Язов Ю. К.
Соловьев С. В.

Методология оценки эффективности защиты информации в информационных системах от несанкционированного доступа

Монография

Ю. К. Язов, С. В. Соловьев

МЕТОДОЛОГИЯ ОЦЕНКИ ЭФФЕКТИВНОСТИ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА

Монография

Санкт-Петербург
Наукоемкие технологии
2023

УДК 004.056.53
ББК 32.973
Я40

Рецензенты:
доктор технических наук, доцент С. А. Будников;
доктор технических наук, профессор В. Б. Авдеев

Язов Ю. К., Соловьев С. В.
Я40 Методология оценки эффективности защиты информации в информационных системах от несанкционированного доступа: монография. – Санкт-Петербург: Наукоемкие технологии, 2023. – 258 с.

ISBN 978-5-907618-36-7

В монографии раскрываются проблемные вопросы анализа угроз безопасности информации в информационных системах, показывается необходимость моделирования процессов реализации угроз, дается общая характеристика и классификация моделей оценки эффективности технической защиты информации от них. Акцентируется внимание на состоянии и перспективах применения теории риска для оценки эффективности защиты информации в информационных системах. Раскрываются подходы к оценке возможного ущерба на основе балльного метода, теории нечетких множеств, построения шкал оценок с использованием парадигмы предельного ущерба. Описываются подходы к моделированию процессов реализации угроз в условиях отсутствия и применения мер защиты с использованием формальных моделей безопасности, логико-лингвистических моделей, теории графов, предикатных сетей, сетей Петри, аппарата марковских, полумарковских процессов, составных сетей Петри-Маркова. Показывается важность учета фактора времени для корректной оценки возможностей реализации угроз, количественной оценки рисков такой реализации и эффективности технической защиты информации от несанкционированного доступа на сетевом, системном и прикладном системно-технических уровнях. Раскрываются методологические аспекты формирования количественных требований к эффективности технической защиты информации в информационных системах, а также требований к эффективности средств защиты.

Монография предназначена для научных работников, занимающихся исследованиями в области технической защиты информации в информационных системах.

УДК 004.056.53
ББК 32.973

ISBN 978-5-907618-36-7
© Язов Ю. К., Соловьев С. В., 2023

ОГЛАВЛЕНИЕ

СПИСОК СОКРАЩЕНИЙ 6

ВВЕДЕНИЕ 7

1. ПРОБЛЕМНЫЕ ВОПРОСЫ ОЦЕНКИ ЭФФЕКТИВНОСТИ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ 9

1.1. Онтологические аспекты проблематики оценки эффективности защиты информации от несанкционированного доступа 9

1.2. Основные факторы, подлежащие учету при оценке эффективности защиты информации 16

1.3. Проблемные вопросы анализа угроз безопасности информации и пути их решения 24

1.4. Методологические основы типизации информационных систем 28

2. ОБЩАЯ ХАРАКТЕРИСТИКА МОДЕЛЕЙ ОЦЕНКИ ЭФФЕКТИВНОСТИ ТЕХНИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ ... 31

2.1. Классификация моделей оценки эффективности защиты информации от несанкционированного доступа 31

2.2. Модель оценки эффективности технической защиты информации от несанкционированного доступа с применением функционального подхода 33

2.3. Модель оценки эффективности технической защиты информации на основе оценочных уровней доверия 36

2.4. Модели оценки эффективности технической защиты информации от несанкционированного доступа с использованием теории риска 44

2.4.1. Общий подход к использованию теории риска при оценке эффективности технической защиты информации от несанкционированного доступа 44

2.4.2. Парирование сложностей, связанных с неопределенностью исходных данных и большой размерностью задачи оценки эффективности защиты информации 48

2.5. Балльный метод оценки эффективности защиты информации 56

3. МЕТОДОЛОГИЯ ОЦЕНКИ УЩЕРБОВ ОТ РЕАЛИЗАЦИИ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ 61

3.1. Классификация ущербов от реализации угроз безопасности информации 61

3.2. Некоторые аспекты методологии количественной оценки финансового ущерба от реализации угроз безопасности информации 64

3.3. Шкалы оценок ущерба, парадигма предельного ущерба и универсальная шкала оценок ущербов 68

3.3.1. Построение универсальной шкалы оценок ущерба для одной информационной системы 68

3.3.2. Построение универсальной шкалы оценок ущерба для совокупности информационных систем 73

3.4. Аппарат нечетких множеств и его применение для оценки ущерба от реализации угроз безопасности информации 75

4. МОДЕЛИ ПРОЦЕССОВ РЕАЛИЗАЦИИ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ С ДИСКРЕТИЗАЦИЕЙ СОСТОЯНИЙ БЕЗ УЧЕТА ФАКТОРА ВРЕМЕНИ 84

4.1. Классификация моделей процессов реализации угроз безопасности информации от несанкционированного доступа 84

4.2. Формальные модели безопасности и их применение при оценке возможностей реализации угроз безопасности информации 88

