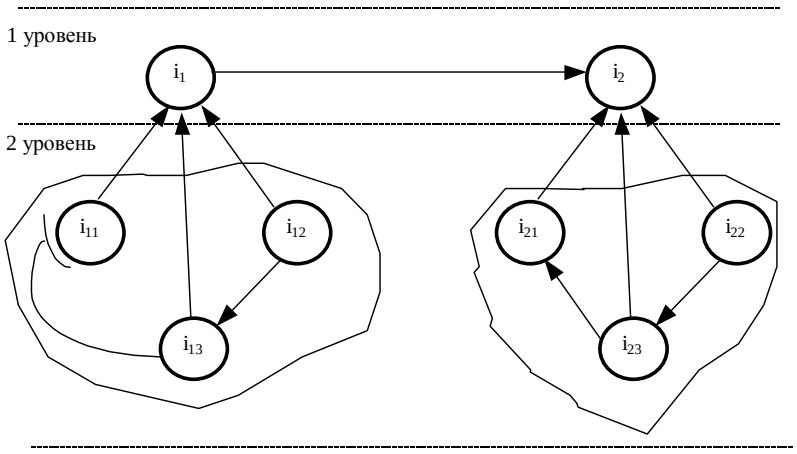


Рис.3.2. Фрагмент двухуровневой М-сети

Связи между i-моделями различных уровней отражают “родо-видовые” отношения соответствующих понятий. Кроме того, между i-моделями сети могут быть установлены “ассоциативные” связи.

Часто в М-сети бывает удобно выделять такие непересекающиеся подмножества i-моделей, каждое из которых удовлетворяет требованию смысловой однородности, близости содержательной интерпретации входящих в него i-моделей. Будем называть такие подмножества сферами М-сети. Взаимосвязь сфер осуществляется благодаря наличию связей между i-моделями, входящими в разные сферы.

Процессы в М-сети. М-сеть является, по существу, статической моделью, отображающей взаимосвязь определенных образов и понятий, а также степень их участия в процессах формирования воспроизводимой деятельности. Сами эти процессы и собственно деятельность могут быть реально воспроизведены только после того, как будет задан некоторый алгоритм, обеспечивающий функционирование М-сети во времени. Такой алгоритм должен

реализовывать изменение состояний i-моделей и связей М-сети в соответствии с введенными характеристиками.

Рассмотрим динамику процессов, которые могут протекать в М-сети, если алгоритм ее функционирования будет задан. Пусть на основе М-сети разрабатывается некоторая модель поведения. Будем считать, что М-сеть уже задана. Пусть модель находится в некоторой среде, содержащей "объекты". Будем полагать, что восприятие моделью определенного объекта среды возможно только в том случае , если в М-сети существует i-модель, соответствующая этому объекту. Совокупность i-моделей, соответствующих различным объектам среды, составляет сферу восприятия М-сети. Таким образом, восприятие моделью информации об объектах среды (то есть восприятие ситуаций) заключается в возбуждении соответствующих i-моделей в сфере восприятия М-сети. Отсюда по существующим между i-моделями связям возбуждение распространяется внутрь сети. Уточним это представление.

Функционирование М-сети рассматривается в дискретном времени. Пусть в некоторый момент t-1 каждая из i-моделей сети имела определенную возбужденность. Используя характеристики передачи связей (3.8), можно определить для каждой i-модели суммарные значения поступающих в нее входных воздействий. Далее, по характеристике торможения (3.9) для каждой i-модели можно установить значение коэффициента ее возбудимости, а по характеристике затухания (3.10) - степень влияния, или "переноса", ее возбужденности в момент t-1 на ее же возбужденность в момент t. И, наконец, с помощью характеристики возбуждения (3.11) для каждой i-модели можно найти значение того компонента ее возбужденности в момент t, который возникает как ответ на входное воздействие по усиливающим связям. Используя найденные величины, можно определить окончательное значение возбужденности каждой i-модели в момент t с помощью специальной формулы пересчета, которая строится на основе отдельных характеристик (3.8) - (3.11) и в общем виде может быть представлена так:

$$П^t_i=Ф(E^t_i,\ \tilde{E}^{\,t}_i,\ П^{t-1}_i,\ K^{t-1}_{ні},\ Q^{t-1}_i). \qquad (3.13)$$

Следует отметить, что формула пересчета характеризует М-сеть в целом: *конкретный вид этой формулы определяется один раз при построении модели и не изменяется при ее применении к каждой из i-моделей данной М-сети*. Это отличает ее от характеристик торможения, возбуждения и затухания, вид которых для различных i-моделей может быть разным.

Итак, применив формулу пересчета к каждой из i-моделей сети, мы определим значения их возбужденностей в момент t. Будем полагать, что расчет возбужденности производится для всех i-моделей сети одновременно. Будем называть эту процедуру пересчетом. В результате пересчета, таким образом, вычисляется распределение возбужденностей i-моделей в момент t на основе распределения возбужденностей в момент t-1. Повторяя пресчет, можно определить возбужденности в моменты t+1, t+2, ... Такие последовательные пересчеты и реализуются в любой модели, построенной с помощью М-сети.

После восприятия ситуации возбуждение по имеющимся между i-моделями связям распространяется внутрь сети (в ходе последовательных пересчетов).Через некоторое время оказываются в разной степени возбужденными определеннные (может быть, все) i-модели М-сети. При выполнении специальных условий, которые задаются в соответствии с задачами исследования, модель осуществляет некоторое действие - реакцию на воспринятую ситуацию. (Таким условием может быть, например, возбуждение до наперед заданной степени i-модели некоторого действия.)

Действие модели изменяет среду, эти изменения фиксируются в сфере восприятия М-сети путем возбуждения новых i-моделей. Возбуждение распространяется по сети до момента выполнения условий действия. Затем выполняется новое действие и т.д. Таким образом модель осуществляет в среде некоторое поведение.

В М-сети существует i-модели, возбуждение которых можно интерпретировать как интегральную оценку состояния М-сети в каждый момент времени. Эти i-модели соответствуют корковым центрам (гипотетическим) общей оценки состояния организма. Будем называть их i-моделями "удовлетворительно" (Уд) и "неудовлетворительно"(НУд). В процессе

функционирования М-сети возбужденность і-моделей Уд и НУд постоянно изменяется , так что в любой момент времени может быть вычислено значение общей оценки состояния, например в виде $\Delta^t=П^t_{Уд}-П^t_{НУд}$. Следует отметить, что состояние М-сети тесно связано с эффективностью вырабатываемых ею решений. Таким образом, оценка Δ характеризует не только состояние М-сети, но и эффективность поведения модели в целом, то есть эффективность функционирования М-сети как системы, принимающей решения. Структура оценки и механизмы ее формирования определяются структурой М-сети и задаются ее предварительной организацией.

Таким образом модель, построенная на основе М-сети, включает в себя ряд вспомогательных устройств, или алгоритмов. Эти вспомогательные системы непосредственно взаимодействуют со средой, выполняя роль рецепторов и эффекторов модели; реализуют специальные вычисления и другие операции, обслуживающие работу М-сети. Последняя же выступает здесь в роли высших отделов “мозга” модели, интегрирующего информацию о ее внешнем и внутреннем состоянии и вырабатывающего решения о выполнении действий.

Рассмотрим основные процессы, протекающие в М-сети при ее функционировании.

1. В зависимости от “истории” возбуждений каждой і-модели и в соответствии с характеристиками гипертрофии и адаптации (3.12) изменяются параметры ее возбудимости Q и $K_н$. Соответственно изменяются и характеристики возбуждения, торможения и затухания і-модели.

2. В зависимости от “ истории” совместных возбуждений каждой пары і-модели и в соответствии с характеристиками проторения (3.4) и затухания (3.6) и (3.7) связей изменяется проходимость связей М-сети.

Изменение характеристик і-моделей и проходимостей связей изменяет характер распространения возбуждения в М-сети. Это, в свою очередь, вызывает новые изменения характеристик и связей. Направление и вид этих изменений определяются состоянием “оценивающих” і-моделей Уд и НУд. Процессы, описанные в пп. 1 и 2, представляют собой процессы самообучения М-сети.

3. В ходе распространения возбуждений в М-сети и в соответствии с характеристикой установления (3.8) начинают функционировать новые, т.е. бывшие ранее непроторенными, связи между і-моделями. Таким образом, изменяется общая конфигурация связей сети.

4. В М-сети имеется некоторое множество і-моделей, в исходном состоянии не связанных ни друг с другом, ни с другими і-моделями сети. Для этих і-моделей не устанавливаются также соответствия с содержательными понятиями. Элементы такого рода, строго говоря, не являются і-моделями. Будем называть их резервными элементами. На множестве резервных элементов задется закон их спонтанного (случайного) возбуждения.

Пусть в некоторый момент времени спонтанно возбуждается один из резервных элементов. В этот же момент оказывается возбужденной некоторая совокупность других і-моделей М-сети. Между і-моделями этой совокупности и возбудившимся резервным элементом в соответствии с характеристикой (3.8) устанавливаются новые связи. Элемент становится, таким образом, “представителем”, т.е. і-моделью совокупности і-моделей, и получает некоторое семантическое значение, определяемое семантикой і-моделей, входящих в возбужденную совокупность.

Если и в дальнейшем і-модели той же совокупности часто оказываются возбужденными одновременно, то вновь установившиеся связи проторяются в еще большей степени и новая і-модель закрепляется. В противном случае она вскоре распадается из-за естественного затухания связей. Аналогичным образом образуются і-модели временных последовательностей. Описанные процессы лежат в основе образования в М-сети новых понятий из понятий, имевшихся в ней ранее.

Процессы, описанные в пп. 3 и 4, представляют собой процессы самоорганизации в М-сети. Процессы самообучения и самоорганизации могут приводить к образованию і-моделей “второго слоя”, то есть ансамблей из исходных і-моделей, которые в свою очередь, могут образовывать ансамбли “третьего слоя” и т.д. Ансамбли такого рода можно рассмат-

ривать как новые функциональные элементы М-сети, а процесс их образования - как процесс формирования новых сложных понятий на базе имевшихся ранее.

Группы характеристик. Введенные ранее характеристики элементов М-сети (i-моделей и связей) удобно разделять на несколько основных групп. Группу характеристик пересчета составляют те характеристики элементов сети, которые непосредственно используются при пересчете возбуждений и объединены в формуле пересчета (3.13). Сюда входят характеристики (3.8) - (3.11). Группу характеристик самообучения составляют (3.5) - (3.7), описывающие процессы изменения проходимости связей и параметров возбудимости i-моделей. Группу характеристик самоорганизации составляют характеристики установления связей (3.5) и законы спонтанного возбуждения резервных элементов М-сети.

Дадим определение М-сети [4]. М-сеть μ есть семерка:

$$\mu = <P, S, R, L, F, C, I>, \qquad (3.14)$$

где P - множество i-моделей; S - множество связей между i-моделями; R - группа характеристик самообучения; F - группа характеристик самоорганизации; C - начальное распределение проходимостей связей; I - начальное распределение возбуждений i-моделей.

Важным элементом функционирования М-сети является система усиления - торможения (СУТ). В связи с принципиальной важностью понимания роли СУТ в функционировании М-сети остановимся на этом вопросе достаточно подробно.

3.2.3. Вопросы функционирования системы усиления-торможения

С формальной стороны роль СУТ состоит в организации положительной обратной связи в процессах переработки информации М-сетью. Это обеспечивает на каждом временном промежутке доминирование наиболее важной в приспособительном плане программы переработки информации над другими программами, параллельно развивающимися в М-сети.

СУТ функционирует следующим образом. Пусть задана некоторая М-сеть. В процессе переработки информации возбужденности i-моделей сети изменяются. Величина возбуждения каждой i-модели косвенно свидетельствует о “важности”, или ценности, зафиксированной в ней информации. Естественно предположить, что выделение в каждый момент времени наиболее возбужденной i-модели и усиление ее влияния на общий ход переработки информации увеличит эффективность работы сети. Эти задачи и решает СУТ. В каждый момент времени она выбирает наиболее возбужденную i-модель, дополнительно повышает ее возбужденность и уменьшает возбудимость остальных i-моделей (притормаживает их). Если в некоторый момент времени одинаковое наибольшее возбуждение имеют n i-моделей, то дополнительная возбужденность от СУТ для каждой из них будет в n раз уменьшена: СУТ обладает конечным “энергетическим” запасом, величина которого зависит от общего состояния сети (в частности, от состояния i-моделей Уд и НУд).

Алгоритмы СУТ таковы, что возбужденность выделенных ею i-моделей постепенно уменьшается во времени. В то же время пропорционально растормаживаются остальные i-модели. Возбуждение от i-моделей, первоначально выделенных СУТ, распространяется по сети, увеличивая возбужденность связанных с ними i-моделей. В результате одна из них становится максимально возбужденной, СУТ переключается на нее и весь процесс повторяется.

СУТ содержит иерархически организованные подсистемы. Чем ниже уровень подсистемы, тем меньшее количество i-моделей находится под ее влиянием. Подсистемы СУТ более высоких уровней производят сравнение не возбужденностей отдельных i-моделей, а интегральных активностей более или менее обширных зон или сфер сети.

Выше описан только один из возможных вариантов реализации СУТ. Более подробный анализ имеющихся здесь возможностей и задач проведен в работах [4].

3.2.4. Понятие М-автомата

Пусть задана некоторая М-сеть μ (3.14). Совокупность конкретных реализаций каждого из элементов семерки (3.14) есть состояние М-сети. Алгоритм функционирования преобразовывает состояние М-сети в момент t в ее состояние в момент t+1. Алгоритм содержит следующие основные блоки:

- блок пересчета, выполняющий операции в соответствии с формулой пересчета (3.13); в этом блоке определяются возбужденности всех i-моделей М-сети в момент t+1;
- блок обучения, в котором в соответствии с характеристиками обучения определяются новые значения проходимостей связей и параметров возбужденности i-моделей;
- блок дополнения, или “роста”, М-сети; здесь в соответствии с характеристиками самоорганизации устанавливаются новые связи между i-моделями и формируются “спонтанные” возбуждения резервных элементов;
- блок СУТ, в котором производятся операции, реализующие алгоритм работы системы усиления - торможения;
- блок проверки логических условий; вид этих условий определяется отдельно для каждой конкретной задачи моделирования.

Функционирование М-сети обеспечивается многократным применением описанного алгоритма. Порядок выполнения различных блоков строго фиксируется и может быть частично изменен при построении конкретных моделей. Совокупность операций, выполняемых при однократном применении алгоритма, назовем тактом функционирования М-сети. За один такт, следовательно, осуществляется полное определение состояния М-сети в определенный момент дискретного времени.

Алгоритм функционирования М-сети будем в дальнейшем для краткости называть алгоритмом А, соответственно, пару $<\mu$, A> - М-автоматом. Такой автомат построен на основе М-сети и включает в себя алгоритм ее функционирования А.

Рассмотрим некоторый М-автомат $<\mu$, A>. Если μ задана в виде (3.13), будем считать такой М-автомат полным. Иногда оказывается целесообразным строить М-автоматы, в которых реализованы не все функции М-сети. В зависимости от полноты задания М-сети будем различать самообучающиеся М-автоматы:

$$\mu = <P, S, R, L, \varnothing, C, I> \qquad (3.14)$$

и необучающиеся М-автоматы:

$$\mu = <P, S, R, \varnothing, \varnothing, C, I> \qquad (3.15)$$

знак $\varnothing$ обозначает, что соответствующий элемент не вводится.

Алгоритм А в случае самообучающегося М-автомата не содержит блока 3, а в случае необучающегося - блоков 2 и 3.

Иногда выражение в М-сети отдельных (из множества) моделируемых функций или программ оказывается неэкономичным или связано со значительными техническими трудностями. В этих случаях указанные функции целесообразно описать некоторым специальным алгоритмом (т.е. построить их функциональную модель) и организовать его совместную работу и обмен информацией с алгоритмом А. М-автомат, часть функций которого выражена специальным алгоритмом, работающим совместно с алгоритмом А, назовем неполным М-автоматом. Неполный М-автомат является удобной формой сочетания в единой системе функциональных и структурных моделей.

М-автомат, алгоритм А которого не содержит блока СУТ, является вырожденным М-автоматом.

Построение М-автомата. Пусть поставлена задача построения модели некоторой функции φ и определена цель моделирования. Кратко рассмотрим этапы построения соответствующего М-автомата.

Прежде всего, собираются и систематизируются сведения из предметной области об этой функции φ. Если необходимо, проводятся новые исследования.

Исходя из целей моделирования и сведений п.1 определяется необходимый тип М-автомата. Если принимается решение о разработке неполного М-автомата, конструируется алгоритмическая модель соответствующих функций.

Выдвигается гипотеза о составе программ, участвующих в формировании моделируемой функции φ.

Исходя из целей моделирования, задаются "внешние" объекты и законы их взаимодействия с М-автоматом, то есть задается среда модели.

Определяется "уровень" моделирования. В соответствии с гипотезой п.3 фиксируется набор понятий, необходимый для описания на выбранном уровне.

В соответствии с гипотезой п.3 задается множество связей между i-моделям.

Определяются проходимости связей, вид и параметры характеристик i-моделей и связей. Для их уточнения могут понадобиться специальные эксперименты. Однако, как правило, они определяются эвристически.

Аналогично определяются (если необходимо) характеристики обучения и самоорганизации. При выполнении пп. 6-8 широко используются аналогии, сопоставления и т. п. Направляющим здесь является содержание гипотезы п.3. Успех моделирования во многом зависит от удачного выбора величин в пп. 6 и 7. Именно здесь, прежде всего, необходима дальнейшая систематизация, совершенствование и разработка методов эвристического моделирования.

Задается исходное состояние М-сети.

Задается алгоритм функционирования А.

М-автомат и его среда реализуются в виде действующих устройств или программ для ЭВМ.

3.2.5. Решение задачи выбора вида противодействия на М-сети

Как отмечалось выше (п.3.4.4.) М-сеть представляет собой аппарат моделирования человеческого мышления. Это свойство М-сетей проверено в ряде областей исследования [ ], доказана их адекватность поведению человека в конкретных приложениях.

Таким образом М-сеть представляет собой апробированный математический аппарат искусственного интеллекта и может успешно использоваться для решения задачи классификации НСД и выбора типа стратегии противодействия.

Зададим М-сеть μ в виде семерки (набора):

$$\mu = <P, S, R, L, F, C, I>,$$

где P - множество i-моделей; S - множество связей между i-моделями; R - группа характеристик самообучения; F - группа характеристик самоорганизации; C - начальное распределение проходимостей связей; I - начальное распределение возбуждений i-моделей.

Множество i-моделей P М-сети μ. Как было введено ранее, множество P i-моделей М-сети μ можно представить в виде объединения трех непересекающихся множеств i-моделей:

множества рецепторных i-моделей P^1;

множества внутренних i-моделей P^2;

множества i-моделей действия (результирующих i-моделей) P^3.

Упорядочив элементы множеств P^1 P^2 P^3 по возрастанию условных номеров получим соответствующие вектора imr, imi, ima.

Рецепторным i-моделям $imr = < imr^1, \ldots, imr^{n^r} >$ поставим в соответствие:

- элементы вектора Z;
- элементы вектора Z'.

С *внутренними i-моделями* $imi = < imi^1, \ldots, imi^{n^i} >$ будем ассоциировать:

- уровни классификационных признаков НСД Cl;

- вспомогательные i-модели;
- множество типов стратегий противника M.

К внутренним i-моделям также отнесем i-модели интегральной оценки состояния M-сети μ.

Результирующим i-моделям $ima = < ima^1, ..., ima^{n^a} >$ поставим в соответствие:

- множество типов стратегий противодействия L.

Множество связей S между i-моделями M-сети μ. Связи могут быть усиливающими и тормозными.

Связи между i-моделями внутри уровней классификационных признаков и между i-моделями разных классификационных уровней устанавливаются согласно RCl.

Остальные связи между i-моделями устанавливаются на основе экспертных оценок.

Первоначальные значения элементов множеств R, L, F, C, I M-сети μ устанавливаются также на основе экспертных оценок. Определенная таким образом M-сеть μ готова к работе (рис. 3.3).

Таким образом, результатом работы M-сети μ в нашем случае будет выделение СУТ i-модели действий l с наибольшим возбуждением Π_l^*.

При этом, Центр интегральной оценки M-сети μ должен определить соответствие полученного результата (стратегии противодействия) установленному критерию.

Предлагается следующий критерий.

$$\Delta\Pi_l \geq \Delta\Pi_l^{доп},$$

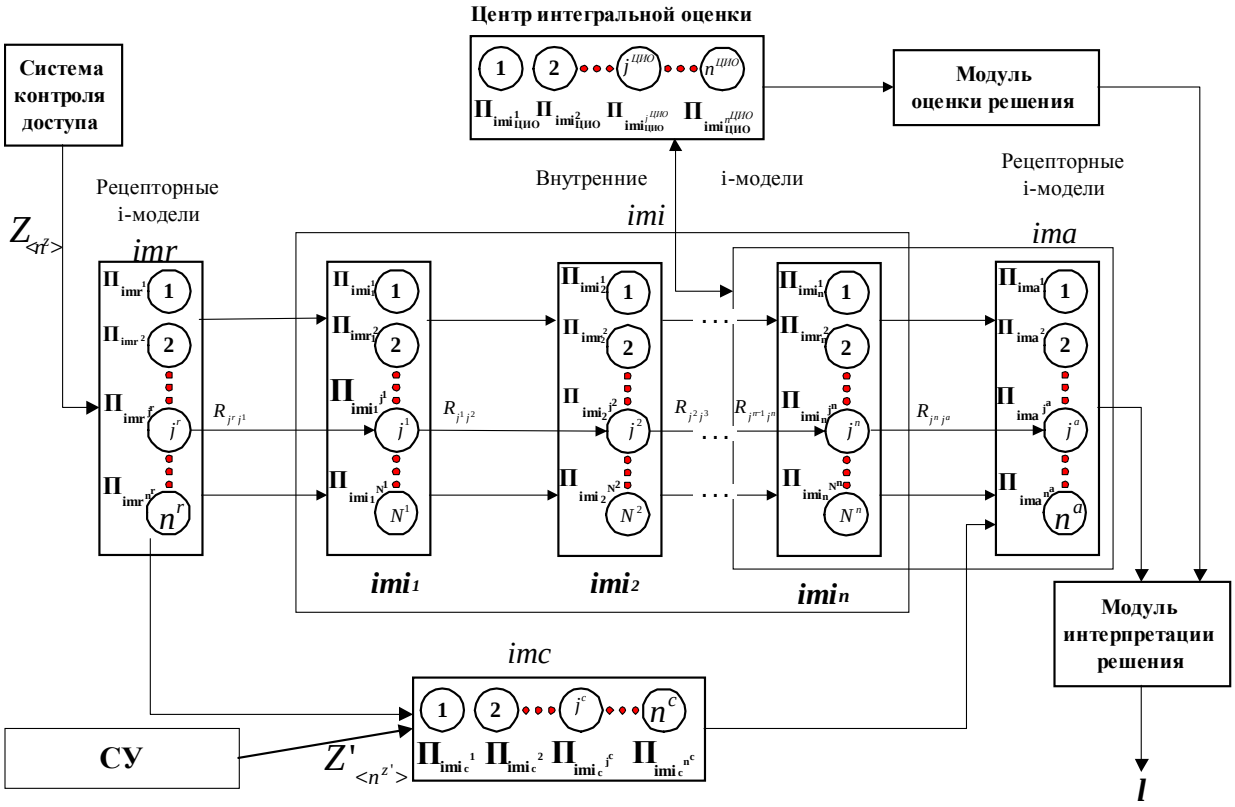


Рис. 3.3. Структура M-сети для решения задачи выбора стратегии интеллектуального противодействия

где

$\Delta\Pi_l = \min_{i\ i \neq l} (\Pi_l^* - \Pi_i)$; Π_l^*. Π_i – возбуждение i-моделей действий; i=$\overline{1, L}$;

$\Delta\Pi_l^{доп}$ - заданное значение критерия.

Если решение задачи выбора удовлетворительно, согласно выбранного критерия, то в качестве номера выбранного типа стратегии противодействия имеем ***l***.

Если решение неудовлетворительно в течение заданного времени $t^{зад}$, тогда принимается одно из следующих решений:

- привлечение эксперта-человека;
- уточнение вектора Z;
- уточнение вектора Z';
- принятие неудовлетворительного решения $\Pi_l^* \rightarrow l$;
- корректировка структуры М-сети μ ;
- корректировка начальных параметров М-сети.

Очевидно, что М-сеть должна быть построена таким образом, чтобы существовала возможность задавать достаточно большое значение $\Delta\Pi_l^{доп}$, что повышает адекватность (правильность) выбора типа стратегии противодействия.

Для этого М-сеть μ должна быть соответствующим образом настроена.

Рассмотрим небольшой пример, иллюстрирующий процесс построения М-автомата (задания М-сети и алгоритма функционирования М-автомата), реализующего выбор вида противодействия.

Итак, моделируемая функция ψ есть поведение автомата выбора вида ИП (АВИП).

Пусть противник (или процесс его представляющий) осуществляет в СОИ несанкционированные действия. Ограничим его действия лишь потоком запросов на получение доступа к определенной информации. Тогда множество типов стратегий противника представим в виде M={1, 2}, 1 – соответствует блокировка объекта СОИ (узла или приложения); 2 – доступ к защищаемой информации. Действия противника также характеризуются некоторой интенсивностью запроса. Спецификацию НСД представим кортежем из трех элементов $Z_C =< z_1, z_2, z_3 >$. Элемент z_1 – ценность информации, по отношению, к которой осуществляется НСД. Заметим, что в исходной спецификации НСД системой контроля доступа элемент z_1- это просто указание на конкретную информацию, а определение ее ценности – задача этапа выбора вида ИП. Но для упрощения мы принимаем, что ценность уже определена и при этом складывается из уровня секретности и степени важности информации (не тождественность этих понятий была показана выше). Элемент z_2 – важность объекта (узла/приложения, в исходной спецификации это также лишь указание на объект). Важность узла определяется положением соответствующего органа управления в системе управления, а также интенсивностью своего участия в информационном обмене посредством СОИ. Элемент z_2 – интенсивность запросов. Элементы вектора Z определенным образом приведены к единому масштабу значений и выражены в условных единицах.

Представим множество типов стратегий противодействия также в виде множества двух элементов L={1, 2}, где 1 – соответствует некоторой программе ИП на базе вполне определенного ложного объекта атаки (например, ложной БД); 2 – соответствует простому реагированию (например, отказу в доступе). Будем считать, что, во-первых, чем ценнее информация , по отношению к которой осуществляется НСД, тем важнее ее защитить и тем опаснее вести какие-то ни было интеллектуальные игры; во-вторых, чем важнее узел, тем больше необхо-

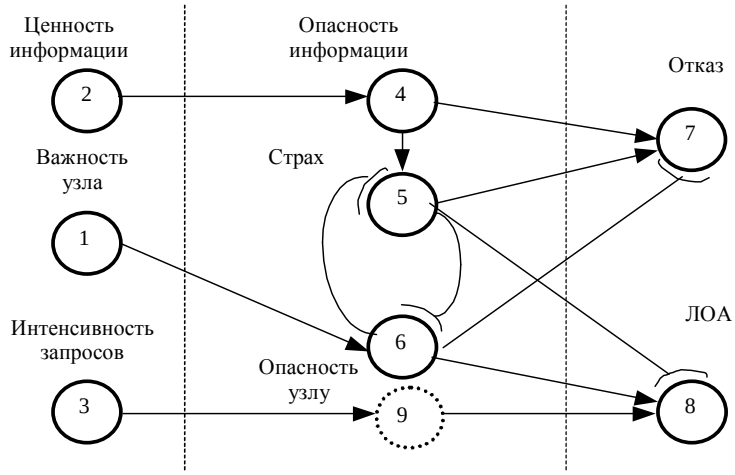


Рис. 3.4. Вариант архитектуры М-сети АВИП

димость оградить его от потока запросов, блокирующих его работу.

Для того, чтобы упростить модель и сократить число уровней в М-сети, стратегии 1 противника поставим в соответствие стратегию 1 противодействия; а стратегии 2 противника – стратегию 2 противодействия. То есть, по нашей М-сети определение стратегии противника автоматически определяет стратегию ИП. Построим М-сеть автомата выбора интеллектуального противодействия (АВИП). *Рецепторные i-модели* – элементы вектора спецификации НСД:

- важность узла.
- ценность информации.
- интенсивность запросов.

Внутренние i-модели (ограничимся одним классификационным уровнем):

- опасность, грозящая информации.
- опасность, грозящая узлу.

Дополнительная внутренняя i-модель, ассоциированная с уровнем классификации:

- страх (мы имели в виду «страх» за потерю информации).

i-Модели действий, связанные с определенными стратегиями противодействия из множества М:

- отказ (простое реагирование).
- ложный объект атаки (ЛОА) (интеллектуальное противодействие с использованием некоторого ЛОА).

Граф М-сети АВИП показан на рис.3.4. Так как работа АВИП тактирована в соответствии с дискретными моментами времени t, то для синхронизации работы М-сети можно ввести дополнительные (пустые) i-модели для устранения связей «через уровни». На рис. 3.4. такая i-обозначена пунктиром, при этом должно выполняться условие $\Pi_9^t = \Pi_3^{t-1}$.

Далее работа по формированию АВИП ничем не отличается от традиционной [4]. Блок-схема алгоритма функционирования М-автомата представлена на рис. 3.5. и также ничем не отличается от традиционной [4].

Таким образом, главная задача разработчика заключается в том, чтобы согласно предложенному методу разработать протокол взаимодействия с СКД, построить классификацию НСД и переложить ее на М-сеть. Задача эта достаточно сложная, но выполнимая, благодаря тому, что М-сеть обладает уникальным свойством легкой наращиваемости без перестройки всей М-сети. Предложенный же в данном параграфе и представленный ниже генетический алгоритм настройки М-сети призван способствовать корректному функционированию М-автомата выбора вида интеллектуального противодействия.

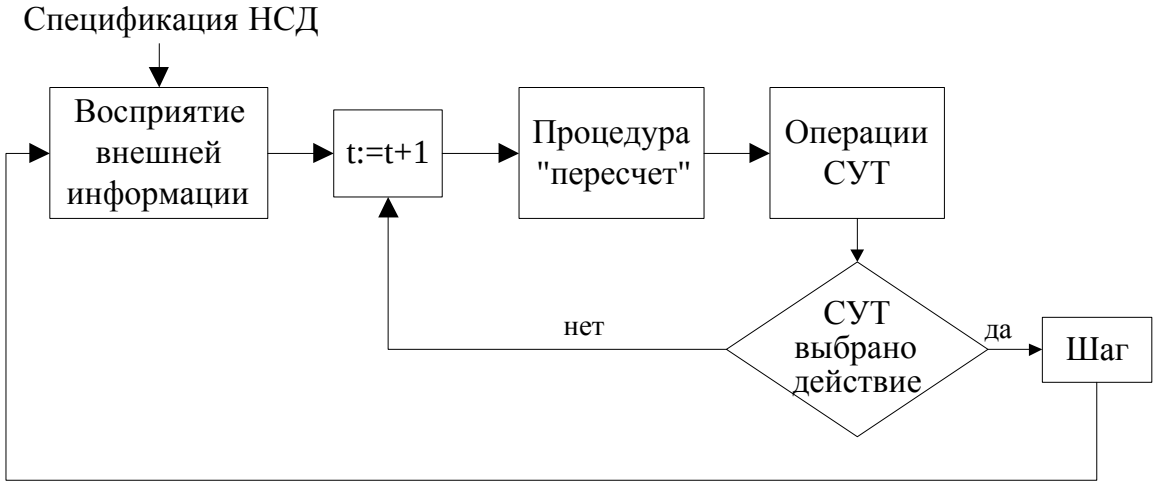


Рис. 3.5. Блок-схема алгоритма функционирования М-автомата

3.3. Генетический алгоритм настройки М-сети.

Выше приведен порядок построения М-сети для выработки типа стратегии противодействия.

Результат работы М-сети μ будет удовлетворительным только в том случае, если ей удается выделить соответствующий конкретной попытке НСД тип стратегии противника.

При экспертной настройке параметров М-сетей не всегда удается добиться ее правильной работы. В то же время, подстройка параметров М-сети экспертным путем потребует значительных затрат времени и сил и не обязательно будет успешной.

К сожалению, в литературе по теории М-сетей [4] не приводится эффективных методов настройки параметров М-сети.

В общем случае можно рассмотреть задачу обучения М-сети на конкретных имеющихся примерах.

Задача обучения М-сети на конкретном примере состоит в следующем.

Дано: вектор **Z**, вектор $\mathbf{\Pi_{<M>}}$ - вектор возбужденностей i-моделей, ассоциированных с типами стратегий противника (элементами вектора М).

Значения Π_i – компонент вектора $\Pi_{<M>}$ теоретические и зависят от вероятности, с которой можно утверждать, что в наблюдаемой в прошлом попытке НСД противник применил стратегию типа m.

Обозначим $\mathbf{\Pi'_{<M>}}$ - вектора возбужденностей i-моделей, ассоциированных с типами стратегий противника, полученный путем инициации М-сети вектором Z.

Тогда задача обучения М-сети μ на примере состоит в том, чтобы минимизировать различия вектора $\Pi_{<M>}$ и вектора $\Pi'_{<M>}$ поэлементно.

Формально это выглядит следующим образом.

$$\min_{R,L,F,C,I} \sum_{i=1}^{L} | \Pi_i - \Pi'_i |$$

Решение данной задачи оптимизации может быть ограничено решением сатисфакционной задачи.

$$\sum_{i=1}^{L} | \Pi_i - \Pi'_i | \le \Delta_{\Pi}^{доп}$$

где $\Delta_{\Pi}^{доп}$ - максимально допустимое значение суммы модулей разностей компонент векторов $\Pi_{<M>}$ и $\Pi'_{<M>}$.

Но даже в этом случае методом экспертной оптимизации по всем параметрам М-сети (компонентам множеств R, L, F, C, I) задачу решить сложно, а при достаточно большой размерности М-сети – практически невозможно за приемлемое время.

При этом в настоящее время все большее применение для решения оптимизационных задач большой размерности используются генетические алгоритмы оптимизации. Ряд несомненно успешных работ в области генетических алгоритмов оптимизации [27, 29, 99, 105, 96, 97], уже позволяют считать этот аппарат достаточно эффективным и апробированным и применить в данной работе для настройки М-сети.

3.3.1. Общие вопросы теории генетических алгоритмов оптимизации

С математической точки зрения ***генетические алгоритмы оптимизации (ГАО)*** - это разновидность алгоритмов оптимизации, объединяющая черты вероятностных и детерминированных оптимизационных алгоритмов [100].

Главная трудность в синтезе генетического алгоритма оптимизации заключается в построении генетического кода, представляющего структуру различных процессов, подобно тому как ДНК (дезоксирибонуклеиновая кислота) представляет структуру организма (например, человека). Обычно генетический код (“хромосома”) **s** составляется в виде цепочек единиц и нулей, каждая из которых кодирует наличие или отсутствие одного из свойств у данного индивидуума (рис.3.6).

$s = <f_1, f_2, ..., f_k, ..., f_n>$

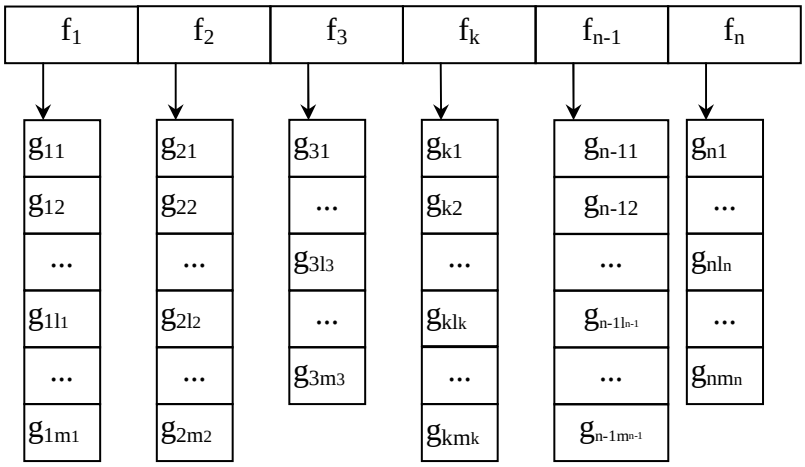


Рис. 3.6. Структура хромосомы (генетического кода) моделируемого процесса или объекта

На рис. 3.6. под $f_k <g_{k1}, ..., g_{klk}, ..., g_{kmk}>$ понимается k-ое свойство моделируемого объекта (процесса), при этом $f_k \in F_k$. Множество F_k - область значений k-ого свойства моделируемого объекта (процесса). Элемент g_{ij} может принимать значение 0 или 1, то есть $g_{ij} \in G$ ($G=\{0, 1\}$). Таким образом, $F_k=G^{mk}=\{< g_{k1}, ..., g_{kj}, ..., g_{kmk}> | g_{kj} \in G, j=\overline{1, m_k}\}$.

Множество P, на котором определены все возможные индивидуумы моделируемого объекта (процесса), можно записать следующим образом:

$$P = F_1 \times F_2 \times ... \times F_i \times ... \times F_{n-1} \times F_n = \{<f_1, f_2, ... f_i, ... f_{n-1}, f_n> | f_1 \in F_1, f_2 \in F_2, ... f_i \in F_i, ... f_{n-1} \in F_{n-1}, f_n \in F_n, i=\overline{1, n}\},$$

где n - количество рассматриваемых свойств моделируемого объекта (процесса).

Множество объектов (процессов) $S \in P$, по аналогии с живым миром, называется популяцией эволюционирующих объектов (процессов).

Поиск оптимальных решений заключается в поиске двоичных цепочек из всего их многообразия. Отдельные области в пространстве решений определяются по наличию нулей и единиц в определенных позициях цепочек. Например, множество всех цепочек, начинающихся с 1, представляет собой некоторую область в пространстве решений. Одним из традиционных способов решения задач оптимизации в таких пространствах являются алгоритмы наискорейшего спуска. Однако при значительной сложности оптимизационных задач применение этих алгоритмов требует значительных вычислительных ресурсов. Генетические же алгоритмы покрывают функциональный “ландшафт” задачи сетью [88]. Множество цепочек в развивающейся популяции одновременно тестирует множество его областей. Точнее частота, с которой генетический алгоритм тестирует различные области, пропорциональна близости к оптимальному решению, что повышает вероятность отыскания оптимального решения. Эта способность генетических алгоритмов концентрировать внимание на наиболее перспективных областях пространства решений является прямым следствием их способности комбинировать цепочки, содержащие частные решения. Сначала оценивается каждая цепочка популяции, чтобы определить оптимальность кодируемого ей решения. На втором этапе получившие наибольшие оценки цепочки спариваются. Две цепочки располагаются друг пе-

ред другом, на линии их соприкосновения случайным образом выбирается точка, и части, расположенные слева от этой точки, меняются местами, образуя двух потомков: один из них содержит символы первой цепочки вплоть до точки перекрещивания, а далее - символы второй; другой же потомок содержит дополнительный набор символов. Этот процесс называется *кроссинговером* и представлен на рис.3.7.

После этого потомки возвращаются в популяцию, но не замещают родителей, а замещают цепочки с наихудшей оценкой кодируемого ими решения, которые отбрасываются в каждом поколении, так что численность популяции со временем остается одинаковой. Формально определим кроссинговер как 5-арное отношение, заданное на множествах S, P и I:

$$r_c = <S, P, I, R_c>, \quad (3.16)$$

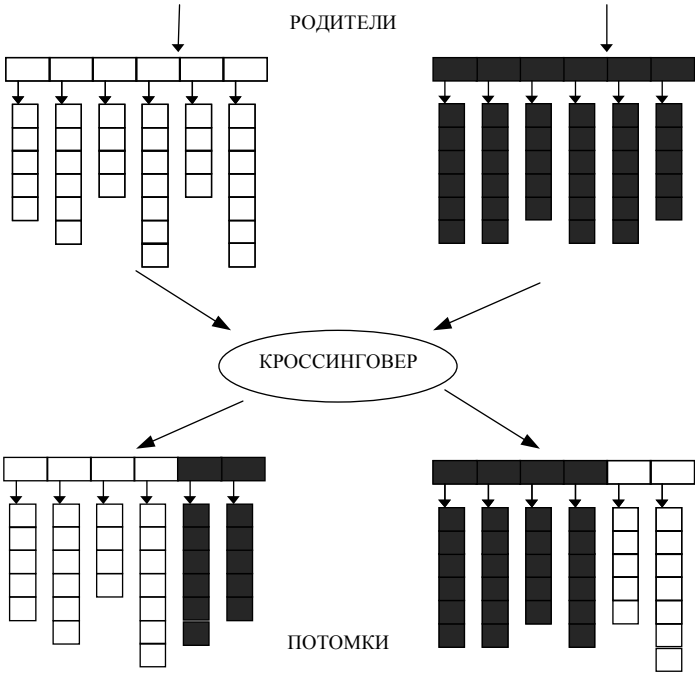


Рис. 3.7. Оператор кроссинговера

где $I = \{1, 2, ..., n\}$, $i \in I$; $R_c \subseteq S^2 \times P^2 \times I$ - график отношения кроссинговера, представляющий собой множество пятерок вида <хромосома 1 (s_1), хромосома2 (s_2), позиция кроссинговера (i), новая хромосома 1 (s'_1), новая хромосома 2 (s'_2)>.

Таким образом, если $s_1 = <f^1_1, f^1_2, ..., f^1_k, ..., f^1_n>$, $s_2 = <f^2_1, f^2_2, ..., f^2_k, ..., f^2_n>$ и i=k, то $s'_1 = <f^1_1, f^1_2, ..., f^2_k, ..., f^2_n>$, $s'_2 = <f^2_1, f^2_2, ..., f^1_k, ..., f^1_n>$.

На третьем этапе *мутации* (рис. 3.8.б) изменяют небольшой участок цепочек: примерно один на каждые 10000 символов переключается с 0 в 1 или наоборот. Сами по себе мутации обычно не дают прогресса в поиске решения, но они страхуют от возникновения однородной популяции, не способной к дальнейшей эволюции [ ].

Формально определим мутацию как тернарное отношение, заданное на множествах F_k и J_k:

$$r_m = <F_k, J, R_m>, \quad (3.17)$$

где $J_k = \{1, 2, ..., m_k\}$, $j \in J_k$;

$R_m \subseteq F_k^2 \times J_k$ - график отношения отношения мутации, представляющий собой множество троек вида < значение свойства (f_k), позиция мутации (j), новое значение свойства (f'_k) >.

Таким образом, если $f_k = <g_{k1}, g_{k2}, ..., g_{kl}, ..., g_{kmk}>$ и j=l, то $f_k = <g_{k1}, g_{k2}, ..., \overline{g_{kl}}, ..., g_{kmk}>$.

$$\overline{g_{ij}} = \begin{cases} 1, \text{ если } g_{ij} = 0; \\ 0, \text{ в противном случае.} \end{cases}$$

Генетический алгоритм интенсивно исследует целевые области пространства решений, поскольку в результате многократного размножения и скрещивания в этих областях скапливается все большее количество цепочек. В качестве родительских кодовых последовательностей алгоритм выбирает наилучшие цепочки, и поэтому цепочки с показателями выше среднего (они то и попадают в целевые области) в следующем поколении будут иметь больше потомков.В действительности численность цепочек в данной области пространства возрастает со скоростью, пропорциональной статистической оценке жизнеспособности этой области. Следуя статистическим методам, необходимо было бы оценивать десятки пробных цепочек из миллионов областей, чтобы определить средние показатели каждой области. Генетический алгоритм достигает тех же результатов со значительно меньшим количеством

цепочек и практически не производя никаких вычислений. Это обусловлено тем, что одна цепочка принадлежит сразу всем областям, в которых присутствует любой из составляющих ее битов. Например, цепочка 11011001 одновременно принадлежит областям 11****** (значение бита * безразлично), 10000001, **0**00* и так далее. Наиболее крупные области (содержащие большое число неопределенных битов) обычно тестируются значительной частью всех цепочек, составляющих популяцию. Так, генетический алгоритм манипулирующий популяцией в несколько тысяч цепочек, на самом деле тестирует немного большее количество областей целевого пространства. Этот неявный параллелизм обеспечивает генетическому алгоритму его главное преимущество по сравнению с другими процессами поиска решений задач [88].

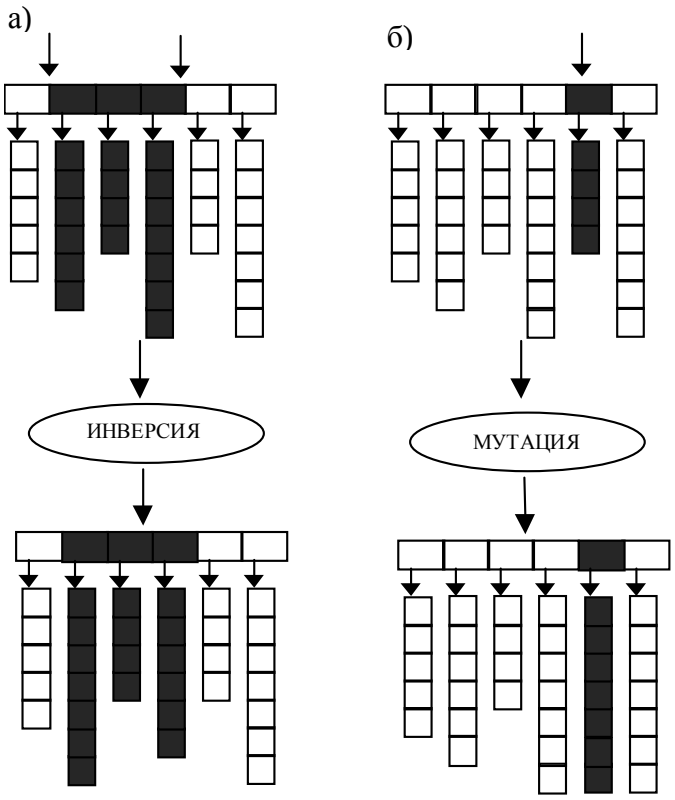


Рис. 3.8. Операторы инверсии и мутации

Наличие механизма скрещивания цепочек усложняет эффект неявного параллелизма. Цель скрещивания в генетическом алгоритме заключается в том, чтобы протестировать новые области, вместо того, чтобы тестировать одну и ту же цепочку в новых поколениях. Но этот процесс может переместить потомство из одной области в другую, в результате чего частота тестирования различных областей отклоняется от строго пропорциональной зависимости от среднего показателя. Это отклонение замедляет процесс эволюции.

Вероятность того, что потомство двух цепочек покинет родительскую область, зависит от расстояния между единицами и нулями, которые определяют границы этой области. Потомство цепочки, тестирующей область 10****, например, выйдет за пределы этой области только в том случае, если перекрещивание начнется во второй позиции цепочки: в одном случае из пяти для данной цепочки. Потомство цепочки, тестирующей область 1****1, имеет вероятность выйти из этой области независимо от того, где произойдет перекрещивание цепочек.

Близко соседствующие единицы и нули, определяющие пределы области, называются *компактными строительными блоками (КСБ)*. У них наибольшая вероятность сохраниться после кроссинговера в неизменном виде и, таким образом, распространиться в будущих поколениях со скоростью, пропорциональной средней эффективности цепочек, которые будут содержать в себе эти блоки. Хотя механизм размножения, включающий скрещивание, не дает возможности тестировать все области с частотой, пропорциональной их качеству, он обеспечивает ее для всех областей, которые определяются КСБ. Количество же КСБ в популяции цепочек намного превосходит число самих цепочек, благодаря чему генетический алгоритм все же проявляет свойства неявного параллелизма.

Операция, называемая в генетике живых организмов *инверсией*, время от времени меняет расположение генов таким образом, что гены, расположенные у родительских цепочек далеко друг от друга, могут стать соседями в потомстве (рис.3.8. а).

Формально определим инверсию как кватернарное (4-арное) отношение, заданное на множествах S, P и I:

$$r_i = <S, P, I, R_i>, \qquad (3.18)$$

где $I = \{1, 2, ..., n\}$, $i \in I$;

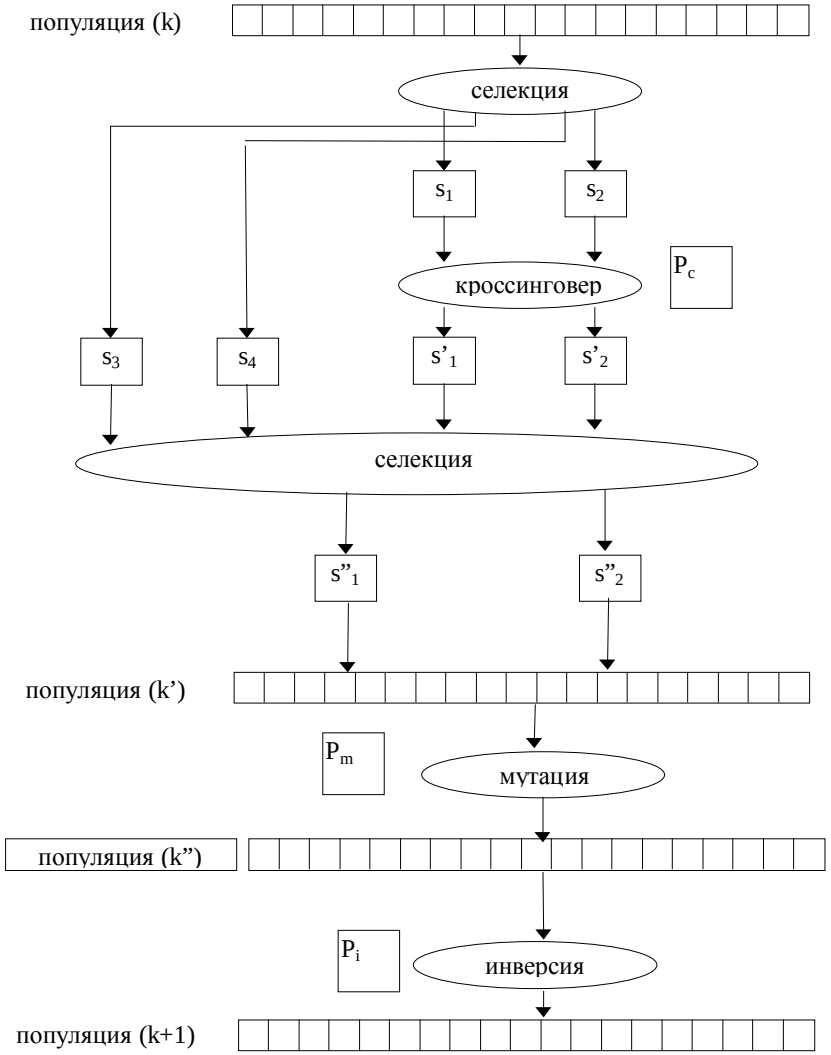
$R_i \subseteq S \times P \times I^2$ - график отношения инверсии, представляющий собой множество четверок вида <хромосома (s), позиция 1 инверсии (i_1), позиция 2 инверсии (i_2), новая хромосома (s')>.

Таким образом, если $s = <f^1_1, f^1_2, ..., f^1_k, f^1_{k+1},..., f^1_{l-1}, f^1_l, ..., f^1_n>$, $i_1=k$, $i_2=l$, то $s' = <f^1_1, f^1_2, ..., f^1_l, f^1_{l-1}, ...,f^1_{k+1}, f^1_k,..., f^1_n>$.

Инверсия, по сути, равносильна переопределению КСБ, в результате которого он становиться более компактным и менее уязвимым для кроссинговера. Если КСБ определяет область с высокой средней эффективностью, то более компактная версия блока замещает менее компактную, поскольку теряет меньше потомства в результате ошибок, возникающих при воспроизведении. Как результат система адаптации, использующая инверсию, может обнаруживать и отдавать предпочтение компактным версиям полезных блоков.

Принцип работы генетического алгоритма представлен на рис.3.9.

Неявный параллелизм генетического алгоритма позволяет ему тестировать и использовать большее количество областей в пространстве решений, работая с относительно небольшим числом цепочек. Этот механизм неявного параллелизма позволяет генетическому алгоритму справляться с нелинейными задачами, то есть в тех случаях, когда эффективность цепочки, содержащей два полезных КСБ, может оказаться значительно выше, чем сумма эф-



s_1, s_2 - две наилучшие с точки зрения целевой функции кодовые цепочки; s_3, s_4 - наихудшие кодовые цепочки; p_c, p_m, p_i - вероятности операторов кроссинговера, мутации и инверсии;
1, 2, ...k, k+1, ... - номер цикла работы генетического алгоритма;

Рис. 3.9. Функционирование генетического алгоритма

фективностей каждого из этих КСБ. Кроме этого, генетический алгоритм решает проблему, которая долгое время не поддавалась решению при помощи других методов: отыскание баланса между исследованием и эксплуатацией найденных стратегий. Например, как только генетический алгоритм находит оптимальную стратегию поведения в некоторой игре, возникает соблазн немедленно воспользоваться ею. Но это может привести к нежелательным последствиям, поскольку эксплуатация найденной стратегии делает открытие действительно новаторских стратегий уже маловероятным. Прогресс достигается путем апробирования новых, рискованных вариантов. Поскольку многие рискованные варианты оказываются неудачными, исследование ведет за собой снижение эффективности ведения игры. Определить, в какой степени настоящее можно заложить на будущее, - это классическая проблема для всех систем, которые обучаются и адаптируются [88].

Подход генетического алгоритма к этой проблеме основан на кроссинговере генетических цепочек. Хотя перекрещивание может помешать эксплуатации КСБ, разбив его, этот процесс рекомбинации тестирует КСБ в новых сочетаниях и в новых контекстах. При скрещивании образуются новые выборки областей, обладающих качеством выше среднего, подтверждая или опровергая оценки, полученные в предыдущих тестированиях. Кроме того, при перекрещивании разбивается КСБ, образуется новый блок, позволяющий алгоритму тестировать области, по которым ранее отсутствовали выборки.

Более подробно теория генетических алгоритмов оптимизации изложена в работе [100, 103, 88].

Программная реализация генетического алгоритма, реализующая все основные операции (селекция, инверсия, кроссинговер, мутация), использованная в данной работе для демонстрации возможностей генетических алгоритмов оптимизации, а также для получения количественных оценок ряда зависимостей, приведена в Приложении.

3.3.2. Особенности построения генетического алгоритма настройки М-сети

Описанные выше свойства ГАО позволяют применить их для решения поставленной нами задачи – настройки параметров М-сети μ, путем ее обучения на конкретных примерах.

Далее мы рассмотрим генетический алгоритм настройки М-сети[ ]. Алгоритм включает несколько основных операций, приведенных ниже.

Описание исходных данных:

- *определить популяцию эволюционирующих объектов;*
- *определить генетическую последовательность, кодирующую структуру объекта в популяции;*
- *определить целевую функцию;*
- *определить вероятности генетических операторов.*

Селекция:

- вычислить значение целевой функции для каждой кодовой цепочки популяции;
- выбрать две наилучшие кодовые цепочки;
- выбрать две наихудшие кодовые цепочки.

Применение генетических операторов:

- к выбранным в результате селекции наилучшим кодовым цепочкам применить оператор кроссинговера;
- полученными в результате применения кроссинговера новыми кодовыми цепочками заменить наихудшие цепочки в популяции.
- ко всей популяции применить оператор мутации;
- *ко всей популяции применить оператор инверсии.*

Описанная последовательность действий выполняется до тех пор, пока не будет достигнуто необходимое значение заданной целевой функции.

Рассмотрим каждый шаг алгоритма.

Описание популяции эволюционирующих объектов. Описание исходных данных начнем с определения популяции эволюционирующих объектов. В качестве одного такого объекта примем М-сеть μ с определенными структурными характеристиками (например, разрядность блока внутренних i-моделей - n^i, матрицы связей - n^s, количество уровней возбужденности i-модели - k).

Тогда в качестве популяции можно выбрать множество всех n^i+n^s –разрядных М-сетей. Процесс эволюции в такой популяции представляется как процесс адаптации (настройки) М-сети на внешнее воздействие, воспринимаемое через блок рецепторных i-моделей размерностью n^r.

Описание генетической последовательности, кодирующей структуру объекта популяции. Как отмечалось выше, разработка генетической последовательности для кодирования структуры объекта является одним из самых сложных шагов в описании исходных

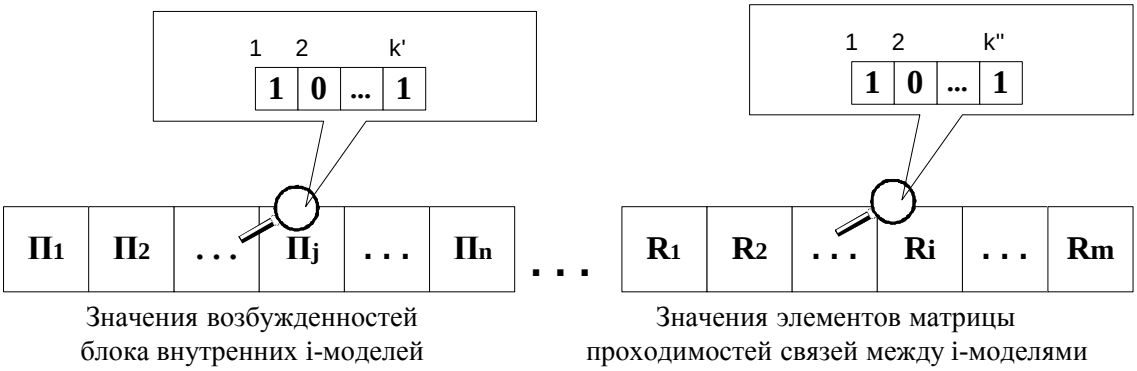


Рис. 3.10. Генетическое кодирование структуры М-сети

данных для решения задачи посредством генетических алгоритмов.

Прежде всего, необходимо изменить архитектуру М-сети μ таким образом, чтобы представить блок i-моделей и матрицу связей между i-моделями в бинарной форме. Для этого в описании М-сети заменим десятичное описание возбужденностей i-моделей и проходимостей связей между i-моделями на их двоичный аналог.

Теперь структуру М-сети можно представить генетическим кодом в виде последовательности двоичных цепочек (рис. 3.10.). Относительное положение в этом коде значений отдельных структурных единиц подбирается экспериментально и зависит от структуры М-сети, вероятности генетических операторов и некоторых других факторов. Ввиду того, что при больших значениях n^i и n^s длина кодовой цепочки и количество возможных цепочек также велико, в этих случаях целесообразно использовать последовательную многоступенчатую ГА-оптимизацию. Можно рассмотреть, например, следующие ступени ГА-оптимизации:

- ГА-оптимизация по возбужденностям i-моделей;
- ГА-оптимизация по значениям проходимостей связей между i-моделями.

При этом количество рассматриваемых цепочек уменьшится с $2^{(n^i+n^S)\times k}$ до $2^{n^i\times k}+2^{n^S\times k}$.

Аналогично можно использовать последовательную многоступенчатую ГА-оптимизацию внутри каждого из двух рассмотренных шагов (например, по слоям возбужденностей внутренних i-моделей), получая при этом многоуровневую ГА-оптимизацию.

Определение целевой функции.

В ходе экспериментов установлено, что предпочтительным является выбор целевой функции в виде

$$\sum_{i=1}^{L} | \Pi_i - \Pi'_i |$$

Причем значения возбужденностей последнего уровня внутренних i-моделей Π'_1 являются результатом функционального преобразования F, реализуемого М-сетью.

Условия останова ГАО:

- если $\sum_{i=1}^{L} | П_i - П'_i | \le \Delta_П^{доп}$;

- если $n_Ц = n_Ц^{max}$ ($n_Ц$ - число циклов ГАО; $n_Ц^{max}$ - максимальное число циклов ГАО);

- если в течение заданного числа циклов ГАО $n_Ц^{зад}$ значение целевой функции изменяется на величину меньшую, чем на $\Delta^{зад}$.

Достоинством целевой функции построенной на основе функционального преобразования F М-сети является то, что она рассчитывается на основе конкретной реальной М-сети, а не является результатом экспертного конструирования.

Определение вероятностей применения генетических операторов. Вероятности применения генетических операторов определяют экспериментально. Они зависят от условий задачи (численность популяции, длина генетической цепочки и др.). При выборе диапазона вероятностей следует учитывать, что для биологических объектов они лежат в следующих пределах:

- для мутации от 10^{-3} до 10^{-5};
- для инверсии от 10^{-2} до 10^{-4}.

Закон распределения вероятностей применения для кроссинговера, мутации и инверсии также выбирается экспериментально, но в общем случае используется нормальный закон распределения.

Все остальные действия по применению генетического алгоритма для настройки М-сети являются классическими, с точки зрения теории генетических алгоритмов, и подробно не рассматриваются.

3.4. Анализ результатов решения задачи выбора вида противодействия в корпоративной сети обмена информацией

Многие, наверное, обратили внимание на то, что решаемая с помощью предложенного метода задача выбора вида интеллектуального противодействия, в части классификации НСД и стратегии противника, тяготеет к задачам распознавания образов. Поэтому результативность и оперативность предложенного метода можно сравнить с соответствующими характеристиками существующих методов распознавания образов [62].

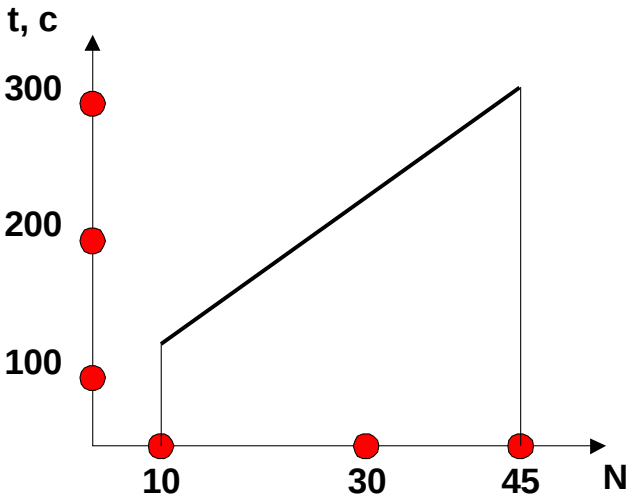
В теории распознавания образов под ***образом*** понимают множество явлений, объединяющихся общими свойствами.

Каждое конкретное явление, входящих в образ, в теории распознавания принято называть *изображением*.

Процессу распознавания предшествует процесс обучения, во время которого система знакомится с некоторым количеством явлений (изображений), зная заранее об их принадлежности к определенному образу. Ранее мы подробно рассмотрели такие достоинства генетических алгоритмов оптимизации, как способность концентрироваться на наиболее перспективных областях пространства решения и неявный параллелизм, позволяющий тестировать большее количество областей целевого пространства, чем численность популяции (генетических цепочек). Эти свойства обеспечивают ГАО его главные преимущества по сравнению с другими процессами поиска решения. Для сравнительного анализа с ГАО был выбран широко используемый метод оптимизации – метод наименьших квадратов [14]. Программная реализация ГАО приведена в Приложении 1. Для МНК использована его стандартная программная реализация, приведенная в работе [14]. Проведенные эксперименты по настройке (оптимизации параметров) М-сети с помощью ГАО и МНК позволили определить зависимость расчетного времени (характеризующего сложность оптимизации) от размерности задачи (количества параметров М-сети). Данная зависимость для каждого алгоритма представлена на рис.3.11. Из этих графиков видно, что для небольших размерностей задачи

оптимизации программа, реализующая МНК выглядит более предпочтительной, с точки зрения времени, нежели программа, реализующая ГАО. Но с другой стороны, зависимость времени t от размерности N для ГАО близка к линейной, что намного лучше, чем аналогичная зависимость для МНК, которая, как минимум, квадратичная. Это дает основание полагать, что для больших размерностей задачи оптимизации ГАО предпочтителен для использования. Именно такой задачей с большой размерностью и является настройка М-сети для классификации НСД и выбора вида противодействия, так что разработка генетического алгоритма настройки М-сети явилась вполне обоснованной

а) генетический алгоритм



а) метод наименьших квадратов

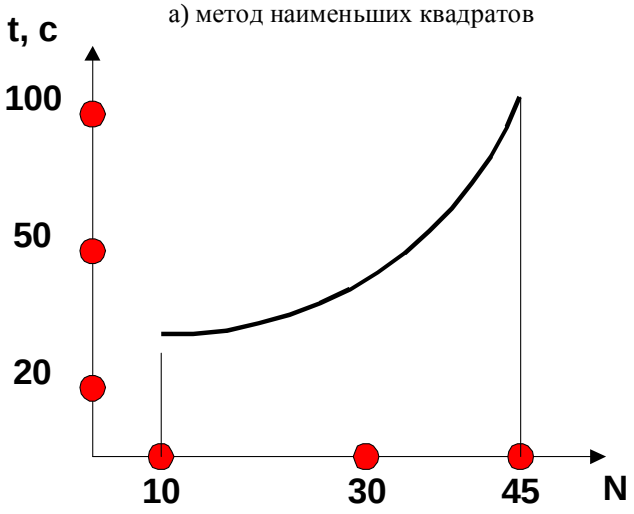


Рис. 3.13. Зависимость времени вычислений от размерности задачи

3.5. Метод выбора зон интеллектуального противодействия, основанный на перераспределении нагрузки в сети обмена информацией

Если метод, описанный в п. 3.6., разработан для классификации несанкционированных действий в СОИ и выбора вида противодействия, то метод, предлагаемый в данном параграфе должен обеспечивать выбор зоны интеллектуального противодействия.

Под *зоной* СОИ j будем понимать зону СОИ, соответствующую j-ому узлу базовой СОИ.

Базовая СОИ – это ядро СОИ, которое объединяет в одно целое локальные вычислительные сети (ЛВС), отдельные ЭВМ, терминалы и другие периферийные устройства, обеспечивая их взаимодействие между собой на основе единой системной идеологии, реализуемой в сетевых протоколах.

Базовая СОИ в зависимости от размеров обслуживаемой ею территории, числа выделенных географических зон и необходимого числа узлов коммутации в каждой из них для обслуживания зональной нагрузки может иметь в наиболее типичных ситуациях двух- или трехуровневую структуру.

Очевидно, что базовая СОИ (БСОИ) является наиболее критичной частью ИВС, с точки зрения нагрузки в ее линиях связи.

Под *нагрузкой линии связи между i-м и j-м узлами сети* будем понимать интенсивность λ_{ij} трафика (потока) x_{ij}, проходящего по линии (i, j).

В базовой СОИ можно выделить: λ_{ij} (полезная нагрузка) – интенсивность информационного потока x_{ij}; $\lambda_{ij}^{ПрОА}$ (нецелевая нагрузка) – интенсивность информационного потока Противник – Объект атаки (Пр-ОА) $x_{ij}^{ПрОА}$.

Под *зоной противодействия* будем понимать зону СОИ, в которой расположен ложный объект атаки.

При расположении ЛОА в определенной зоне СОИ, изменяются и потоки пакетов в базовой СОИ. В частности, потоки типа Противник – Объект атаки (Пр-ОА) преобразуются в потоки типа Противник – Ложный объект атаки. То есть потоки Пр-ОА имеют конечную точку j^+, отличную от первоначальной конечной точки j.

Так как в общем случае противник в некоторый $k \in \tau$ дискретный момент времени может реализовывать информационные атаки из разных зон СОИ в отношении объектов, в свою очередь расположенных в разных зонах СОИ, то и выбор стратегии противодействия, используя описанный в п. 3.6 метод, в каждом конкретном случае может быть различным.

Поэтому результат решения задачи выбора вида ИП целесообразно представить в виде матрицы $\mathbf{U}^{(k)}_{<N\times N>}$. Элемент $u_{ij}^{(k)}$ является элементом матрицы $\mathbf{U}^{(k)}_{<N\times N>}$ и определяет выбранный тип стратегии противодействия противнику, осуществляющему атаку из зоны i по отношению к объекту атаки, расположенному в зоне j СОИ. При этом $u_{ij}^{(k)} \in L$.

Таким образом, прежде чем осуществить противодействие выбранного вида, необходимо оптимизировать, с точки зрения выбранного показателя эффективности управления нагрузкой, направление потоков $x_{ij}^{ПрОА}$ (согласовать их нагрузку на базовую сеть с имеющимся резервом пропускной способности), то есть определить их конечную точку в базовой сети (БС), из зоны которой и будет в дальнейшем осуществляться противодействие с применением ложного ОА.

В качестве показателя эффективности (согласно поставленной в п. 1.4 задачи исследования) выбрана сумма коэффициентов недоиспользования пропускных способностей ветвей связи

$$D = \sum_{i=1}^{N}\sum_{j=1}^{N}\left(1 - \frac{\lambda_{ij}^{\sum}}{K \bullet \mu_{ij}}\right), \qquad (3.19)$$

Сформулируем задачу разработки метода выбора зон интеллектуального противодействия.

3.5.1. Постановка задачи

Исходные данные.

Структура базовой сети G (N, N_1), где N – множество узлов (центров коммутации), а

L– множество ветвей связи.

Матрица интенсивностей потоков пакетов разных видов информации $\Lambda^{(k)}_{[N\times N]}$. Элемент $\lambda_{ij}^{(k)}$ является элементом матрицы $\Lambda^{(k)}_{[N\times N]}$ и представляет собой поток пакетов от узла i к узлу j в $k \in \tau$ момент времени.

Матрица пропускных способностей ветвей связи $M_{N\times N} = \left\| \mu_{ij} \right\|$, где μ_{ij} – пропускная способность связи между узлами i и j.

Матрица коэффициентов готовности ветвей $K_{N\times N} = \left\| K_{ij} \right\|$, характеризующая надежность ветвей связи.

Вероятность $P_в$ выведения противником ветви связи из строя в результате неинформационного воздействия.

Матрицы интенсивностей потоков служебных пакетов: на канальном уровне – матрица $A_{N\times N} = \left\| a_{ij} \right\|$; на сетевом уровне – матрица $B_{N\times N} = \left\| b_{ij} \right\|$; на транспортном уровне – матрица $D_{N\times N} = \left\| d_{ij} \right\|$.

В сети используется адаптивный алгоритм маршрутизации V.

Требования к качеству обслуживания заданы требуемым временем доставки информации $t_{TP} = \nu_{TP}^{-1}$, допустимой вероятностью потерь пакетов $P_{ПИ}^{доп}$, требуемой вероятностью своевременной доставки пакетов $P_{СД}^{TP}$.

Матрица интенсивностей потоков пакетов типа Пр-ОА $\Lambda_{N\times N}^{ПрОА\,(k)}$. Элемент $\lambda_{ij}^{(k)ПрОА}$ является элементом матрицы $\Lambda_{N\times N}^{ПрОА}$ и представляет собой поток пакетов типа Пр-ОА от узла i к узлу j в $k \in \tau$ момент времени.

Необходимо разработать метод, позволяющий в дискретный момент времени $k \in \tau$ определить значения элементов матрицы $J_{[N\times N]}^{+}$, исходя из критерия максимизации суммы коэффициентов недоиспользования пропускных способностей ветвей связи базовой СОИ

$$D = \max_{j\in J}(F_{D\,i}(G,\ \Lambda, \Lambda^{ПрОА}, U)) \tag{3.20}$$

при ограничениях

$$\left.\begin{array}{l} P_{ПИ\,ij}^{(k)} \le P_{ПИ}^{доп} \\ P_{СД\,ij}^{(k)} \ge P_{СД}^{TP} \end{array}\right\}, \tag{3.21}$$

где $F_{D\,i}$ – функция, определяемая способом перераспределения нагрузки в СОИ.

Элемент j_{ij}^{+} есть элемент матрицы $J_{[N\times N]}^{+}$ и представляет собой номер зоны ИП противнику в ответ на его атаку из зоны узла i БСОИ по отношению к объекту атаки, находящемуся в зоне узла j БСОИ.

3.5.2. Решение задачи перераспределения не целевой нагрузки в сети обмена информацией. Выбор зоны интеллектуального противодействия.

Управление процессом обмена информацией в сети может быть разделено на управление интенсивностью передаваемых по сети информационных потоков и распределением этих потоков по сети.

Исходя из этого, предлагается задачу выбора зон ИП на основе перераспределения потоков $x_{ij}^{ПрОА}$ свести к двухшаговой оптимизации по следующим переменным:

λ_{ij} - интенсивность потока информационных пакетов;

j – номер узла базовой сети, из зоны которого может быть реализована стратегия ИП выбранного типа *l*.

По сути, на первом шаге решается задача управления интенсивностью информационных потоков в СОИ x_{ij}, а на втором – распределением потоков $x_{ij}^{ПрОА}$ по сети.

На первом шаге решается традиционная задача, состоящая в нахождении на сети, заданной графом $G_k(N, N_1)$ для каждого приоритетного потока максимальных входящих потоков $\Lambda_{N\times N}^{(k)+} = \left\|\lambda_{ij}^{(k)+}\right\|$, при которых еще выполняются условия:

$$\left.\begin{array}{l} P_{ПИ\,ij}^{(k)} \le P_{ПИ}^{доп} \\ P_{СД\,ij}^{(k)} \le P_{СД}^{доп} \end{array}\right\} \tag{3.22}$$

то есть, таких потоков $\lambda_{ij}^{(k)+}$, $i, j = \overline{1, N}$ увеличение хотя бы одного из которых $\lambda_{ij}^{(k)+}$ приводит к невыполнению условия (3.22).

В дальнейшем индекс (k) будем опускать, подразумевая его.

В виду того, что адаптивная маршрутизация в условиях насыщения сети приводит к равномерному использованию ресурсов сети, то потоки по сети необходимо распределять так, чтобы максимизировалась сумма коэффициентов недоиспользования пропускных способностей ветвей связи.

В этом случае математическая постановка задачи поиска Λ^{+} сводится к следующему.

Необходимо найти

$$\Lambda^{+} \rightarrow \max_{\lambda_{ij} \in \Omega} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(1 - \frac{a_{ij} + b_{ij} + d_{ij} + \lambda_{ij}}{(1 - P_B) K_{ij} \mu_{ij}}\right) \quad (3.23)$$

при ограничениях (3.22) на область Ω.

То есть для оптимизации информационных потоков необходимо обеспечить максимизацию функции D_1.

Функция $D_1 = \sum_{i=1}^{N} \sum_{j=1}^{N} \left(1 - \frac{a_{ij} + b_{ij} + d_{ij} + \lambda_{ij}}{(1 - P_B) K_{ij} \mu_{ij}}\right)$ является стоимостной функцией и используется для количественного описания эффективности сети [87].

В результате первого шага оптимизации определяется матрица $\Lambda^{+}_{[N \times N]}$ ограничений интенсивностей входящих потоков информационных пакетов в БСОИ без учета не целевых потоков (потоков типа Пр-ОА).

Далее необходимо определить резерв входящих потоков по интенсивности $\Delta\Lambda$, как разность матрицы максимально возможных интенсивностей входящих потоков информационных пакетов Λ^{+} и матрицей интенсивностей входящих потоков информационных пакетов в Λ в дискретный момент времени $k \in \tau$.

Таким образом, резерв по интенсивности определяется следующим образом

$$\Delta\Lambda = \Lambda^{+} - \Lambda. \quad (3.24)$$

Применение для решения задачи (3.23) методов нелинейного программирования приводит к чрезмерной сложности.

Поэтому для решения задачи (3.23) при условиях (3.22) используем следующий итеративный алгоритм нахождения субоптимального решения [87].

Производится ранжирование потоков по критерию максимума интенсивности

$$x_{ij} \rightarrow \max_{i,j} \lambda_{ij}.$$

Выбирается первый из ранжированных потоков из матрицы Λ .

Для выбранного потока с помощью алгоритма Флойда-Уоршелла (или любого другого подобного алгоритма) находится путь с минимумом транзитных участков (ij).

Для данного пути (ij) определяются вероятности $P_{Сдij}$ и $P_{Пиij}$ и проверяются условия (3.21). Если условия (3.21) не выполняются, то данный путь исключается из рассмотрения и осуществляется переход к пункту 3. при выполнении условия (3.21) из сети исключается часть пропускной способности, задействованной под передачу потока и вычисляется значение целевой функции $D_1(1)$ в выражении (3.20).

Задействованный под передачу потока путь исключается из рассмотрения и процедура повторяется с пункта 3.

Вычисленное в пункте 4 новое значение целевой функции $D_1(2)$ сравнивается со старым. Если $D_1(2) > D_1(1)$, то под установление виртуального соединения выделяется последний рассмотренный путь и $D_1(1)$ присваивается значение $D_1(2)$.

Процедура повторяется с пункта 3 до тех пор пока не будут просмотрены все возможные пути между узлами i и j.

Из ранжированного ряда потоков выбирается следующий поток и осуществляется снова переход к пункту 3 и так до тех пор пока не будет просмотрен весь ряд потоков.

Если все потоки обслужены, то каждый элемент матрицы Λ увеличивается на величину ε и осуществляется переход к пункту 2. Если же имеется хотя бы один необслуженный поток, то каждый элемент матрицы Λ увеличивается на величину ε и осуществляется переход к пункту 2.

Процедура выполняется до тех пор, пока с точностью до ε не будет удовлетворено условие:

$$\left.\begin{array}{l} P_{ПИ\,ij}^{(k)} \le P_{ПИ}^{доп} \\ P_{СД\,ij}^{(k)} \le P_{СД}^{доп} \end{array}\right\},$$

i, j = $\overline{1, N}$

Номер приоритета увеличивается на единицу и осуществляется переход к пункту 2.

Когда все потоки обслужены, осуществляется присвоение

$$\Lambda^{+} = \Lambda + \left\| m\varepsilon \right\|_{N \times N}, \; r = \overline{1, R}$$

Величина m является числом циклов изменения матрицы Λ на величину ε.

Блок-схема алгоритма представлена на рис. 3.14.

В силу специфики решения задачи, мы не рассматривали приоритетные потоки, так как основная задача метода это перераспределение потоков $x_{ij}^{ПрОА}$, когда уже определен резерв по интенсивности. Но, несмотря на это, описанный алгоритм может быть легко расширен путем рассмотрения приоритетных потоков, всего лишь за счет того, что описанная в алгоритме процедура последовательно применяется ко всем приоритетным потокам.

Таким образом, в ходе выполнения алгоритма будут найдены все пороговые значения интенсивностей входящих потоков Λ^{+}, а также суммарные потоки на входе каждой ветви.

Так как условия (3.22) заданы в неявном виде, то необходимо определить $P_{СДij}$ и $P_{ПИij}$.

Воспользовавшись результатами работы [68], получим:

$$P_{СДij} = \prod_{k=1}^{S}\left(\frac{\mu_k - \alpha_k\lambda_{ij}}{\mu_k - \alpha_k\lambda_{ij} + \nu}\right) \bullet \prod_{l=1}^{S+1}\left(\frac{\mu_{Цl} - \Lambda_l m_l^{-1}}{\mu_{Цl} - \Lambda_l m_l^{-1} + \nu}\right),$$

где

$\mu_k = t_{пk}^{-1}$, $t_{пk}$ – время передачи сообщения по ветви k;

$\mu_{Цl} = t_{обр\,l}^{-1}$, $t_{обр\,l}$ – время обработки сообщения в центре коммутации пакетов *l*;

S – число транзитных участков в пути;

λ_{ij} – интенсивность пакетов рассматриваемого сообщения в пути i,j;

Λ_l – общая интенсивность пакетов в ***l***-й ветви;

α_k – число логических каналов проключенных в одном физическом канале ветви k;

m_l – число вычислительных модулей на центре коммутации пакетов ***l***, обрабатывающих пакеты.

Вероятность потери пакетов в пути (i, j), состоящим из S транзитных участков, определяется по формуле:

$$P_{ПИ} = 1 - \prod_{k=1}^{S}(1 - P_{ПИ\,k}^{КС}) \bullet \prod_{l=1}^{S+1}(1 - P_{ПИ\,l}^{ЦКП}) \bullet P_{СД\,ij}$$

где $P_{ПИ\,k}^{КС}$ – вероятность потери пакетов на уровне канала связи; $P_{ПИ}^{КС} = 1 - P_{ОИ}^{КС} \bullet K_{Г}$;

$P_{ОИ}^{КС}$ – вероятность обнаружения искажения в канале связи.

В силу независимости событий обнаружения искажений информационного кадра в канале связи:

$P_{ОИ}^{КС} = P_{И}^{КС} - P_{ПИ}^{КС}$ **;**

где $Р_И^{КС} = 1-(1-Р_{ОШ})^l$, $Р_{ОШ}$ – вероятность побитовой ошибки в канале связи; ***l*** – средняя длина кадра,

$\mathbf{Р_{ПИ}^{КС} \approx Р_И^{КС}\, 2^{-}} - \eta$

η - число проверочных бит в информационном кадре.

В завершении первого этапа определяется значение матрицы $\Delta\Lambda$ по формуле ( ).

Второй шаг оптимизации оптимизации посвящен решению задачи распределения потоков пакетов $x_{ij}^{ПрОА}$ по сети.

Построим матрицу $U_{NxN.}$ Элемент матрицы U_{NxN} есть вектор $u_{ij} = < u_{ij}^{1}, ... u_{ij}^{l}, ... u_{ij}^{L} >$, где u_{ij}^{l} - индикатор стратегии ИП типа *l*, которая может быть

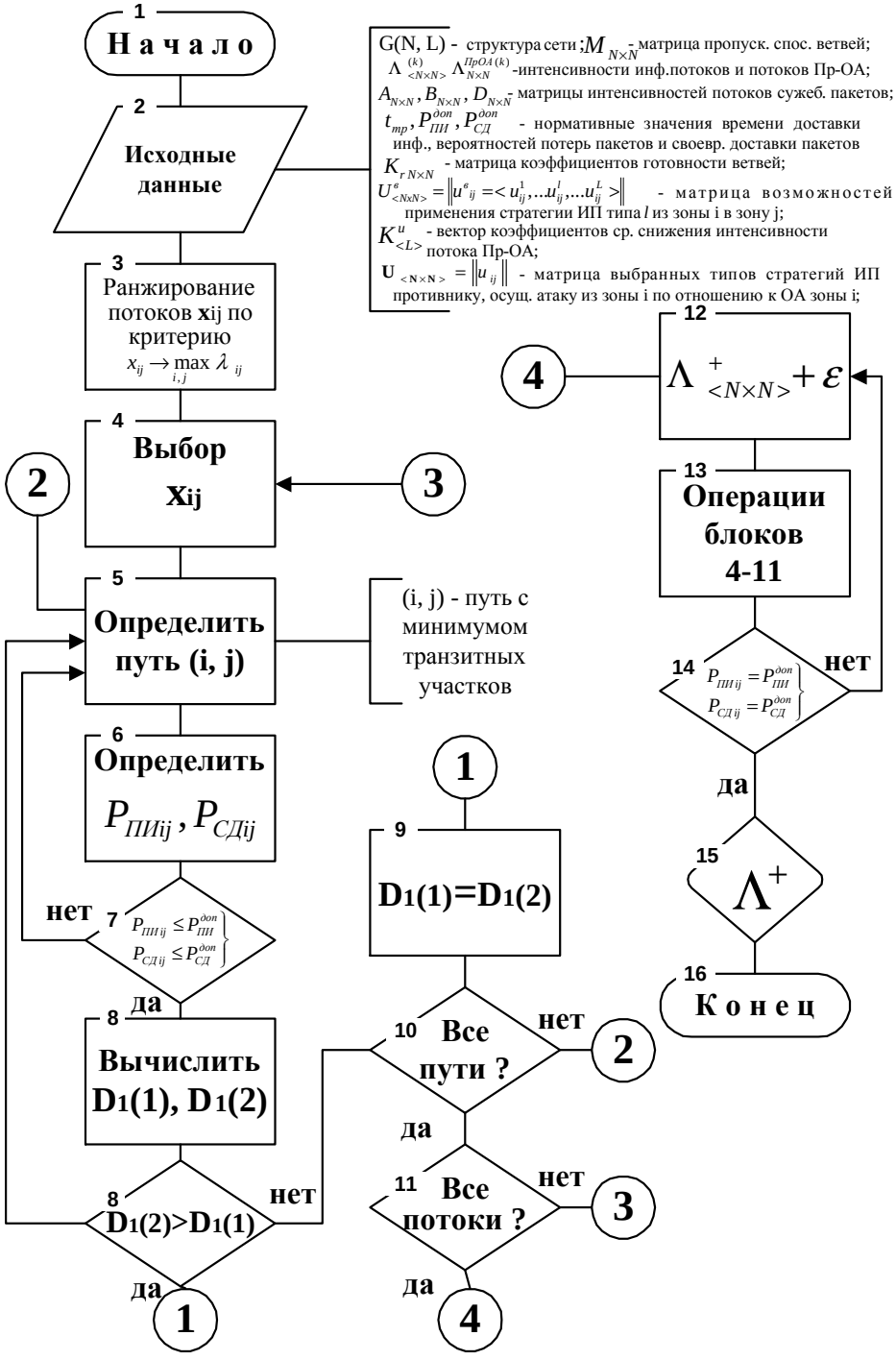


Рис. 3.14 Блок-схема алгоритма выбора зон интеллектуального противодейст-вия.

применена из зоны узла базовой сети в отношении противника, находящегося в зоне узла i базовой сети (определяется наличием ПО и ТС).

$$u_{ij}^{l} = \begin{cases} 0, \textit{ если стратегия может быть реализована}; \\ 1, \textit{ в противном случае.} \end{cases}$$

$l \in L$ - номер типа стратегии ИП.

n_L – число типов стратегий ИП, для реализации которой в СОИ имеются необходимые технические и программные средства.

Подчеркнем, что с течением времени $L^{(k)}$ может быть равно или не равно $L^{(k-1)}$.

Введем вектор $K_u =< K_{u^1}, ...K_{u^l}, ...K_{u^L} >$, где K_{u^l} - коэффициент среднего снижения интенсивности $\lambda_{ij}^{ПрОА}$ при использовании стратегии ИП типа l.

Коэффициенты K_{u^l} определяются на основе экспертных оценок с учетом анализа соответствующих информационных потоков и уточняются при помощи использования средств искусственного интеллекта.

На первом шаге мы определили матрицу ${}_{\Delta}\Lambda$, элементы ${}_{\Delta}\lambda_{ij}$ которой формируются по следующему правилу:

$${}_{\Delta}\lambda_{ij} = \begin{cases} \lambda_{ij}^{+} - \lambda_{ij}, \textit{ если } \lambda_{ij} \le \lambda_{ij}^{+} \\ 0, \textit{ если } \lambda_{ij} > \lambda_{ij}^{+} \end{cases}$$

Элемент ${}_{\Delta}\lambda_{ij}$ представляет собой резерв интенсивности потока пакетов от узла i к узлу j БСОИ.

Так как ${}_{\Delta}\Lambda$ определена с учетом максимизации основного показателя D_1, то на втором шаге оптимизации в качестве разделяемого ресурса БСОИ можно рассматривать резерв интенсивности потоков, заданный матрицей ${}_{\Delta}\Lambda$.

При этом задача распределения потоков $x_{ij}^{ПрОА}$ с интенсивностями $\lambda_{ij}^{ПрОА}$ может быть поставлена как задача, состоящая в нахождении на сети, заданной графом $G_k(N, N_1)$, таких точек приложения ИП j_{ij}^{+} (зон ИП), при которых максимизируется сумма коэффициентов недоиспользования резерва интенсивностей потоков в ветвях.

Элемент j_{ij}^{+} есть элемент матрицы J^{+}, представляющий собой номер зоны СОИ, куда перенаправляется поток $x_{ij}^{ПрОА}$ (то есть номер зоны ИП). Нулевому потоку $x_{ij}^{ПрОА}$ ($\lambda_{ij}^{ПрОА} = 0$) соответствует $j_{ij}^{+} = 0$. При этом, номера узлов БС пронумерованы от 1 до N включительно.

В этом случае математическая постановка поиска матрицы J^{+} сводится к следующему.

$$J^{+} \to \max_{j^{+} \in J} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(1 - \frac{\lambda_{ij}^{ПрОА} \bullet u_{ij^{+}}^{l} \bullet K_{u^l}}{{}_{\Delta}\lambda_{ij^{+}}} \right), \qquad (3.25)$$

где $D_2 = \sum_{i=1}^{N} \sum_{j=1}^{N} \left(1 - \frac{\lambda_{ij}^{ПрОА} \bullet u_{ij^{+}}^{l} \bullet K_{u^l}}{{}_{\Delta}\lambda_{ij^{+}}} \right)$.

Примечание. На множестве L типов стратегий ИП задано отношение строгого порядка $r_{\succ} =< L, R_{\succ} >$, таким образом, L – представляет собой строго упорядоченное множество по важности типа реализуемого ИП. Предположим, что множество L упорядочено таким образом, что с возрастанием номера типа стратегии растет и ее важность (приоритет) с точки зрения первоочередной реализации.

Таким образом, при всех равных условиях стратегия типа i должна быть реализована в первую очередь по отношению к стратегиям i+k, где $k=\overline{1, L-i}$.

Для решения сформулированной задачи (3.25) предлагается следующий алгоритм нахождения оптимального решения J^{+}.

Фиксируются потоки Пр-ОА в отношении которых предполагается реализация стратегии ИП с наивысшим приоритетом.

Производится ранжирование потоков выбранного приоритета по критерию максимума интенсивности.

Выбирается первый из ранжированных потоков из матрицы $\Lambda^{ПрОА}$ (попросту, выбирается поток $x_{ij}^{ПрОА}$ с наибольшей интенсивностью $\lambda_{ij}^{ПрОА}$).

Для выбранного потока $x_{ij}^{ПрОА}$ принимается j=1.

Для выбранного потока вычисляется значение целевой функции $D_2(1)$ в выражении (3.25).

Значение j увеличивается на 1. (j=j+1) и процедура повторяется с пункта 5.

Вычисленное в пункте 6 новое значение целевой функции $D_2(2)$ сравнивается со старым. Если $D_2(2)>D_2(1)$, то в качестве зоны ИП рассматривается зона СОИ с номером соответствующим последнему значению j и $D_2(1)$ присваивается значение $D_2(2)$.

Процедура повторяется с пункта 5 до тех пор, пока j не достигнет значения N (N – число узлов БСОИ, количество зон СОИ).

Из ранжированного ряда потоков выбирается следующий поток и осуществляется переход к пункту 4.

Номер приоритета типа ИП увеличивается на единицу и осуществляется переход к пункту 2.

В ходе выполнения алгоритма будет найдена матрица J^+.

Блок-схема алгоритма приведена на рис. 3.15.

Таким образом, результате второго шага оптимизации получаем матрицу J^+, содержащую номера j^+_{ij} зон ИП противнику в ответ на его несанкционированные действия из зоны узла i БСОИ по отношению к объекту атаки, находящемуся в зоне узла j БСОИ, то есть новые конечные точки потоков $x_{ij}^{ПрОА} \rightarrow x_{ij^+_{ij}}^{ПрОА}$.

Очевидно, что определение зон ИП, путем перераспределения в СОИ потоков

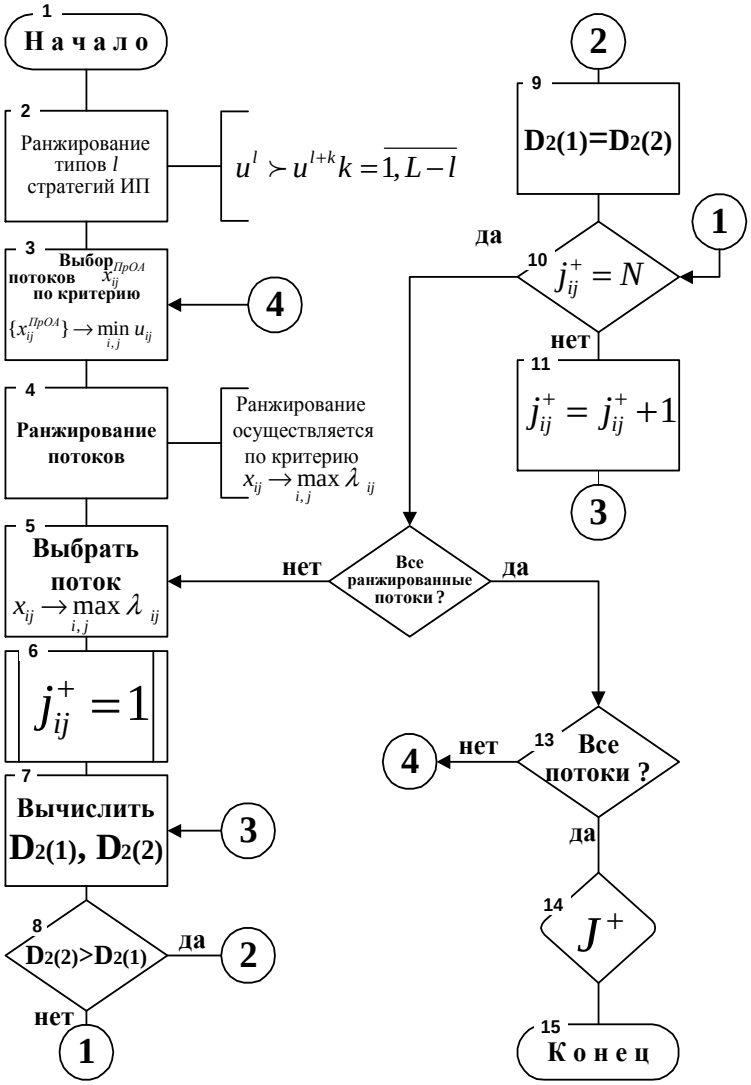


Рис. 3.15. Блок-схема алгоритма нахождения зон противодействия

$x_{ij}^{ПрОА}$ наиболее эффективно при многочисленных и массированных атаках на СОИ по разным направлениям.

Использование предложенного метода выбора зон ИП позволит локализовать ИБ по зонам СОИ и освободит ресурсы БСОИ, необходимые для передачи информационных потоков в целях управления нашей корпорацией.

3.5.3. Анализ результатов решения задачи перераспределения нагрузки в сети обмена информацией

Для отработки решения задачи перераспределения нагрузки в СОИ, связанной с действиями противника был проведен ряд экспериментов на программной модели базовой СОИ.

Эксперименты по моделированию проводились на IBM-совместимой ПЭВМ (CPU Inteal Pentium 166 MMX).

В ходе экспериментов исследовалась зависимость суммы коэффициентов недоиспользования пропускной способности (СКНПС) от суммарной интенсивности входящих потоков, распределенной по сети случайным образом.

При этом сравнивались значения СКНПС D^1, полученные по предложенному в параграфе 3.5. методу, с аналогичными значениями СКНПС D^2, полученными при использовании только традиционного метода ограничения нагрузки в СОИ [87].

Другое направление экспериментальных исследований – это зависимость СКНПС от увеличения структуры сети (увеличению числа узлов N_1в СОИ).

Сравнительный анализ произведен на сети, структура которой показана на рис. 3.16. В кружках проставлены номера узлов, а на ребрах – пропускная способность линий связи μ_{ij}.

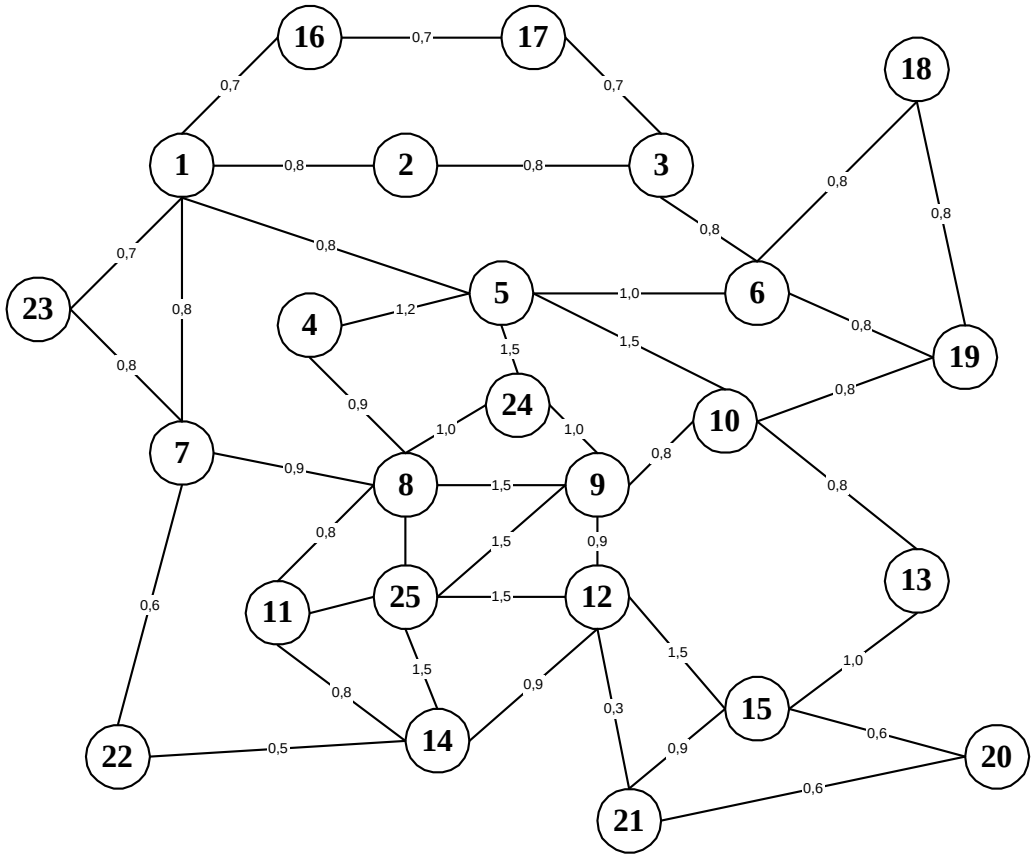


Рис. 3.16. Экспериментальная сеть обмена информацией

Так как значения $\hat{D}^1(\hat{D}^2)$ представляют собой случайные величины, то каждое конкретное значение D^1 (D^2) вычислялось как математическое ожидание M^* случайной величины $\hat{D}$

$$M^*_{\hat{D}} = \frac{\sum_{i-1}^{n} D_i}{n},$$

где n – количество испытыний (опытов) в эксперименте (в данных экспериментах n=300).

Полученные зависимости приведены на графиках (рис. 3.17). На рисунке представлена зависимость D^1 (D^2) от интенсивности потоков типа Противник-Объект атаки для обоих сравниваемых методов.

На рисунке представлена зависимость нормированных значений D^1/MS (D^2/MS) от интенсивности потоков типа Противник-Объект атаки для обоих сравниваемых методов, где MS – максимально возможное значение СКНПС при структуре сети $G(N_1,N_2)$. Значение MS – численно равно сумме элементов матрицы смежности.

Рост выигрыша значения D^2 по отношению к D^1 с ростом интенсивности атак противника приведен на рис.

Из графика (рис. 3.18) виден выигрыш значения D^2 по отношению к D^1 с увеличением структуры сети (с ростом N_1). Кривые 1, 2, 3 соответствуют разным значениям интенсивностей потоков в сети. При построении каждой кривой (при увеличении N_1) суммарная интен-

сивность потоков увеличивалась пропорционально увеличению N_1, а направления потоков выбирались случайным образом.

Из графиков (рис. 3.19., 3.20) видно, что во всех случаях предложенный алгоритм перераспределения нагрузки намного лучше традиционного алгоритма ограничения нагрузки [87, 9], применяемого самостоятельно, и с увеличением интенсивности информационных атак противника в СОИ, а также с увеличением структуры сети этот выигрыш увеличивается.

Таким образом, предложенный алгоритм перераспределения не целевой нагрузки в СОИ, используемый в предложенном методе для определения зон интеллектуального противодействия является высокоэффективным алгоритмом.

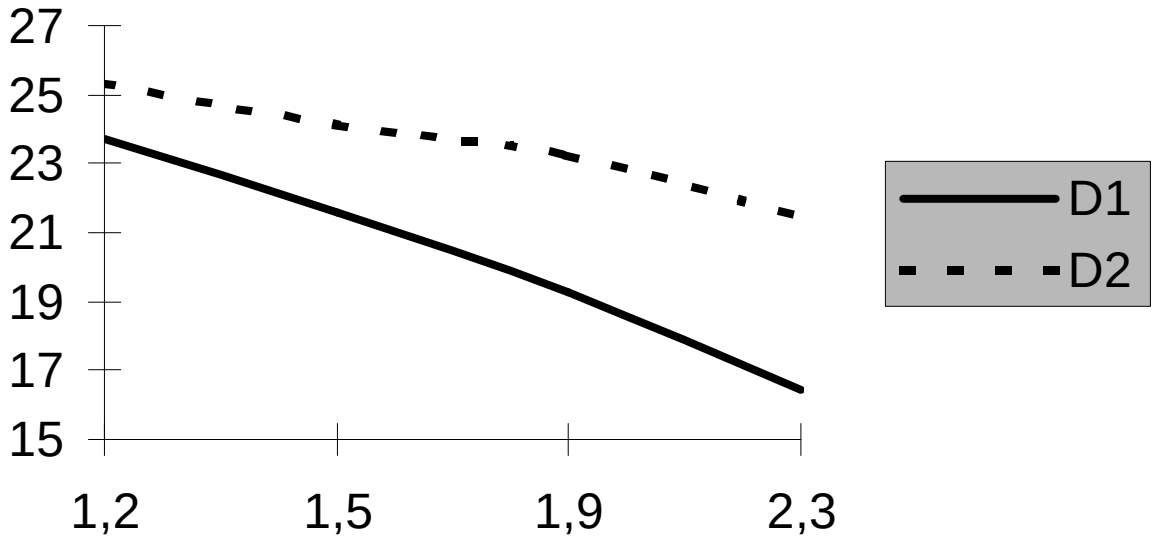


Рис. 3.17. Зависимость СКНПС от интенсивности потоков типа Противник-Объект атаки для двух сравниваемых алгоритмов

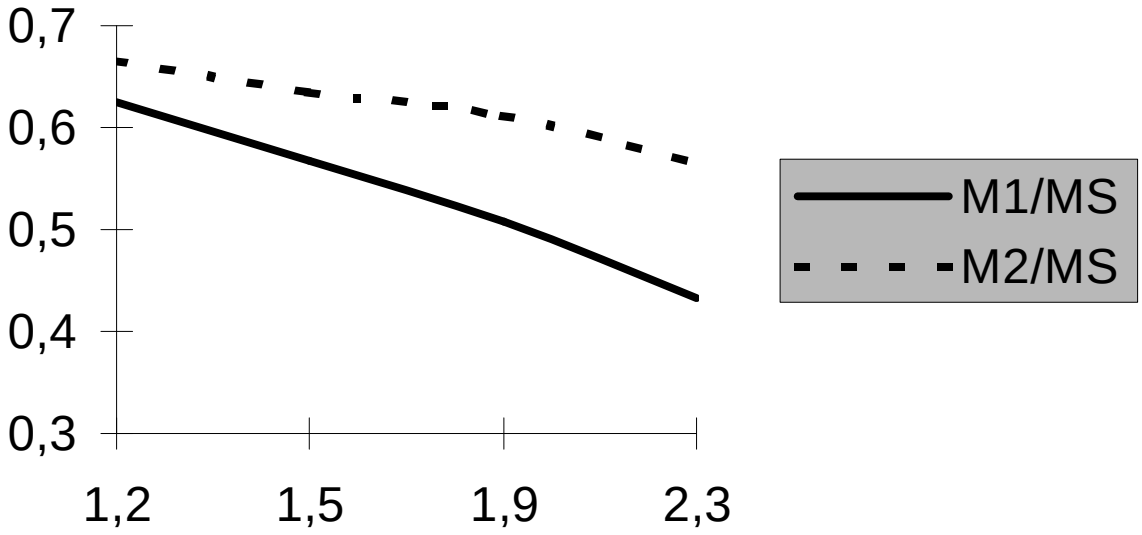


Рис. 3.18. Зависимость нормированных значений СКНПС от интенсивности потоков типа Противник-Объект атаки для двух сравниваемых алгоритмов

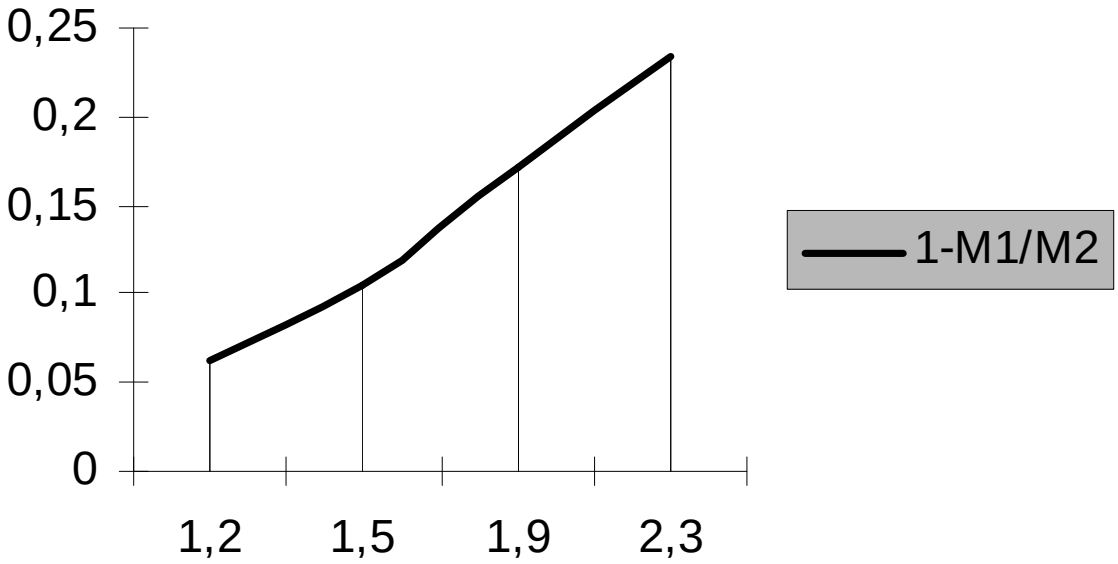


Рис. 3.19. Зависимость выигрыша значения D^2 по отношению к D^1 с ростом интенсивности атак противника

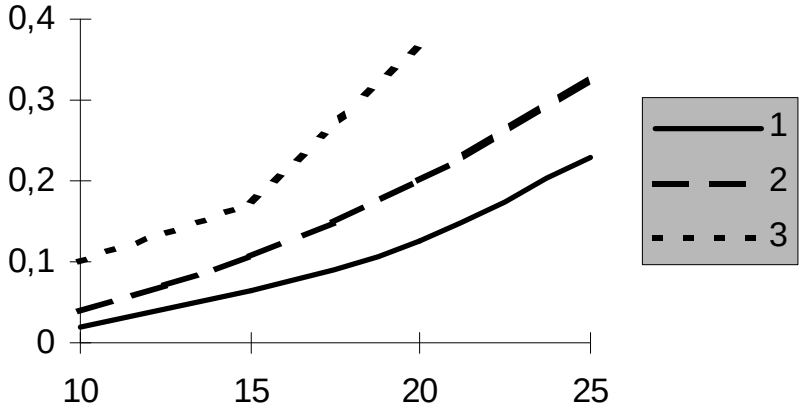


Рис. 3.20. Зависимость выигрыша значения D^2 по отношению к D^1 от увеличения структуры СОИ (от величины N_1)

3.6. Метод построения программы интеллектуального противодействия в корпоративной сети обмена информацией

Согласно принятому в первой главе подходу к осуществлению интеллектуального противодействия в корпоративной СОИ, данный вид противодействия реализуется на основе использования ложного объекта атаки, имитирующего результаты или процесс функционирования объекта или элемента СОИ, выбранного противником в качестве объекта атаки.

Ранее также было определено, что совокупность ЛОА и модуля управления ЛОА со стороны субъекта информационной борьбы (администратора сети) представляет собой программу интеллектуального противодействия (ПИП), которая является инструментарием субъекта интеллектуальной борьбы в ходе осуществления ИП.

Программа интеллектуального противодействия работает согласно выбранного или построенного в оперативном режиме протокола интеллектуального противодействия.

Протокол интеллектуального противодействия определяется, исходя из выбранного, например, на основе описанного в настоящей работе метода, вида противодействия l и спецификации протокола ИП R_C.

Затем формируется ПИП, либо за счет имеющегося программного обеспечения, либо за счет разработки нового программного обеспечения.

В виду того, что стратегия противника имеет определенный период жизни, ему может соответствовать и жизненный цикл ПИП (но это совсем не обязательно).

Очевидно, что необходим метод, позволяющий строить ПИП в короткое время и обеспечивающий корректность ее работы.

При этом следует отметить, что в виду нарастающей тенденции к резкому увеличению арсенала информационного оружия, соответствующих вариаций возможных информационных атак в СОИ, необходимости участия субъектов информационной борьбы (администратор сети) в процессе ИП, нецелесообразно идти по пути создания специализированных протоколов управления для некоторых видов ИП. Появляется настоятельная необходимость использования унифицированной для всех видов ИП концепции управления, независимой от структуры конкретной реализации ИП.

В этой связи примечательно, что описанная проблема управления ИП во многом схожа с проблемой управления разными видами телеслужб в СОИ.

Общность проблем управления телеслужбами и ИП состоит в том, что в первом случае необходимо предоставить пользователю набор служб $S=\{S_i\}$, каждая из которых удовлетворяет вполне определенной информационной потребности абонента. Во втором случае требуется предоставить субъекту ИБ набор служб (ПИП) $S=\{S_j\}$, каждая из которых удовлетворяет его потребности по осуществлению вполне определенной стратегии ИП. То есть наблюдается некоторое соответствие: субъект ИБ – пользователь; служба или ее аспект (услуга) – программа ИП.

Так как в отличие от методов управления ИП, проблемы управления телеслужбами (в частности построения интегрированных услуг в ЦСИО) в достаточной степени изучены [48], то для построения программ ИП можно воспользоваться некоторыми результатами, полученными при решении задач управления телеслужбами [48, 9] в СОИ.

В этой связи интересна архитектурная концепция создания и предоставления услуг связи в СОИ, получившая название Интеллектуальная сеть (ИнС) [48]. С этой концепцией в настоящее время связано довольно много работ [48, 9, 49, 70], а по тематике интеллектуальной сети в последние годы было организовано несколько международных конференций.

Не вдаваясь глубоко в концептуальные аспекты интеллектуальной сети, отметим, что основная цель создания ИнС заключается в упрощении процессов введения новых услуг в СОИ.

При этом можно выделить два неоспоримых преимущества ИнС [48]. Пользователь получает определенную возможность управлять услугами, которые представляет сеть, формировать и заказывать на определенное время необходимые ему новые услуги. Распределенная по сети база данных обеспечивает доступ пользователя к необходимым услугам независимо от их географического положения в данный момент времени.

Важной особенностью ИнС является то, что в ней услуги разделяют на две группы: основные услуги (ОУ) и дополнительные услуги (ДУ), и для построения ДУ используют не зависящие от видов услуг и друг от друга функциональные компоненты (ФК). Это обеспечивает гибкость, оперативность построения новых ДУ и активное участие самого пользователя ИнС. Кроме того, используя повторно ФК при построении разных ДУ, исключается необходимость проектирования повторяющихся компонент протоколов ДУ.

Поэтому в данной работе по аналогии с ИнС выбран, а далее обосновывается подход к построению ПИП на основе ФК, которые выступают в роли своеобразных «кирпичиков», из которых и строится ПИП.

Построение ФК может осуществляться высококвалифицированными специалистами с учетом знаний и опыта ведущих экспертов в области обеспечения информационной безопасности и информационного противодействия в СОИ. Примечательно, что в дальнейшем субъ-

екты ИБ могут использовать эти ФК для построения своей программы ИП, не вникая глубоко в логику функционирования отдельных ФК. Для этого наиболее перспективно использовать удобную интегрированную среду с графическим интерфейсом.

В связи с тем, что сети Петри обладают широкими возможностями в описании структуры и поведения сложных систем, в данной работе они выбраны в качестве аппарата описания и анализа протокольных блоков (функциональных компонент) и протоколов ИП, в соответствии с которыми работают программы ИП.

3.6.1. Постановка задачи

Пусть $M = \{M_i\}$ - множество спецификаций протоколов ИП.
Построение специального математического обеспечения ИП (программы ИП) можно разбить на несколько этапов:

проектирование протокола ИП на основе спецификации;

анализ протокола ИП на логическую корректность;

построение программного обеспечения, реализующего ИП (ПИП);

предоставление ПИП в распоряжение субъекта ИБ.

Так как первые два этапа являются итеративными, то их можно объединить в один этап – проектирование, время же этого этапа будем обозначать $\mathbf{T_{ПРОЕКТ}}$.
Следовательно, время построения ПИП и предоставления ее субъекту ИБ $\mathbf{T_{ПИП}}$ определяется как

$$\mathbf{T_{ПИП}=T_{ПРОЕКТ.}+T_{ПОСТРОЕНИЯ};\ T_{ПРОЕКТ.}>> T_{ПОСТРОЕНИЯ.}},$$

где время $\mathbf{T_{ПРОЕКТ}}$ представляет собой длительность этапа проектирования, а $\mathbf{T_{ПОСТРОЕНИЯ}}$ – длительность этапа построения ПО.

При этом, время проектирования ПИП определяется, главным образом, методом проектирования

$$\mathbf{T_{ПРОЕКТ.}=F(M_c)},$$

где $F(M_c)$- функция, определяемая способом проектирования услуги.

Время построения ПИП $\mathbf{T_{ПИП}}$ является самым критичным показателем успешного ведения ИБ, так как большая длительность построения ПИП обладает значительными демаскирующими свойствами и не позволяет осуществить ИП противнику скрытно, а значит, целевой результат ИП будет отрицательным.

Таким образом, для достижения высокой оперативности осуществления ИП в СОИ должна быть решена задача сокращения времени построения ПИП $\mathbf{T_{ПИП}}$. При решении данной задачи необходимо решить ряд частных задач, которые заключаются в разработке:

- модели ИП как динамической системы;
- метода корректного построения ПИП из функциональных компонент.

Существующие методы проверки любого протокола на корректность сводятся к построению множества допустимых состояний [45].
Эта задача принадлежит к классу задач экспоненциальной сложности [45].

$$\mathbf{O(F(M_c)) \subset O(exp(n))},$$

где $P = \{p_i\}$ - множество состояний протокола ИП; $i=\overline{1,n}$; n- количество состояний протокола.

Поэтому эти методы неприменимы для анализа протоколов ИП, которые могут иметь значительное число состояний. Для уменьшения сложности задачи анализа протоколов ИП на корректность целесообразно их рассматривать, как состоящие из множества функциональных компонент (модулей) - подпротоколов, фаз, процедур.

В дальнейшем наравне с термином «функциональная компонента» будем использовать термин «модуль протокола».

Пусть $M_c=\{m_i\}$, где m_i - модуль протокола ИП, $i=\overline{1,n}$; n-количество модулей протокола ИП.

s_i -мощность подмножества состояний протокола ИП, реализуемого в модуле m_i.

Тогда сложность задачи анализа протокола ИП на логическую корректность определяется как

$$O(F_k\ (M_c)) \subset O(\exp(\max_{i\in[1,n]}(s_i))),$$

а время проектирования ПИП

$$T_{ПРОЕКТ.} = F_k\ (M_C) = k \bullet \left(\sum \exp(s_i) \right),$$

где k - индивидуальный коэффициент разработчика [32].

Учитывая то, что ПИП представляет собой композицию n функциональных компонент, а число разработчиков l>1, тогда имеем $K_{<l>} =< k_1 ,...k_j ,...k_l >$ - вектор индивидуальных коэффициентов разработчиков и время проектирования ПИП

$$T_{ПРОЕКТ.} = \max_{j\in[1,l]} k_j \bullet \left(\sum_{i=1}^{n_j} \exp(s_i) \right),$$

где n_j – количество функциональных компонент ИП, разрабатываемых *j*-ым разработчиком ($j = \overline{1,l}$).

Кроме того, за счет повторного использования протокольных модулей достигается существенное сокращение времени построения программного обеспечения ИП :

$$T_{ПОСТР.} = \sum_{i=m}^{n} T_i + T_{СОСТ.},$$

где m- количество построенных модулей;

T_i - время, необходимое на разработку i-го модуля;

$T_{СОСТ.}$ - время, необходимое на составление ИП из модулей.

$$T_{сост.} < T_i .$$

Необходимо разработать метод построения ПИП в корпоративной СОИ, отвечающий следующему критерию оптимальности

$$T_{ПИП.} = \min_{i\in I}(F_i\ (M_c))$$

при ограничениях

$S(M_C)$ – *язык, пророждаемый протоколом ПИП* M_C

WF – *множество допустимых слов протокола ПИП*;

$\forall L \in S(M_C), L \subset WF$ – *свойство корректности протокола ИП*,

где L – *слово из языка* $S(M_C)$

Функционально построение ПИП возлагает на соответствующую распределенную подсистему СЗИ СОИ.

3.6.2. Решение задачи построения программы интеллектуального противодействия из функциональных компонент

Разрабатываемые ПИП должны удовлетворять следующим требованиям:

- максимальная адаптируемость к субъекту ИБ;
- возможность комбинирования с другими ПИП;
- простота в использовании.

Представим СОИ как множество сетевых ресурсов и выделим из него множество сетевых ресурсов (далее просто ресурсов), которые могут быть задействованы для ИП Q={q_1, …, q_n}. Каждый ресурс будем характеризовать именем $q_i \in Q$ и алфавитом A_i. Парой (a, q_i) обозначим состояние ресурса q_i, тогда множество состояний всех ресурсов S={(a,q)}, $a \in A_i$ –

глобальное состояние системы в определенный момент времени, а $S' \subseteq S$ – локальное состояние.

Преобразуя данные, ресурс изменяет свое состояние, состояние других связанных с ним ресурсов, то есть производит их локальное преобразование, которое будем называть операцией.

Каждую ФК представим как микропрограмму ИП.

Определение. *Микропрограммой (программой) ИП* называется множество операций, объединенных по какому-либо признаку в определенном взаимодействии

$$\Phi_{\{n\}}=\{O_j\},$$

где O_j : $S`_j \rightarrow S"_j$,

($S`_j=\{(a,q`_i)\}$: $q`_i \in Q`_i$, $Q`_i \subseteq Q$) - локальное состояние, называется условием готовности операции O_j;

($S"_j=\{(b,q"_i)\}$: $q"_i \in Q"_i$, $Q"_i \subseteq Q$)- локальное состояние, называемое условием завершения операции O_j.

Операция O_j применима при глобальном состоянии S, если $S`_j \subseteq S$. В результате ее применения происходит смена состояний $(a,q`_i) \in S`_j$ на $(b,q"_i) \in S"_j$ для всех $q_i \in Q_i$.

Здесь Q_i - множество сетевых ресурсов, участвующих в операции O_j. Такая запись операций называется подстановкой [45].

Условия применимости и завершения, различных операций могут иметь непустые пересечения.

Так, если $S`_j \cap S"_k \neq \varnothing$, то состояние завершения операции O_k содержит пару (a,q), которая находится в условие применимости для O_j. Это значит, что между операциями O_k и O_j существует причинно-следственная связь, которая устанавливает порядок их выполнения. Таким образом, пересечения правых и левых частей разных подстановок определяют временную структуру ФК.

Подстановочная запись определяет также пространственную структуру программы ИП, т.е. физические связи между сетевыми элементами. Эти связи обусловлены тем, что участие в одной операции невозможно без взаимного обмена информацией. Например, если $(a,q`_i) \in S`_j$, $(b,q`_k) \in S"_j$, то из $q`_i$ в $q`_k$ должна существовать связь, так как $(a,q`_i)$ является условием для появления $(b,q`_k)$.

Программа ИП представляет собой композицию определенного количества ФК $\Phi=\{O_j\}$, которая осуществляется через интерфейсные состояния операций – точки доступа (ТД).

Для композиции ФК необходимо, чтобы условия или постусловия взаимодействующих операций пересекались, то есть необходимым условием композиции ФК является

$\Phi_1=\{O_{1i}\}$, где O_{1i} : $S`_{1i} \rightarrow S"_{1i}$,

$\Phi_2=\{O_{2j}\}$, где O_{2j} : $S`_{2j} \rightarrow S"_{2j}$,

$S`_{1i} \cap S"_{2j} \neq \varnothing \vee S`_{2j} \cap S"_{1i} \neq \varnothing$ -условие последовательной композиции;

$S`_{1i} \cap S`_{2j} \neq \varnothing \vee S`_{1j} \cap S`_{2i} \neq \varnothing$ -условие параллельной композиции.

Таким образом, логика ПИП, это структура $\Phi=\{\Phi_i\}$,где Φ_i - множество операций i-ой ФК.

При заполнении логики данными субъекта ИБ, то есть при задании конкретных значений из соответствующего алфавита множествам S` и S", получается возможная реализация этой программы ИП, состоящая из конкретных *процессов ИП*.

Определение. Под процессом ИП будем понимать последовательность вида $p=S_0, O_{i1}, S_{i1}, O_{i2}, \ldots$, где S_{ij} - глобальные состояния; O_{ij} - операции или последовательность операций, переводящие $S_{i\,j-1}$ в S_{ij}. Проекции процесса p на множества Ф и $\{S_i\}$ называются соответственно последовательностью операций $E_i= O_{i1}, O_{i2}, \ldots$ и последовательностью глобальных состояний $m_i= S_0,S_{i1},\ldots$. Процесс p в программе ИП $\Phi=\{O_i\}$ при исходном состоянии S_0 называется допустимым, если для каждого ij справедливо: O_{ij} применима при $S_{i\,j-1}$.

Определение. Множество всех допустимых при S_0 процессов ИП Ф называется динамикой ИП $П(Ф,S_0)$.

Определение. Глобальное состояние S , которое входит хотя бы в один допустимый процесс p=П(Ф, S), называется достижимым в динамике П(Ф,S).

Определение. Множество достижимых глобальных состояний называется множеством достижимости R(Ф,S) [45].

Динамика П(Ф,S) наглядно отображается графом ИП G(Ф,S). Граф ИП - ориентированный граф. Множество его вершин соответствует множеству достижимости R(Ф,S). Из вершины S_j в вершину S_k ведет дуга, если существует применимая при S_j операция О, результатом применения которой является S_k. Если из вершины графа не исходит ни одной дуги, то она называется тупиковой.

Построение ФК осуществляется с помощью объектно-ориентированного проектирования на раннем этапе построения системы интеллектуального противодействия.

При этом должно достигаться оптимальное сочетание следующих противоречивых требований к множеству ФК:

- покрытие всех функций ИП;
- минимизация количества ФК при минимально возможной избыточности реализованных в них функций.

Для формального описания ФК используется аппарат сетей Петри. При преобразовании результатов объектно-ориентированного проектирования в аппарат сетей Петри используются известные методы и алгоритмы, широко представленные в литературе [45, 60].

Если состоянию каждого ресурса сопоставить позицию сети Петри, а действию каждого ресурса по изменению состояния системы – переход сети Петри, то получим описание моделируемой ФК с помощью сети Петри.

В нашей работе мы не будем подробно останавливаться на теории сетей Петри, приведем только необходимые сведения.

3.6.3. Общие вопросы сетей Петри как формального аппарата описания и анализа протоколов

Определение. Сеть Петри (СП) – это набор

$$\mathbf{N=<P,T,B,F,\ M_0>,} \tag{3.26}$$

где

P={p_i} – конечное множество позиций, соответствующее множеству условий в системе;

T={t} – конечное множество переходов, соответствующее множеству событий; $B: P \times T \to N$ - функция входов;

$F: T \times P \to N$ - функция выходов, которые ставят в соответствие парам (p_i,t_j) и (t_i , p_j) целые неотрицательные числа b_{ij} или f_{ij} соответственно;

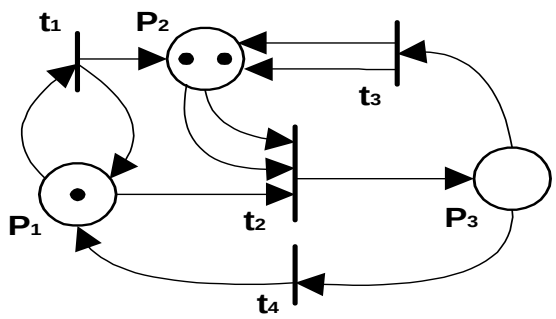


Рис. 3.16. Пример сети Петри

функция $M_0: P \to N$ - сопоставляет каждой позиции p∈ P некоторое число $M_0(p)$∈ N и называется начальной маркировкой.

Наиболее удобно представить СП в виде двудольного мультиграфа с множеством вершин P⋃T. Элемент из множества P изображается кружком, а элемент из множества T – чертой. Элементы разных множеств могут соединяться направленными дугами. Если из позиции p к переходу t ведет *k* дуг, то B(p, t)=*k*, и если из перехода t к позиции p ведет *l* дуг, то F(t, p) = *l*. Каждая позиция сети Петри характеризуется наличи-

ем определенного числа меток и определяется функцией $M : P \to N$, которая называется разметкой. Графически разметка представляется как распределение меток в позициях сети, изображаемых точками. Пример сети Петри приведен на рис. 3.16.

Матричное представление сети Петри - это две матрицы инциденций $B=[b_{ij}]_{mxn}$ и $F[f_{ij}]_{nxm}$, элементами которых служат значения функций B и F соответственно равны $b_{ij}=B(p_i,t_j)$, $f_{ij}= F(p_i,t_j)$. Строки матрицы B и столбцы матрицы F, соответствующие позиции p и определяющие инцидентные p переходы и кратности дуг к ним, обозначаются векторами $B_{(i,-)} = (b_{i1},...b_{in})$, $F_{(-,i)} = (f_{1i}, ...,f_{ni})$. Часто для обозначения инцидентных подмножеств используются "жирная точка" [66, 60], т.е. подмножество позиций, имеющих дуги к переходу t, обозначается $^\bullet t$, подмножество позиций, в которые ведут дуги из t, - $t^\bullet$. Аналогично обозначаются входные и выходные подможества переходов к позициям, т.е.

$$^\bullet t=\{p\in P: B(p,t)>0\};\quad t^\bullet=\{p\in P: F(t, p)>0\};$$
$$^\bullet p=\{t\in T: F(t, p)>0\};\quad p^\bullet=\{t\in T: B(p,t)>0\}.$$

Позиции $p\in {}^\bullet t$ называются входными к переходу t, позиции $p\in t^\bullet$ -выходными к переходу t. Аналогично переходы $t\in {}^\bullet p$ и $t\in p^\bullet$ называются входными и выходными к позиции p соответственно.

Если определена функция M_0, то СП называется маркированной и обозначается парой символов $<N,M_0>$. Маркировка сети представляется вектором $M= (M(p_1),..., M(p_m))$, причем если M(p)=k, то говорят, что в позиции p находится k маркеров. Маркировка, отображающая исходное состояние системы, называется начальной и обозначается M_0. В маркированной СП может происходить движение маркеров, которое подчиняется следующим правилам функционирования сети Петри.

1. Переход $t_j\in T$ считается возбужденным при маркировке M, если в каждой его входной позиции $p_i\in {}^\bullet t_j$ число маркеров равно или превышает кратность дуги b_{ij} или в векторном виде

$$\mathbf{M \geq B_{(-,j)}}. \qquad (3.27)$$

2. Возбужденный при маркировке M переход t_j срабатывает, изымая из каждой его входной позиции $p_i\in {}^\bullet t_j$ по b_{ij} маркеров и помещая в каждую его выходную позицию $p_k\in t_j{}^\bullet$ по f_{jk} маркеров. При этом маркировка M заменяется на M', что обозначается $M\ [t_j\rangle M'$, причем

$$\mathbf{M`=M - B_{(-,j)} + F_{(-,j)}}. \qquad (3.28)$$

3. Переход не может находится в состоянии возбуждения бесконечно долго; он должен либо срабатывать, либо возбуждение с него должно быть снято срабатыванием другого перехода. Этот закон называется "свойством конечной задержки".

4. Срабатывание перехода происходит мгновенно.

5. В один и тот же момент времени срабатывает только один переход.

Таким образом, если задана начальная разметка и хотя бы один переход оказывается возбужденным, то в сети Петри начинается движение меток.

Определение. Последовательность переходов $\sigma_i = t_{i1}, t_{i2}, ..., t_{ik}, ...$ такая, что $M_{i0}\ [t_{i1}>M_{i1}[t_{i2}>M_{i2} ... [t_{ik-1}>M_{ik}$ называется допустимой последовательностью срабатываний в сети $<N, M_0>$. Каждая подпоследовательность допустимой последовательности срабатываний допустима. Если σ_i переводит M_{i0} в M_{ik}, то это обозначается $M[\sigma_i>M$.

Определение. Маркировка, которая может быть получена в сети Петри $<N,M_0>$ в результате применения допустимой последовательности срабатываний, называется допустимой в сети $<N,M_0>$. Множество всех допустимых маркировок в сети $<N,M_0>$ называется множеством достижимости и обозначается $R<N,M_0>$.

Полная картина функционирования сети Петри $<N,M_0>$ наглядно представляется ориентированным графом $G<N,M_0>$,называемым графом достижимости. Каждая вершина графа $G<N,M_0>$ обозначается вектором достижимой маркировки $M\in R(N,M_0)$.

Маркировка, при которой ни один переход сети Петри не возбужден, называется тупиковой. В тупиковой маркировке сеть прекращает свое функционирование.

Соответствие между ФК (ПИП) и моделирующей ее сетью Петри установить нетрудно. Для простоты сначала предположим, что каждая операция O_i в ФК (ПИП) $\Phi=\{O_i\}$ выполняется мгновенно и рассматривается как одно событие. Этому событию соответствует в сети Петри $N=<P,T,B,F,M_0>$ переход $t_j \in T$. Локальные состояния в левой части O_i, которые могут иметь разный смысл, в сети Петри отображены условиями возбуждения перехода t_j, выражающимися в виде неравенства (3.27).

Локальные условия завершения, которые для O_i также могут быть самыми разными (от результата элементарной замены символа до результата выполнения программы), в сети Петри отображаются результатами арифметического суммирования целых чисел. Исходному состоянию соответствует начальная, глобальному состоянию - текущая маркировка.

Таким образом, структурное задание ФК (ПИП) Ф отображается в сети Петри N, а динамика обслуживания $\Pi(\Phi,S)$ - в граф достижимости $G(N,M_0)$.

Ниже приводится соответствие понятий, используемых при построении ФК (ПИП), и понятий сетей Петри.

Таблица 1

ФК(ПИП)	Ф	Сеть Петри	N
Операция	O	Переход	t
Ресурс ИП q в состоянии a	(a,q)	Позиция	P
Глобальное состояние	S	Маркировка	M
Исходное состояние	S_0	Начальная маркировка	M_0
Условия готовности операции Oj	S'j	Условия возбуждения	$M \geq B_{(-,j)}$
Условия завершения операции Oj	S"j	Приращение маркировки в результате события t	$M+\Delta M$
Граф обслуживания	$G(\Phi,S_0)$	Граф достижимости	$G(N,M_0)$
Последовательность операций	σ	Последовательность срабатываний переходов	σ

Таким образом, результатом проектирования будет набор ФК, функционирование которых описывается сетями Петри. Построение ПИП из ФК представляет собой композицию сетей Петри. При этом как сами ФК, так и их композиция должны обеспечивать корректную реализацию процесса ИП.

3.6.4. Модель программы интеллектуального противодействия.

Для разработки метода построения программы ИП сначала необходимо создать адекватную модель ПИП с учетом выбранного подхода к ее построению на основе функциональных компонент.

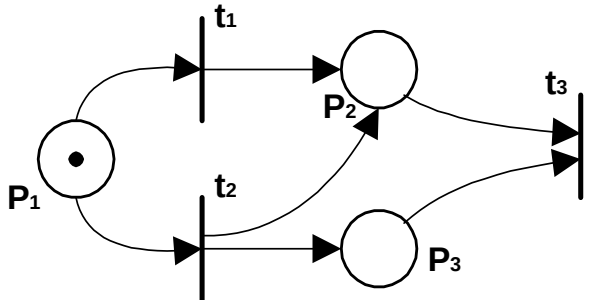


Рис.3.17.Фрагмент свободной сети Петри

Программу ИП будем рассматривать как кортеж:

$$C =< M_C, K >,$$

где $M_{C\{n_C\}} = \{< m_i, A_i >\}$ - множество протокольных модулей (функциональных компонент), $i=\overline{1,n_C}$; n_C - количество модулей ПИП; $A_{i<n_i>} =< a_1, a_2, ...a_{n_i} >$ -вектор атрибутов протокольного модуля.

Определение. Свободная сеть (или сеть со свободным выбором) – это сеть Петри (P, T, B, F, M_0) такая, что $\forall(p,t) \in F : p^\bullet = \{t\} \vee {}^\bullet t = \{p\}$, то есть любая дуга, ведущая от места к переходу, или начинается местом, из которого не выходит ни одной дуги, или заканчивается переходом, в который не ведет никакая другая дуга.

Пример свободной сети показан на рис. 3.17

Каждый модуль ПИП представляется в виде Сети Петри со свободным выбором

m=<P,T,B,F,M>,

где P-множество позиций, T-множество переходов, B:P→T, F:T→P - функции, задающие матрицы инциденции $B_{[m\times n]} = \|b_{ij}\|$ и $F_{[n\times m]} = \|f_{ij}\|$, M- маркировка.

Логика ПИП описывается ее формулой K, которая порождает язык ИП. K - правильно построенная формула ПИП, порождающая язык $S(M_c)$.

Правила построения формул [ ]:

1. $M_c \in S(M_c)$, $M_c \subset WF$.

2. Если $A, B \in S(M_c)$, $A,B \subset WF$, то $\forall K_i(A,B)$, $K_i(A,B) \in S(M_c)$, $K_i(A,B) \subset WF$,

где K_i -множество операций композиции, i=1,m.

WF - условия корректности протокола.

Таким образом, построив корректные протокольные модули (ФК) и правильно выбрав язык композиции ФК в ПИП, сохраняющий корректность результирующего протокола, мы получим механизм создания ПИП из ФК.

3.6.5. Способы построения корректных функциональных компонент интеллектуального противодействия

Построение любого протокола должно отвечать свойствам корректности.

Традиционно выделяют следующие свойства протокола, которые необходимо доказать при анализе корректности [60]:

- отсутствие статических блокировок.
- полнота.
- однозначность.
- отсутствие избыточности.
- ограниченность.
- отсутствие динамических блокировок.
- завершаемость.
- самосинхронизация.

При моделировании протоколов ИП с помощью СП свойства корректности ПИП соответствуют определенным свойствам СП. В частности, свойству 1 соответствует живая Сеть Петри; свойствам 2, 3, 4 соответствует СП с Т-инвариантами, соответствующим спецификации протокола; свойству 5 соответствует ограниченная СП; свойствам 6, 8 – СП без непроизводительных Т-инвариантов, а свойству 7 – свойство достижимости терминальных маркировок у СП.

Таблица 2

Свойство корректности	Описание	Свойство сети Петри
Отсутствие статических блокировок	Отсутствие тупиков, случаев бесконечного ожидания события	$\forall t, \exists \sigma : t \in \sigma$
Отсутствие динамических блокировок	Отсутствие циклов, не предусмотренных спецификацией протокола	$\forall \sigma, \sigma \in P(S)$
Ограниченность	Отсутствие состояний протокола, при которых возможно переполнение буферов	$\exists f : f * C = 0$
Завершаемость	Наличие последовательности смены состояний, завершающейся конечным состоянием	$\exists \sigma : end \in \sigma$
Полнота	Реакция протокола на все входные сообщения	$\forall a \in A, \exists F : A \to \Omega,$ $\forall \sigma \in \Omega, \exists F^{-1} : \Omega \to A$

Соответствие свойств корректности протоколов свойствам сетей Петри приведено в таблице .

Следовательно, для проверки корректности моделируемого протокола ИП необходимо, чтобы сеть Петри, его описывающая, обладала этими свойствами.

Таким образом, для построения корректных ФК необходимо решить задачу синтеза сети Петри с заданными свойствами.

Для решения задачи построения корректной ФК в настоящей работе предложено два способа:

- покрытие сети Петри со свободным выбором (FC-сети) параллельными автоматными подсетями (автоматными компонентами);
- покрытие FC-сети альтернативными маркированными графами.

Суть способа покрытия сети Петри автоматными компонентами (АвК) заключается в следующем.

1.По спецификации ФК строится структура сети Петри. При этом первоначально строится последовательная структура с возможностью условных ветвлений. В случае моделирования события распараллеливания процесса, этот переход помечается как переход синхронизации и моделируется первая ветвь параллельного процесса.

2. Построенная автоматная компонента анализируется на связность и достраивается, если необходимо, до сильносвязной. При этом сильная связность автоматной сети является необходимым и достаточным условием ее живости и безопасности.

3. Каждая ветвь, моделирующая параллельный процесс в сети Петри, представляется в свою очередь как автоматная компонента с единственной входной позицией и одной выходной.

4. Построенная автоматная компонента встраивается в первоначальную сеть через переходы синхронизации.

Доказательство сводимости этого способа к построению сетей, удовлетворяющих теоремам Хака и Коммонера очевидно. После построения первой компоненты, ее сильная связность означает, что в ней каждый тупик содержит ловушку.

При композиции последующих компонент должно удовлетворяться следующее правило. Каждый переход, синхронизирующий окончание компоненты с исходной сетью, входит в стационарно повторяющуюся последовательность срабатывания (t-инвариант), содержащую заданный переход, инициализирующий эту компоненту.

Это правило позволяет утверждать, что в сети каждому процессу размножения фишек соответствует процесс их изъятия, а полученная в итоге композиции сеть Петри будет живой и безопасной.

Последнее условие, обеспечивающее корректность проектируемой сети Петри - условие начальной маркировки. Так как в данном случае минимальным тупиком является вся сеть Петри - следовательно, сеть Петри должна иметь ровно одну фишку в начальной позиции.

Способ разложения сети Петри на МГ-компоненты по своей сути напоминает способ покрытия сети Петри автоматными компонентами с той лишь разницей, что строится реверсивно-двойственная сеть.

3.6.6. Правило начальной маркировки

$$\forall C(p):\ \forall p_i \in C(p), \sum_{i=0}^{n} |M(p_i)| = 1.$$

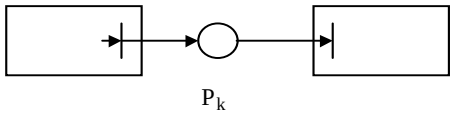
Различия способов маркированных графов и автоматных компонент связано, прежде всего, в физике описываемых процессов. Если процессы, содержащие преимущественно последовательные процессы с ветвлениями, удобнее представлять в виде автоматных компонент, то параллельные процессы проще представлять в виде композиции маркированных графов.

Данные способы позволяют решить задачу построения сети Петри, эквивалентной с точки зрения протокольной интерпретации, то есть с точки зрения эквивалентности порождаемых помеченных языков.

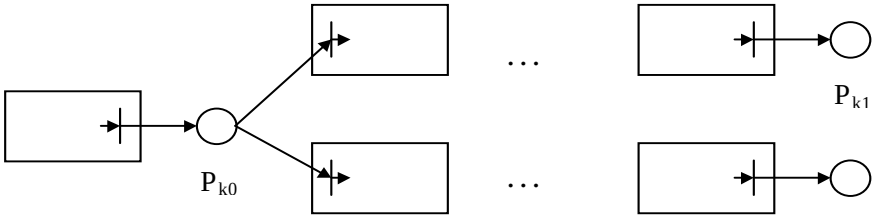
Построенные с применением описанных выше способов протокольные блоки (ФК) являются корректными. Таким образом, для построения корректных ФК сначала производится анализ сети Петри на тяготение к автоматам или МГ. Это делается путем сопоставления количества переходов распараллеливания и позиций ветвления. В случае тяготения к автоматам, применяется алгоритм анализа автоматных покрытий сети Петри. В случае тяготения к МГ, применяется алгоритм разложения сети Петри на МГ-компоненты.

Композиция ФК при построении ПИП может производиться несколькими способами:

1. Последовательная композиция.

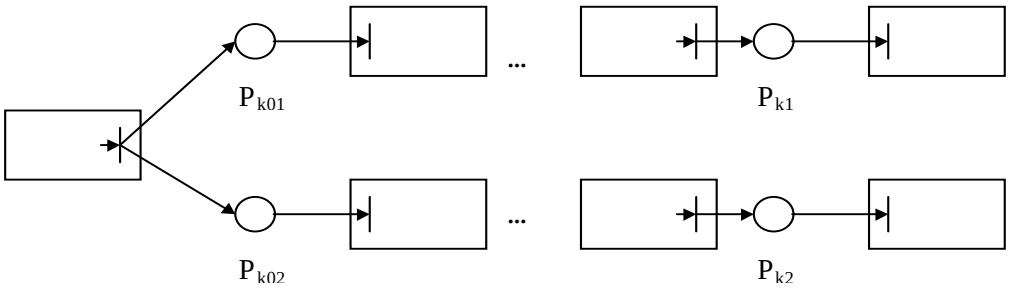


2. Альтернативная композиция.



$P_{k1} = P_{k2}$

3. Параллельная композиция.



$$(P_{k1})^{\bullet} = (P_{k2})^{\bullet},$$
$$N_1, N_2 \subset LS, R(N),$$
$$(N = N_1 \circ N_2) \subset LS.$$

При этом каждая ветвь альтернативной композиции должна завершаться последовательной композицией в одну и ту же позицию, а каждая ветвь параллельной композиции - последовательной композицией в один и тот же блок [45].

Для построения ПИП из ФК целесообразно использовать композиционные методы, предложенные в работе [66] для проектирования интегрированных услуг в СОИ. Эти методы в данной работе не рассматриваются.

Обобщим сказанное выше. Каждая ПИП базируется на некотором протоколе ИП.

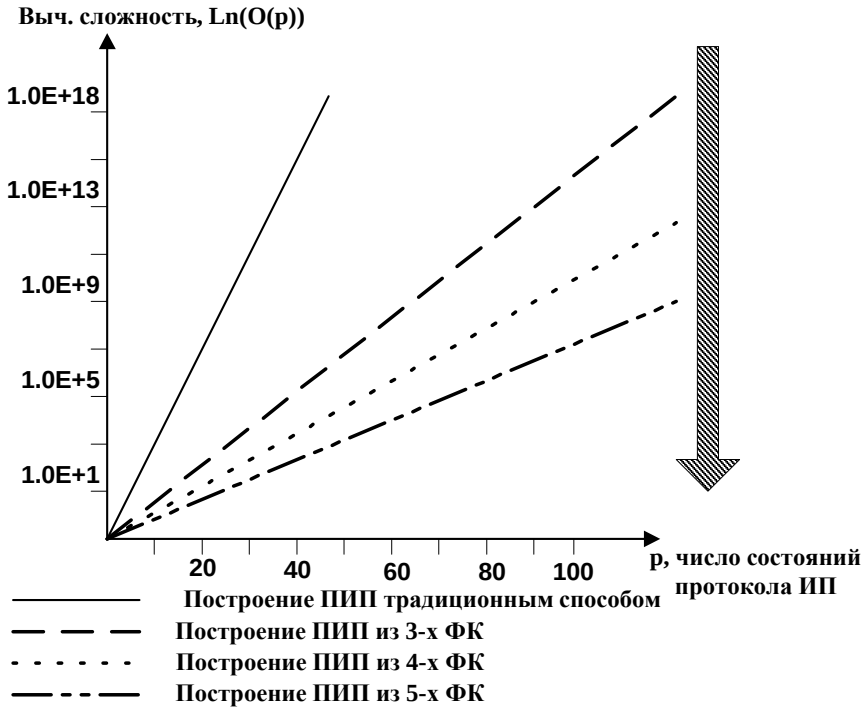


Рис. 3.18. Снижение порядка сложности построения ПИП за счет многомодульной композиции

Процесс построения корректных протоколов ИП можно разбить на два этапа:

Построение протокольных блоков (ФК), удовлетворяющих требованиям корректности и соответствующих спецификациям.

Композиция ФК в ПИП с сохранением свойств корректности.

При этом, часто используемые в ИП процедуры необходимо реализовать в виде ФК в первую очередь, а остальные ФК – по мере необходимости.

В первой главе работы мы отметили, что информационные боевые действия противника в СОИ в конечном счете приобретают форму удаленных атак, то есть «взаимодействие» процессов противника и объекта атаки (ложного объекта атаки) во многом повторяет взаимодействие процессов обычных пользователей СОИ. При этом, естественно, для ЛОА добавляются функции ИП. Это обуславливает то, что для построения ПИП можно использовать в качестве подмножества ФК множество ФК интегрированных услуг СОИ, порядок построения и использования которых достаточно подробно описан в работе [66].

Имея в своем распоряжении достаточный набор ФК и удобную среду их композиции, субъект ИБ может получать достаточно сложные ПИП, наиболее точно отвечающие текущим потребностям по ведению информационных боевых действий в СОИ.

3.6.7. Анализ результатов построения программы интеллектуального противодействия

Предложенный метод построения ПИП позволяет проектировать программы интеллектуального противодействия, которые работают по протоколам, удовлетворяющим требованиям корректности. применение предложенного метода позволяет существенно сократить время разработки специального математического обеспечения интеллектуального противодействия за счет повторного использования функциональных компонент. Понижение порядка сложности решаемой задачи (рис. 3.18.) позволяет добиться существенного повышения оперативности обслуживания субъекта ИБ, выражающееся в предоставлении ПИП,

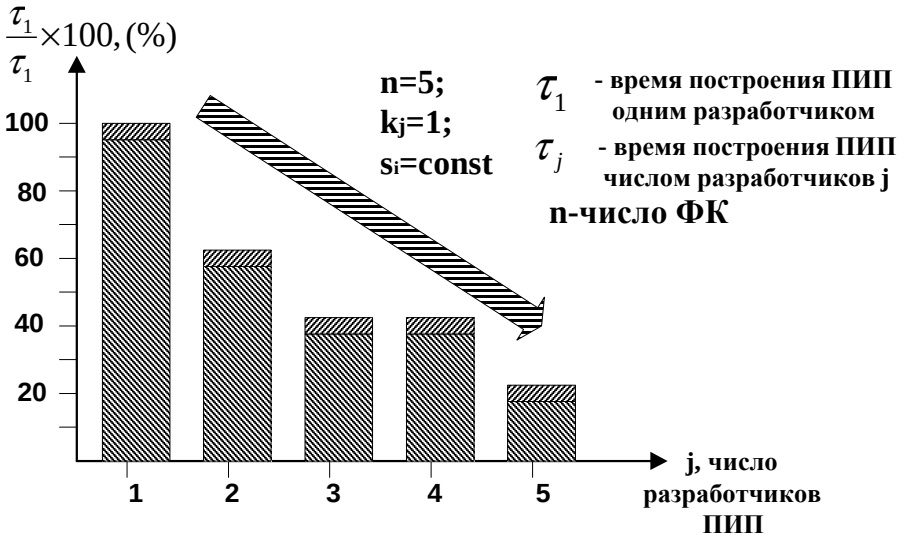


Рис. 3.19. Снижение времени построения ПИП за счет привлечения нескольких разработчиков

удовлетворяющей спецификации.

Наряду с этим, многомодульная композиция позволяет снизить время построения ПИП за счет привлечения нескольких разработчиков (рис. 3.19). Таким образом, предложенный метод по своим временным характеристикам вполне приемлем для использования в СОИ и позволяет добиться решения сформулированной задачи построения ПИП.

Таким образом, подведем некоторый итог. Второй раздел был полностью посвящен разработке методов интеллектуального противодействия противнику в сети обмена информацией.

Была рассмотрена архитектура построения и принципы функционирования М-сетей. Показано преимущество генетических алгоритмов для решения оптимизационных задач большой размерности. Предложен новый алгоритм настройки М-сети. Алгоритм относится к классу генетических алгоритмов оптимизации и позволяет устранить основной недостаток М-сети – сложность настройки ее параметров. Обоснована необходимость использования М-сети для решения задач интеллектуального противодействия противнику в сети обмена информацией.

Описан метод построения программы интеллектуального противодействия в сети обмена информацией, базирующийся на многомодульных композиционных способах построения протоколов программ, обеспечивающих максимальную гибкость и оперативность решения задач. Приведены экспериментальные результаты.

Глава 4. Методика интеллектуального противодействия информационному нападению в корпоративной информационно-вычислительной сети и рекомендации по развитию системы безопасности

4.1. Методика интеллектуального противодействия в сети обмена информацией системы управления силами запуска и управления

При осуществлении интеллектуального противодействия в корпоративной сети обмена информацией исходными данными являются спецификация НСД, структурные и функциональные характеристики СОИ, спецификация ПИП, множество ФК, имеющее описание в виде множества специфицированных реализаций Ω_i .

Методика интеллектуального противодействия противнику в СОИ состоит из четырех основных этапов.

1. Выбор вида интеллектуального противодействия.
2. Выбор зон ИП. Перераспределение не целевой нагрузки в СОИ.
3. Построение ПИП из ФК.
4. Собственно осуществление ИП.

Разработанные в предыдущей главе методы предназначены для реализации первых трех этапов методики ИП.

Первый этап методики ИП заключается в том, чтобы по внешним проявлениям НСД (спецификации НСД) с учетом целей и возможностей ведения ИБ выбрать вид ИП (тип стратегии ИП). Тип стратегии ИП – это выбранный ЛОА и цель его использования, а также некоторые другие характеристики ИП.

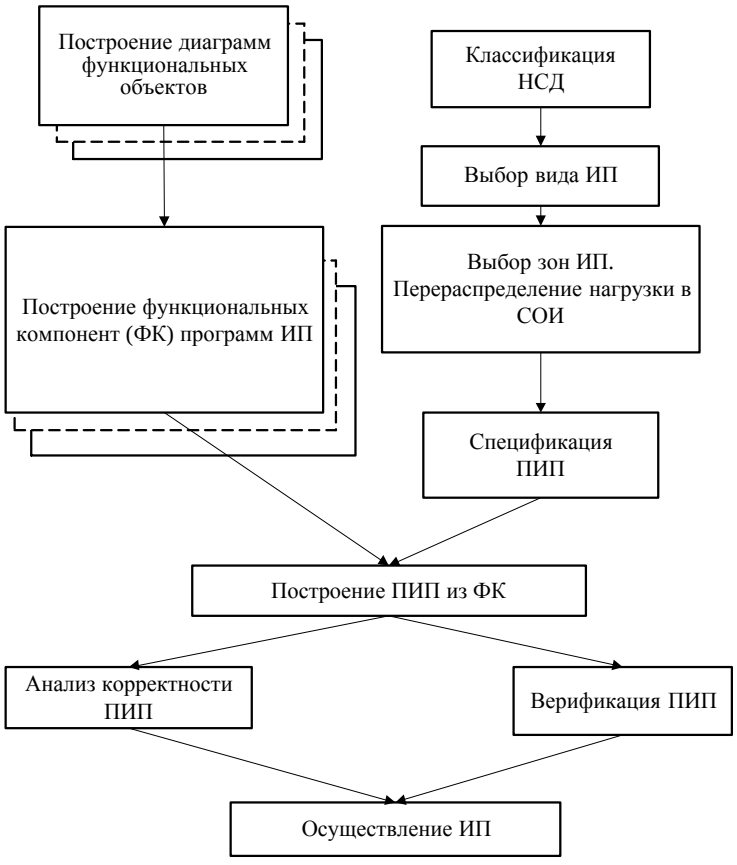


Рис. 4.1. Методика интеллектуального противодействия противнику в сети обмена информацией

На втором этапе решается задача выбора зон интеллектуального противодействия, в ответь на действия противника. Выбор зон ИП осуществляется с учетом снижения нагрузки не целевых потоков в СОИ и возможностей по применению ЛОА в этой зоне.

Если на основе имеющегося множества ФК возможно построить ПИП, то на следующем этапе методики ИП выполняются следующие действия.

На основании спецификации ПИП строится композиция ФК. Выбор композиции осуществляется на основании спецификации путем задания отношения $R_1(\Phi_1, \Phi_2, \ldots, \Phi_k)$, где Φ_i - сеть Петри, описывающая i-ю ФК, на множестве $\Omega=\Omega_1 \times \Omega_2 \times \ldots \times \Omega_k$.

При осуществлении этого вида ИП субъект ИБ вводит данные, путем задания аналогичного отношения R_2 и отношения M, определяющего начальную маркировку.

Затем проверяются свойства корректности сети и соответствие ее

спецификации.

Если имеющихся ФК недостаточно, то с помощью описанных ранее методов строится один или несколько дополнительных ФК. После этого выполняется их композиция описанным выше способом. Описанная методика приведена на рис. 4.1.

4.2. Рекомендации по развитию методов информационной борьбы в сети обмена информацией

4.2.1. Вариант построения в сети обмена информацией системы интеллектуального противодействия противнику с использованием предложенных методов

Рассматривая в первом разделе работы вопросы информационной борьбы в СОИ, мы отметили, что для осуществления противодействия противнику традиционная СЗИ должна быть дополнена системой интеллектуального противодействия. Предлагается при разработке данной СИП использовать методы и методику, предложенные в настоящей работе. В качестве общего подхода построения СИП выбрана концепция интеллектуальной сети, подробно описанная в работах [48, 66], и используемая для построения систем управления интегрированными услугами сетей связи. В нашем случае в качестве такой специфической услуги выступает ПИП, предоставляемая системой ИП субъекту ИБ.

Система ИП состоит из следующих основных модулей:

- модуль выбора вида противодействия (МВП);
- модуль выбора зоны интеллектуального противодействия (МВЗ);
- модуль построения программы интеллектуального противодействия (МПИП).

Модули СИП предлагается размещать на центрах безопасности (ЦБ), расширив тем самым их традиционные функции управления защитой информации в СОИ функциями подготовки интеллектуального противодействия противнику в СОИ в процессе информационной борьбы.

Модуль МВП, получив вектор спецификации НСД от модуля контроля доступа к информационному ресурсу, выполняет задачи классификации НСД и выбора типа стратегии противодействия. Модуль имеет развитый интерфейс для связи с субъектом ИБ (администратором сети), позволяющий последнему вмешиваться в процесс принятия решения на любой его стадии. В качестве основного “мозгового” элемента МВП использует программную реализацию М-сети. На ЦБ также целесообразно разместить модуль настройки , используемый субъектом ИБ для оптимизации параметров М-сети. Функционирование МВП осуществляется с использованием метода, описанного в разделе 3.1.

Исходными данными для модуля МВЗ служат выбранные типы стратегий противодействия противнику как результат работы МВП и параметры СОИ, полученные от системы управления сетью, в частности, от центра управления сетью (ЦУС), а также информация, хранимая в рабочей базе данных (РБД) ЦБ. Функционирование МВЗ осуществляется с использованием метода, описанного ранее в разделе 3.2.

Центральным элементом СИП является модуль МПИП. Он используется СИП для построения ПИП в соответствии с выбранным типом стратегии ИП и спецификацией, вводимой субъектом ИБ. Примечательно, что для выполнения своих функций может использоваться как МПИП ЦБ исходной зоны СОИ (зоны, где расположен объект атаки), так и МЦИП выбранной зоны противодействия (зоны, где планируется разместить ложный объект атаки). Этот выбор зависит во многом от иерархии центров безопасности. Для построения ПИП модуль МПИП использует либо уже имеющиеся ПИП, либо композицию ФК, реализованных программно и хранимых в РБД ЦБ. Функционирование МПИП осуществляется с использованием метода, описанного в разделе 3.3.

Программа интеллектуального противодействия представляет собой реализацию в СОИ ложного объекта атаки, «взаимодействующего» с противником по специальному протоколу ИП и модуля управления им со стороны субъекта ИБ. Ложный объект атаки размеща-

ется на некотором узле (возможно на ЦБ) выбранной для интеллектуального противодействия зоны СОИ.

Таким образом, последовательность работы основных модулей СИП реализует методику ИП, описанную в разделе 4.1.

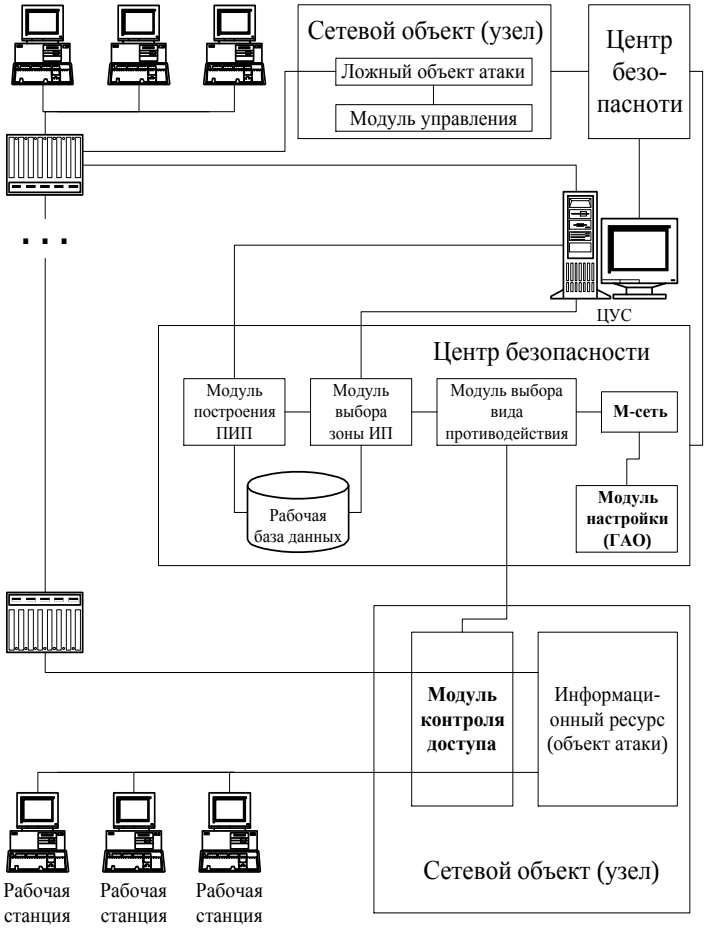
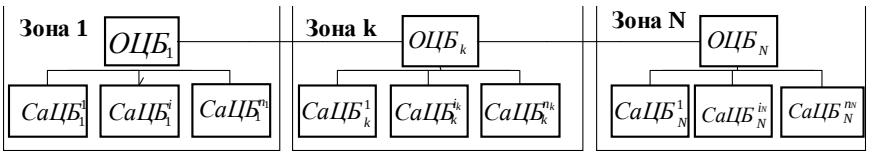


Рис. 4.2. Структурная схема подсистем интеллектуального противодействия во взаимосвязи с другими объектами СОИ

Вариант построения СИП, размещения ее функциональных элементов в СОИ и взаимосвязь с другими элементами СОИ приведен на рис. 4.2.

Как нетрудно заметить, важным и, несомненно, ключевым элементом СИП является рабочая база данных, хранящая готовые ПИП и ФК. Очевидно, что нецелесообразно хранить весь набор ПИП и ФК на всех центрах безопасности, хотя бы потому, что в некоторых зонах СОИ большинство из них, может быть, не понадобятся никогда, но предсказать заранее необходимый набор ПИП и ФК в каждой зоне также невозможно.

Поэтому РБД должна быть единой, но распределенной. При этом наряду с основной БД (ОБД) в основном ЦБ (ОЦБ) имеется сателлитная БД (СаБД) в сателлитном ЦБ (СаЦБ). Поскольку частота осуществления разных видов ИП меняется, целесообразно предусмотреть динамический механизм смены содержимого сателлитной БД при изменении набора используемых ПИП и ФК. Таким образом, должен быть выбран размер сателлитной БД и алгоритм замещения ПИП и ФК в сателлитной БД. Так как РБД распределена по ЦБ СОИ и так как в СОИ существует несколько зон со своими ОБД, то возникает задача минимизации числа пересылок ПИП и ФК между ОЦБ различных зон. Следует отметить, что в общем случае количество уровне распределенной РБД может быть больше двух, что во многом определяется организацией системы центров безопасности СОИ. Распределение РБД по ЦБ СОИ показано на рис. 4.3.



$ОЦБ_k$ – основной ЦБ зоны k, на котором раположена ОБД зоны k; $k = \overline{1, N}$, где N – общее число зон СОИ; $СаЦБ_k^{i_k}$ – сателлитный ЦБ i_k зоны k, на котором раположена СаБД i_k зоны k; $i_k = \overline{1, n_k}$; где n_k – число СаЦБ зоны k.

Рис. 4.3. Двухуровневая система центров безопасности в СОИ с зоново-децентрализованной организацией РБД

4.2.2. Направления дальнейших исследований в области информационной борьбы в сетях обмена информацией

Важным направлением дальнейших исследований является разработка эффективных методов обнаружения несанкционированных действий противника. Используемые в настоящее время, методы статистической оценки интенсивности работы отдельного узла сети, не обладают достаточной точностью классификации. В силу этого значительное число срабатываний системы безопасности является ложным. Другой вид обнаружения НСД, основанный на логическом выводе, также не обладает достаточным набором положительных свойств, чтобы удовлетворять всем требованиям защиты. Основным его недостатком является отсутствие модели внешнего мира, этот недостаток в свое время привел к упадку экспертных систем и языков логического программирования.

На наш взгляд перспективным направлением исследований в этой области может служить разработка средств классификации НСД на основе технологии нейронных сетей и дальнейшее развитие методов на основе искусственной жизни и генетических алгоритмов.

Другое важное направление исследований – разработка протоколов взаимодействия системы контроля доступа к информационным ресурсам и системы интеллектуального противодействия СОИ.

Весьма интересным направлением, на наш взгляд, может служить задача создания на основе анализа используемых в СОИ информационных технологий, протоколов функционирования типовых ложных объектов атаки в виде композиции функциональных компонент.

Интересно также разработать протоколы взаимодействия системы интеллектуального противодействия с автоматизированными рабочими местами абонентов СОИ, позволяющие администратору системы безопасности использовать СИП для решения текущих задач информационной борьбы в СОИ.

Мы не ограничиваемся этими задачами и надеемся, что технология интеллектуального противодействия в сетях обмена данными корпоративного масштаба позволит решать многие проблемы безопасности информации.

По четвертой, заключительной главе нашей работы, можно сделать следующие выводы.

В разделе предложена методика интеллектуального противодействия в сети обмена информацией, основывающаяся на методах, предложенных в настоящей работе.

Приведен вариант расположения элементов СИП в СОИ и их взаимосвязей с другими элементами СОИ (системы управления СОИ, МКД и др.).

Обоснована структура и размещение РБД, хранящей ПИП и ФК и ее взаимосвязь со структурой системы центров безопасности.

В качестве направлений дальнейших исследований выделены следующие:

- так как работа посвящена в основном методам подготовки ИП, то необходимо разработать методы и способы собственно интеллектуального противодействия (обосновать первоначальный набор ложных объектов атаки, соответствующих наборов ПИП и ФК, правила взаимодействия субъекта ИБ с ПИП и т.д.).
- разработка алгоритма замещения ПИП и ФК в сателлитных БД;
- решить задачу минимизации числа пересылок ПИП между ОБД различных зон СОИ;
- исследовать возможность использования моделей биологических объектов при разработке новых методов информационного противодействия в СОИ (нейронные сети, искусственная иммунная система и т.п.).

Заключение

В заключении к нашей работе, посвященной интеллектуальному противодействию информационному нападению в сетях обмена данными, хотелось бы подвести краткий итог всему изложению.

Считаем, что нам удалось провести краткий, но достаточно полный анализ современных средств информационного алгоритмического воздействия на программные комплексы информационно-вычислительных сетей. Дан краткий анализ возможных удаленных атак на информационные ресурсы.

Проведен анализ целей и задач информационной борьбы в СОИ и предложен подход к осуществлению противодействия с использованием средств искусственного интеллекта, основанный на понятии ложного объекта атаки.

Обоснован выбор аппарата М-сетей для решения задачи определения стратегии противодействия и предложен новый алгоритм настройки параметров М-сети, относящийся к классу генетических алгоритмов оптимизации.

Исследовано влияние информационных потоков, порождаемых противником, на загрузку СОИ. Предложен подход к снижению этой нагрузки на СОИ путем оптимального расположения ложных объектов атаки в сети.

Разработана модель ложного объекта атаки, учитывающая композиционный характер построения и особенности применения в СОИ корпоративного масштаба.

Предложен новый метод выбора стратегии противодействия информационным воздействиям противника на основе М-сети.

Предложен метод выбора зон интеллектуального противодействия, позволяющий располагать ЛОА в СОИ, обеспечивая минимум загрузки сети.

Предложен композиционный метод построения ЛОА, обеспечивающий активное участие администратора сети в процессе интеллектуального противодействия противнику.

Предложена методика интеллектуального противодействия противнику в СОИ, основанная на разработанных в работе методах.

Надеемся, что время, потраченное читателем на изучение предложенной работы, не прошло даром. Читатель заинтересуется предложенным в работе направлением защиты информации и отклики на книгу, позволят авторам и далее развивать предложенные методы.

Приложение

Данное приложение содержит исходный текст программы, моделирующей генетический алгоритм оптимизации, использованный в работе.

```
/*********************************************************************/
/* Программная реализация генетического алгоритма, в которой         */
/* целевая функция принимает только положительные значения           */
/* и пригодность индивида принимается равной значению                */
/* целевой функции. Данный генетический алгоритм реализует все       */
/* основные операции: селекция, инверсия, мутация, кроссинговер      */
/*********************************************************************/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

/* При необходимости можно изменить эти параметры */
#define POPSIZE 50      /* размер популяции */
#define MAXGENS 1000    /* максимальное значение поколений (циклов) */
                        /* генетического алгоритма                  */
#define NVARS 3         /* число заданных переменных */
#define PXOVER 0.8      /* вероятность кроссинговера */
#define PMUTATION 0.15  /* вероятность мутации */
#define PINVERSION 0.1  /* вероятность инверсии */
#define TRUE 1
#define FALSE 0

int generation;  /* номер текущего поколения */
int cur_best;    /* лучший индивид */
FILE *galog;     /* файл вывода */
int index1[NVARS]; /* вспомогательный индексов переменных */
int index2[NVARS]; /* массив индексов переменных */

struct genotype /* генотип (член популяции) */
{
double gene [NVARS];       /* строка переменных */
double fitness;            /* пригодность генотипа */
double upper [NVARS];      /* значения верхних границ переменных */
double lower [NVARS];      /* значения нижних границ переменных */
double rfitness;           /* относительная пригодность */
double cfitness;           /* накопленная пригодность */
};

struct genotype population[POPSIZE+1];     /* популяция */
struct genotype newpopulation[POPSIZE+1];  /* новая популяция */
struct genotype newpopulation1[POPSIZE+1];  /* новая популяция1*/

/* Объявление процедур, которые используются данной программной */
/* реализацией генетического алгоритма                          */
void initialize(void);
double randval (double, double);
int randval1 (double, double);
void evaluate(void);
void keep_the_best(void);
void elitist(void);
void select(void);
void crossover(void);
void Xover(int, int);
void swap(double *, double *);
void mutate(void);
void report(void);
void inversion(void);
```

```
/************************************************************/
/* Функция инициализации: инициализирует значения генов в   */
/* пределах заданных границ переменных. Она также инициализиру-*/
/* ет (обнуляет) все значения пригодности для каждого члена по-*/
/* пуляции. Она читает верхние и нижние границы каждой перемен-*/
/* ной из файла ввода "gadata.txt". Она генерирует случайным  */
/* образом значения в рамках границ для каждого гена каждого  */
/* генотипа в популяции. Формат файла ввода "gadata.txt"      */
/* следующий:                                                 */
/* переменная1_нижняя_граница переменная1_верхняя_граница     */
/* переменная2_нижняя_граница переменная2_верхняя_граница . . .*/
/************************************************************/

void initialize(void)
{
FILE *infile;
int i, j;
double lbound, ubound;

                /* открытие файла ввода */
        if ((infile = fopen("gadata.txt","r"))==NULL)
        {
                fprintf(galog,"\nНе удается открыть файл ввода!\n");
                exit(1);
        }

/* инициализация переменных с учетом заданных границ */
        for (i = 0; i < NVARS; i++)
        {
                fscanf (infile, "%lf",&lbound);
                fscanf (infile, "%lf",&ubound);
                index1[i] = index2[i] = i;
                for (j = 0; j < POPSIZE; j++)
                {
                population [j].fitness = 0;
                population[j].rfitness = 0;
                population[j].cfitness = 0;
                population [j].lower [i] = lbound;
                population[j].upper[i]= ubound;
                population[j].gene[i] = randval(population[j].lower[i],
                                                population[j].upper[i]);
                }
        }
 fclose(infile); /* закрывается файл ввода */
 }
/************************************************************/
/* Генератор случайного значения: генерирует значение в     */
/* заданных границах                                        */
/************************************************************/
double randval (double low, double high)
{
        double val;
        val = ((double) (rand()%1000)/1000.0)*(high - low) + low;
        return(val);
}
/************************************************************/
/* Генератор случайного значения: генерирует значение в     */
/* заданных границах                                        */
/************************************************************/
int randval1 (double low, double high)
{
        double val;
        val = ((double)(rand()%1000)/1000.0)*(high - low) + low;
        return(val);
```

```
}

/*************************************************************/
/* Функция оценивания: Она содержит определенную пользовате-*/
/* лем целевую функцию. После каждого изменения код должен  */
/* быть перекомпилирован.                                    */
/* Текущая функция: x[1]~2-x[1]*x[2]+x[3]                    */
/*************************************************************/

void evaluate(void)
{
int mem;
int i;
double x[NVARS+1];

for (mem = 0; mem < POPSIZE; mem++)
        {
        for (i = 0; i < NVARS; i++)
                        x[index2[i]+1] = population [mem].gene [i];

        population[mem].fitness = (x[1]*x[1]) - (x[1]*x[2]) + x[3];
        }
}

/*************************************************************/
/* Функция сохранения наилучшего члена популяции: Эта функция   */
/* хранит след лучшего члена популяции. Последний элемент масси-*/
/* ва population содержит копию лучшего индивида             */
/*************************************************************/

void keep_the_best()
{
int mem;
int i;
cur_best = 0; /* хранит индекс лучшего индивида */
for (mem = 0; mem < POPSIZE; mem++)
{
if (population[mem].fitness >   population[POPSIZE].fitness)
{
cur_best = mem;
population[POPSIZE].fitness =   population[mem].fitness;
}
}

/* после определения лучшего члена популяции, копируем гены */
for (i = 0; i < NVARS; i++)
population[POPSIZE].gene[i] = population[cur_best].gene[i];
}

/*************************************************************/
/* Лучший член предыдущего поколения хранится                */
/* как последний в массиве. Если лучший член текущего поколения*/
/* хуже, чем лучший член предыдущего поколения, последний дол- */
/* жен заменить наихудший член текущей популяции             */
/*************************************************************/

void elitist()
{
int i;
double best, worst;      /* лучшее и худшее значение пригодности */
int best_mem, worst_mem; /* индексы лучшего и худшего члена */

best = population[0].fitness;
worst = population[0].fitness;
for (i = 0; i < POPSIZE - 1; ++i)
```

```
{
if (population[i].fitness > population[i+1].fitness)
{
if (population[i].fitness >= best)
{
best = population[i].fitness;
best_mem = i;
}
if (population[i+1].fitness <= worst)
{
worst = population[i+1].fitness;
worst_mem = i+1;
}
}
else
{
if (population[i].fitness <= worst)
{
worst = population[i].fitness;
worst_mem = i;
}
if (population[i+1].fitness >= best)
{
best = population[i+1].fitness;
best_mem = i+1;
}
}
}
/* Если лучший индивид новой популяции лучше, чем лучший ин- */
/* дивид предыдуще популяции, тогда копировать лучший индивид*/
/* из новой популяции; иначе - заменить худший индивид теку- */
/* щей популяции на лучший индивид из предыдущего поколения  */

if (best >= population [POPSIZE].fitness)
{
for (i = 0; i < NVARS; i++)
population[POPSIZE].gene[i] = population[best_mem].gene[i];
population[POPSIZE].fitness = population[best_mem].fitness;
}
else
{
for (i = 0; i < NVARS; i++)
population[worst_mem].gene[i] = population[POPSIZE].gene[i];
population[worst_mem].fitness = population[POPSIZE].fitness;
}
}
/*************************************************************/
/* Функция селекции:                                         */
/*              Стандартная однородная селекция              */
/*************************************************************/

void select(void)
{
int mem, i, j, k;
double sum = 0;
double p;
/* поиск общей пригодности популяции */
for (mem = 0; mem < POPSIZE; mem++)
{
sum += population[mem].fitness;
}
/* вычисление абсолютной пригодности */
for (mem = 0; mem < POPSIZE; mem++)
{
population[mem].rfitness = population[mem].fitness/sum;
```

```
}
population[0].cfitness = population[0].rfitness;


/* вычисление накопленной пригодности */
for (mem = 1; mem < POPSIZE; mem++)
{
population[mem].cfitness = population[mem-1].cfitness +
                                        population [mem].rfitness;
}

/* окончательный выбор, используя накопленную пригодность. */

for (i = 0; i < POPSIZE; i++)
{
  p = rand()%1000/1000.0;
  if (p < population[0].cfitness)
newpopulation[i]=population[0];
else
{
for (j = 0; j < POPSIZE;j++)
if (p >= population[j].cfitness &&
p<population[j+1].cfitness)
newpopulation[i] = population[j+1];
}
}

/* как только новая популяция сформирована -копировать ее обратно */

for (i = 0; i < POPSIZE; i++) population[i] = newpopulation[i];
}
/*************************************************************/
/*  Селекция для кроссинговера: выбирает двух родителей,      */
/*         которые принимают участие в кроссинговере          */
/*************************************************************/

void crossover(void)
{
int i, mem, one;
int first = 0; /* счетчик числа выбранных членов */
double x;

for (mem = 0; mem < POPSIZE; ++mem)
{
x = rand()%1000/1000.0;
if (x < PXOVER)
{
++first;
if (first % 2 == 0)
Xover (one, mem);
else
one = mem;
}
}
}

/*************************************************************/
/* Кроссинговер: выполняет кроссинговер двух выбранных членов */
/*************************************************************/

void Xover(int one, int two)
{
int i;
int point; /* позиция кроссинговера */
```

```
/* выбор позиции кроссинговера */
if(NVARS > 1)
{
if(NVARS == 2)
point = 1;
else
point = (rand () % (NVARS - 1)) + 1;

for (i = 0; i < point; i++)
swap(&population[one].gene[i], &population[two].gene[i]) ;
}
}
/*************************************************************/
/* Обмен: Функция обмена значениями двух переменных          */
/*************************************************************/

void swap (double *x, double *y)
{
double temp;

temp = *x;
*x = *y;
*y = temp;
}

/*************************************************************/
/* Мутация: Переменная, выбранная для мутации заменяется     */
/* случайным значением между нижней и верхней границей этой  */
/* переменной.                                               */
/*************************************************************/

void mutate (void)
{
int i , j ;
double lbound, hbound;
double x;

for (i=0; i < POPSIZE; i++)
for (j = 0; j < NVARS; j++)
{
 x = rand()%1000/1000.0;
 if (x < PMUTATION)
{
/* определение границ мутирующей переменной */
lbound = population[i].lower[j];
hbound = population [i].upper [j];
population[i].gene[j] =randval(lbound, hbound);
}
}
}

/*************************************************************/
/* Функция отчета: Содержит результаты моделирования.        */
/* Данные записываются в файл вывода                         */
/*************************************************************/

void report (void)
{
int i;
double best_val;   /* лучшая пригодность в популяции */
double avg;        /* средняя пригодность популяции */
double stddev;     /* среднеквадратическое отклонение */
double sum_square; /* сумма квадратов для вычисления stddev */
double square_sum; /*  */
double sum;        /* общая пригодность популяции */
```

```
sum = 0.0;
sum_square = 0.0;

for (i = 0; i < POPSIZE; i++)
{
sum += population[i].fitness;
sum_square += population[i].fitness * population[i].fitness;
}
avg = sum/ (double) POPSIZE;
square_sum = avg * avg * (double)POPSIZE;
stddev = sqrt((sum_square - square_sum)/(POPSIZE -1));
best_val = population[POPSIZE].fitness;
fprintf(galog, "\n%5d,            %6.3f, %6.3f, %6.3f \n\n",
                              generation, best_val, avg, stddev);
}
/*************************************************************/
/* Инверсия: выполняет инверсию применительно ко всей популяции */
/*************************************************************/

void inversion(void)
{
int i, j, k;
int point [2]; /* левая и правая позиции инверсии */
double x;

x = rand()%1000/1000.0;

/* будет ли осуществляться инверсия в данном цикле */
if (x < PINVERSION)
{
/* нет */
return;
}
/* выбор левой и правой позиции инверсии */
if(NVARS > 1)
{
if(NVARS == 2)
{
 point[0] = 0;
 point[1] = 2;
}
else
{
 point[0] = (rand () % (NVARS - 1));
 point[1] = randval1((double)(point[0]+2),(double)NVARS);
}
for (i = 0; i < point[0]; i++)
{
 index1[i] = index2[i];
 for (k = 0; k <= POPSIZE; k++)
 newpopulation1[k].gene[i] = population[k].gene[i];
}
j = point[1];
for (i = point[0]; i < point[1]; i++)
{index1[i] = index2[j-1];
 for (k = 0; k <= POPSIZE; k++)
 newpopulation1[k].gene[i] = population[k].gene[j-1];
 j--;
}
for (i = point[1]; i < NVARS; i++)
{
 index1[i] = index2[i];
for (k = 0; k <= POPSIZE; k++)
 newpopulation1[k].gene[i] = population[k].gene[i];
```

```
}
for (i = 0; i < point[1]; i++)
index2[i] = index1[i];
/* как только новая популяция сформирована -копировать ее обратно */
for (i = 0; i <= POPSIZE; i++)
        for (j = 0; j < NVARS; j++)
                population[i].gene[j] = newpopulation1[i].gene[j];
}
}
/*************************************************************/
/* Главная функция: Каждый цикл включает в себя следующее
/* выбор лучших членовпопуляции, выполнение кроссинговера, му-*/
/* тации и затем оценивание результирующей популяции.         */
/* Число циклов ограничено значением MAXGEN                   */
/*************************************************************/

void main(void)
{
int i;
/* открытие файла вывода */
if ((galog = fopen("galog1.txt","w"))==NULL)
{
exit (1);
}
generation = 0;
fprintf(galog, "\n номер    лучшее средняя ср. квадр. \n");
fprintf(galog, " цикла     значение пригодн. отклонение\n");

initialize();
evaluate ();
keep_the_best();
while (generation<MAXGENS)
{
generation++;
inversion ();
select ();
crossover ();
mutate ();
report ();
evaluate ();
elitist();
}
fprintf (galog, "\n\n Моделирование завершено\n");
fprintf(galog,"\n Лучший член популяции: \n");

for (i = 0; i < NVARS; i++)
{
fprintf (galog,"\n var(%d) = %3.3f",index2[i],
                                        population[POPSIZE].gene[i]);
}
fprintf (galog,"\n\n Лучшее значение пригодности = %3.3f",
                                         population[POPSIZE].fitness);
fclose (galog);
printf("Успешное завершение программы\n");
}
/*****************************************************************/
```

Список использованных источников

1. Автоматы.- М.: Физматгиз, 1956,- 423с.
2. Айерлэнд К., Роузен М. Классическое введение в современную теорию чисел. Пер. с англ.- М.: Мир, 1987.- 416 с.
3. Албертс Б., Брей Д., Льюис Дж. и др. Молекулярная биология клетки: в 3-х томах 2-е изд., перераб. и доп. Т.3. Пер. с англ. –М: Мир, 1994. – 504 с.
4. Амосов Н.М., Касаткин А.М., Касаткина Л.М., Талаев С.А. Автоматы и разумное поведение. - Киев: Наукова думка, 1973.- 373 с.
5. Артамонов В.А., Яценко В.В. Многоосновные алгебры в системах открытого шифрования. // Успехи математических наук, т. 149, вып. 4, 1994, С.149-150.
6. Барсуков В.С., Водолазский В.В. Интегральная безопасность информационно-вычислительных и коммуникационных сетей// Технологии электронных коммуникаций т.34, т. 35.- М. 1992.
7. Барсуков В.С., Дворянкин С.В., Шеремет И.А. Безопасность связи в каналах телекоммуникаций// Технологии электронных коммуникаций М.: МЦНТИ, т. 20, 1992.- 122 с.
8. Батурин Ю.М., Жодзижский А.М. Компьютерная преступность и компьютерная безопасность. - М.: Юридическая литература, 1991.- 169 с.
9. Баушев С., Тормышов С. Технология АТМ для профессионалов// Компьютер пресс, №10, 1996, С. 26-30.
10. Бияшев Р.Г., Диев С.И., Размахин М.К. основные направления развития и совершенствования криптографического закрытия информации. “Зарубежная радиоэлектроника” №12, 1989 г., с. 76-91.
11. Большаков А.А., Петряев А.Б., Платонов В.В., Ухлинов Л.М. Основы обеспечения безопасности данных в компьютерных системах и сетях. Часть 1. Методы, средства и механизмы защиты данных.- СПб.: Конфидент, 1996.- 166 с.
12. Борисов М. Новые стандарты высокоскоростных сетей// Открытые системы, 1994.- Лето.- С. 20-31.
13. Брикелл Э.Ф., Одлижко Э.М. Криптоанализ: обзор новейших результатов// ТИИЭР, 1988.- Т.76, №5.- С. 75-94.
14. Бронштейн И.Н., Семендяев К.А. Справочник по математике для инженеров и учащихся ВТУЗов.- 13-е изд., исправленное. - М.: Наука, Гл. ред. физ.-мат. лит., 1986.- 544 с.
15. Веденов А.А. Моделирование элементов мышления М.: Наука, 1988, 159с.
16. Викторов А.Ф. Информационная война в современных условиях// Информационное общество, 1997.- №1.- С. 58-59.
17. Военный энциклопедический словарь/ Пред Гл. ред. комиссии С.Ф. Ахромеев.-М.: Воениздат, 1986.- 863 с.
18. Гайкович В.Ю., Першин А.Ю. Безопасность электронных банковских систем.- М.: Единая Европа, 1994.- 368 с.
19. Герасименко В.А. Защита информации в АСОД// в двух частях.- М.: Энергоатомиздат, 1994.
20. Герасименко В.А. Комплексная защита информации в современных системах обмена данными// Зарубежная радиоэлектроника, 1993, № 2.- С. 3-29
21. Герасименко В.А. Проблемы защиты данных в системах их обработки// Зарубежная радиоэлектроника, 1989.- №12.- С. 5-21.
22. Герасименко В.А., Размахин М.К. Криптографические методы в автоматизированных системах// Зарубежная радиоэлектроника, 1982.- №8.- С. 97-123.
23. ГОСТ 1.0-92. Государственная система стандартизации Российской Федерации. Основные положения. Госстандарт России. Москва. Изд. 1993 г.
24. ГОСТ 34.003-90. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения.
25. ГОСТ 50992-96. Защита информации. Основные термины и определения.

26. Гостехкомиссия России. Руководящий документ. Защита от несанкционированного доступа к информации. Термины и определения. М., Воениздат, 1992 г.
27. Гриняев С.Н. Генетический алгоритм обучения стохастической нейронной сети // Приборостроение. Известия ВУЗов, 1996. Т.39, №1. с.23-24
28. Гриняев С.Н. Сравнительный анализ отечественных моделей нейронных сетей // Тезисы докладов III Всероссийского семинара “Нейроинформатика и ее приложения”. Красноярск, 1995.
29. Гриняев С.Н. Метод эволюционно-генетического кодирования структуры нейронной сети // Тезисы докладов II Всероссийской школы “Нейроинформатика-96”. Красноярск, 1996.
30. Гриняев С.Н. Эволюционно-генетическая модель компьютерного вируса // Тезисы докладов III Межведомственной научно-технической конференции “Проблемные вопросы сбора, обработки и передачи информации в сложных радиотехнических системах ”. Пушкин, 1997.
31. Деметрович Я., Кнут Е., Радо П. Автоматизированные методы спецификации: Пер. с англ.- М.: Мир, 1989.- 115 с.
32. Диффи У., Хеллмен М.Э. Защищенность и имитостойкость: введение в криптографию// ТИИЭР, 1979.- Т.63, №3.- С. 71-109.
33. Закон РФ “О безопасности”, № 2446-1 от 5.3.1992 г.
34. Закон РФ “О государственной тайне”, №5485-1 от 21.7.93 г.
35. Защита государственных и промышленных секретов. Методы обнаружения вторжений в вычислительные системы. // Ин. Печать об экономическом, научно-техническом и военном потенциале государств-участников СНГ и технических средств разведывательных служб капиталистических государств./ ВИНИТИ.- 1993.- №7.- С. 8-15.
36. Защита информации в компьютерных системах. Выпуск 2. Элементы криптологии / Под ред. П.Д.Зегжды. - СПб.: ГТУ, 1993.- 147 с.
37. Завадский И.И. Информационная война - что это такое? // Защита информации. Конфидент, 1996.- №4.- С.13-20.
38. Иммуногенетика человека. Основные принципы и клиническое значение. В 2-х т. Т. 1: Пер. с англ./ Под ред. С. Литвина. – М.: Мир, 1994. – 496 с.
39. Казарин О.В., Ухлинов Л.М. Использование свойств эллиптических кривых в криптологических протоколах.// Автоматика и вычислительная техника. - 1992.- № 5.- С. 23-32.
40. Калинин В.Н., Резников Б.А., Варакин Е.И. Теория систем и оптимального управления: Часть 2, Понятия, модели, методы и алгоритмы оптимального выбора, издание 2-е: МО СССР, 1988.- 589 с.
41. Киричевский Р.Е. Сжатие и поиск информации М.: Радио и связь, 1989.- 168 с.
42. Кларк Дж., мл., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи: Пер. с англ.- М.: Радио и связь, 1987.
43. Концепция защиты информации в системах ее обработки. Гостехкомиссия России. 21.3.95 г.
44. Концепция формирования и развития законодательства в сфере информации, информатизации и информационной безопасности в РФ. Журнал “Проблемы информатизации. Вып. 1, 1995 г.”
45. Котов В.Е. Сети Петри. – М.: Наука, 1984.
46. Курош А.Г. Теория групп. - М.: Главная ред. физ.-мат. литературы.- 1967.- 648 с.
47. Куртис Дж. Коммутаторы базовых сетей АТМ: результаты тестирования восьми моделей// Computer Week Mosc., 1996.- №34-35.- С. 13-18, 20, 21, 68, 69.
48. Лазарев В.Г. Интеллектуальные цифровые сети. Справочник. / Под ред. академика Н.А. Кузнецова. - М.: Финансы и статистика, 1996.- 224 с.
49. Липшуц Р.П. Технология АТМ на марше // PC MAGAZINE/ RE, 1996.- Специальный выпуск №3.- С. 82-94.
50. Максимов Ю.Н., Еремеев М.А. Метод защиты каналов передачи информации между

абонентскими станциями сетей связи// Тезисы докладов конференции ВИККА, 1996, с.43.

51. Максимов Ю.Н., Еремеев М.А. Построение криптосистем на основе свойств эллиптических кривых// Безопасность информационных технологий, № 2, 1995, С. 52-55.
52. Мамиконов А.Г., Кульба В.В. Синтез оптимальных модульных систем обработки данных. - М.: Наука, 1986.- 276 с.
53. Масалович А. Тропою создателя. Первые шаги “искусственной жизни”// PC Week/ RE, 1996.- №40 (64).- С. 44-46
54. Месси Дж.Л. Современные методы защиты информации. М.: Советское радио, 1980.- 264 с.
55. Мафтик С. Механизмы защиты в сетях ЭВМ.- М.: Мир, 1993.- 216 с.
56. Островский С.Л. Компьютерные вирусы. Выпуск 3.2.- М.: “Диалог Наука”, 1997.-88 с.
57. Палий А.И. Радиоэлектронная борьба. - 2-е изд. переработанное и дополненное- М.: Воениздат, 1989.- 384 с.
58. Петряев А.Б. Современное информационное оружие и его особенности// Тезисы докладов
59. Перин В.А. Генерация, распределение и использование криптографических ключей,, Защита информации, - 1992, № 1, С. 157-184.
60. Питерсон Дж. Теория сетей Петри и моделирование систем. - М.: Мир, 1984.
61. Положение о государственном лицензировании деятельности в области защиты информации. Утверждено Решением Гостехкомиссии при Президенте РФ и Федерального агентства правительственной связи и информации при Президенте РФ от 27.4.1994 г. №10
62. Прибрам К. Языки мозга М.: Прогресс, 1976, 464с.
63. Присяжнюк С.П., Сидак А.А. Анализ современного состояния теории и практики построения моделей систем защиты информации. // Тезисы докладов III Межведомственной научно-технической конференции “Проблемные вопросы сбора, обработки и передачи информации в сложных радиотехнических системах ”. Пушкин, 1997.
64. Проблемы безопасности программного обеспечения /Под ред. П.Д. Зегжды.- СПб.: ГТУ, 1995.- 200 с.
65. Протоколы и методы управления в сетях передачи данных/ Под ред Ф.Ф.Куо.- М.: Радио и связь, 1985.- 480 с.
66. Резников Б.А. Системный анализ и методы системотехники. Ч.1. Методология системных исследований. Моделирование сложных систем. - М.: МО СССР, 1990.- 522 с.
67. Сидак А.А. Обнаружение попыток несанкционированного доступа в сетях обмена информацией с использованием аппарата генетических алгоритмов // Тезисы докладов X Всероссийской научно-технической конференции “Однородные вычислительные структуры, среды и распределенные системы”(ОВС-97).- Москва 1997
68. Советов Б.А., Яковлев С.А. Построение сетей интегрального обслуживания.- Л.: Машиностроение. Ленингр. отделение, 1990.- 332 с.
69. Советсткий энциклопедический словарь/ Гл. ред. А.М. Прохоров. 2-е изд. -М.: Сов. энциклопедия, 1983.- 1600 с.
70. Справочник менеджера. Терминология в области компьютеризации и информатизации общества. МИА “АПЕЙРОН”. ВНИИКИ, 1991.
71. Справочник терминов по РЭБ, МО, 1991 г.
72. Стенг Д., Мун С. Секреты безопасности сетей.- К.: “Диалектика”, 1995.- 544 с.
73. Сунчелей И. Сети FDDI - принцип действия, применяемое оборудование, варианты использования// Открытые системы, 1994.- Лето.- С. 33-40.
74. Теория и практика обеспечения информационной безопасности/ Под редакцией П.Д. Зегжды М.: Издательство Агенства “Яхтсмен”, 1996.- 192 с.
75. Терминология в области защиты информации. Справочник. 1993 г.
76. Толковый словарь по информатике. М. Финансы и статистика, 1991 г.

77. Удалов В.И., Спринцис Я.П. Безопасность в среде взаимодействия открытых систем// Автоматика и вычислительная техника.- 1990.- №3. С. 3-11.
78. Ухлинов Л.М. Международные стандарты в области обеспечения безопасности данных в сетях ЭВМ// Электросвязь, 1991, №6.- с. 31-36.
79. Ухлинов Л.М. Принципы построения системы управления безопасностью данных// Автоматика и вычислительная техника, 1990.- №5.- С. 11-17.
80. Ухлинов Л.М. Принципы построения системы управления безопасностью данных в ИВС// Автоматика и вычислительная техника, 1990, №4, с. 11-71.
81. Федеральный закон “Об информации, информатизации и защите информации”. № 24-ФЗ от 20.02.95 г.
82. Фатьянов А.А. Проблемы защиты конфиденциальной информации, не составляющей государственную тайну// Информационное общество, 1997.- №1.- С. 49-56.
83. Федотов А. АТМ-коммутаторы// Компьютер пресс, №10, 1996, С. 32-38.
84. Филимонов А.Ф. О разработке в США системы мер по защите национальной информационной инфраструктуры// Информационное общество, 1997.- №1.- С. 60-64.
85. Фролов А.В., Фролов Г.В. Осторожно: компьютерные вирусы.- М.: “Диалог-МИФИ”, 1996.- 256 с.
86. Халяпин Д.Б., Ярочкин В.И. Основы защиты промышленной и коммерческой информации. Термины и определения (словарь). ИПКИР. М., 1994 г.
87. Хаусли Т. Системы передачи и телеобработки данных. Пер. с англ. -М.: Радио и связь, 1994.- 456 с.
88. Холланд Дж. Х. Генетические алгоритмы// В мире науки,1992.- №9-10.- С. 32-40
89. Хофман Л.Дж. Современные методы защиты информации. М.: Советское радио, 1980.- 264 с.
90. Цыгичко В.Н., Черешкин Д.С., Смолян Г.Л. Новости информационной борьбы// Конфидент, 1996.- № 6.- С. 19-21.
91. Шеннон К. Теория связи в секретных системах// В сборнике “Работы по теории информации и кибернентике”.- М.: ИЛ, 1963.- С. 333-402.
92. Штарьков Ю.М. Некоторые теоретико-информационные задачи защиты дискретных данных// Проблемы передачи информации, - 1994.- т. 30, № 2.- С.49-60.
93. Шураков В.В. Обеспечение сохранности информации в системах обработки данных.- М.: Финансы и статистика, 1985.- 224 с.
94. // ECAI 94. 11th European Conference on Artificial Intelligence. Edited by A. Cohn. Published in 1994 by John Wiley & Sons, Ltd.
95. // ECAI 94. 11th European Conference on Artificial Intelligence. Edited by A. Cohn. Published in 1994 by John Wiley & Sons, Ltd.
96. A. Giordana, Saitta L., Zini F. Learning Disjunctive Concept Definitions Using a Genetic Algorithm// ECAI 94. 11th European Conference on Artificial Intelligence. Edited by A. Cohn. Published in 1994 by John Wiley & Sons, Ltd., p. 483-486.
97. A. Schneider An adaptive system for generating neural networks with genetic algorithms// SPIE Vol. 2492, p. 284-291.
98. Abrams M., Jeny F. Network security; Protocol reference models and the trusted computer system evaluation criteria// IEEE network magazine. April 1987, v. 1, p. 24-33.
99. Delahaye D., Alliot J.-M. Genetic Algorithms for Air Traffic Assignment// ECAI 94. 11th European Conference on Artificial Intelligence. Edited by A. Cohn. Published in 1994 by John Wiley & Sons, Ltd., p. 33-37.
100. Goldberg D. Genetic Algorithms, Addison Wesley, 1989.
101. J.K. Hao, R. Dorne A New Population-Based Method for Satisfiability Problems// ECAI 94. 11th European Conference on Artificial Intelligence. Edited by A. Cohn. Published in 1994 by John Wiley & Sons, Ltd., p. 135-139.
102. Malheiro B., Jennings N.R., Oliveira E. Belief Revision in Multi-Agent Systems // ECAI 94. 11th European Conference on Artificial Intelligence. Edited by A. Cohn. Published in 1994 by John Wiley & Sons, Ltd., p. 294-298.

103. Michalewicz Z. Genetic algorithms + data structures = evolving programs, Springer-Verlag, 1992.
104. Shirey R.W. Defense date network security architecture// Comput. Commun. Rev., 1990, vol. 20.- № 2, p. 66-71.
105. Y. Lin.Neurons, Psychons, and Emotion// SPIE Vol. 2492, p. 184-191.
106. Ефремов А. Сетевые атаки и средства борьбы с ними // Computer Weekly № 14, 1998г. с. 14-17.