4.3. Логико-лингвистические модели процессов реализации угроз безопасности информации 95

4.4. Логические сети и их применение при моделировании процессов реализации угроз безопасности информации 106

4.5. Сети Петри и их применение при оценке возможностей реализации угроз безопасности информации 117

4.6. Метод анализа возможностей реализации угроз на основе теории графов 123

5. МОДЕЛИ ДИНАМИКИ РЕАЛИЗАЦИИ УГРОЗ С ИСПОЛЬЗОВАНИЕМ АППАРАТА МАРКОВСКИХ ПРОЦЕССОВ 131

5.1. Марковские цепи и их применение при моделировании динамики реализации угроз безопасности информации в информационных системах 131

5.2. Введение в теорию дискретных марковских процессов с непрерывным временем 136

5.3. Элементы теории случайных потоков, существенные для моделирования процессов в предметной области защиты информации 139

5.4. Некоторые модели динамики возникновения угроз, основанные на теории потоков 149

5.5. Общий подход к построению марковских моделей процессов возникновения и реализации угроз безопасности информации с дискретными состояниями и непрерывным временем 155

5.6. Некоторые марковские модели процессов реализации угроз безопасности информации 160

5.6.1. Марковские модели динамики выполнения несанкционированных действий относительно блоков защищаемой информации в отдельном компьютере 160

5.6.2. Марковские модели динамики проникновения в операционную среду компьютера и выполнения несанкционированного действия 167

5.6.3. Марковские модели динамики реализации некоторых сетевых атак 173

5.6.4. Марковские модели динамики инфицирования вредоносными программами информационных систем 183

5.7. Модели динамики реализации угроз с использованием аппарата полумарковских процессов 190

6. МАТЕМАТИЧЕСКИЕ МОДЕЛИ ДИНАМИКИ РЕАЛИЗАЦИИ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ НА ОСНОВЕ АППАРАТА СЕТЕЙ ПЕТРИ-МАРКОВА 196

6.1. Традиционные сети Петри-Маркова и особенности их применения при моделировании динамики реализации угроз безопасности информации в информационных системах 196

6.2. Составные сети Петри-Маркова на основе марковских процессов с пропозициональной логикой срабатывания логических переходов 202

6.3. Составные сети Петри-Маркова на основе марковских процессов с предикатными условиями выполнения 212
6.4. Примеры применения аппарата составных сетей Петри-Маркова на основе марковских процессов для моделирования угроз безопасности информации 214
6.5. Составные сети Петри-Маркова на основе полумарковских процессов и примеры их применения для моделирования процессов реализации угроз безопасности информации 219

7. МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ ФОРМИРОВАНИЯ ТРЕБОВАНИЙ К ЭФФЕКТИВНОСТИ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ 231
7.1. Особенности формирования требований к эффективности защиты информации в информационных системах 231
7.2. Формирование требований к эффективности защиты информационной системы и ее элементов от угроз отказа в обслуживании на основе их сопоставления с требованиями по надежности функционирования информационных систем 234
7.3. Формирование требований к эффективности защиты конфиденциальной информации от угроз, реализуемых путем несанкционированного доступа к ней 242
7.4. Некоторые аспекты обоснования количественных требований к средствам защиты информации в информационных системах 245

ЗАКЛЮЧЕНИЕ 249

ЛИТЕРАТУРА 250

СПИСОК СОКРАЩЕНИЙ

АРМ – автоматизированное рабочее место

ЗИ – защита информации

ИС – информационная система

ICMP – Internet Control Message Protocol – протокол межсетевых управляющих сообщений

IP – Internet Protocol – протокол сетевой адресации

ЛВС – локальная вычислительная сеть

НСД – несанкционированный доступ

ОС – операционная система

ОУД – оценочный уровень доверия

ПМП – полумарковский процесс

ПО – программное обеспечение

СЦ – синтагматическая цепь

СПМ – сеть Петри-Маркова

СС – семантическая сеть

ССПМ – составная сеть Петри-Маркова

СУБД – система управления базами данных

ТЗИ – техническая защита информации

ФП – функция принадлежности

ТМД – типизированная матрица доступа

TCP – Transfer Control Protocol – протокол передачи данных

FTP – File Transfer Protocol – протокол передачи файлов

ЯСУ – язык ситуационного управления

ВВЕДЕНИЕ

Защита информации (ЗИ) в информационных системах (ИС), широко применяемых в органах государственной власти, на предприятиях и в организациях кредитно-финансовой, топливно-энергетической, транспортной и многих других сфер деятельности государства и общества, сегодня становится одной из приоритетных государственных задач. Это обусловлено постоянно расширяющимся спектром угроз безопасности информации, реализация которых приводит к нарушениям, следствием которых могут быть не только финансовые, экономические потери, срывы в выполнении государственных заданий, снижение деловой репутации предприятий, негативные социальные последствия, экологические катастрофы, но и гибель людей. В этих условиях Правительством Российской Федерации, Советом безопасности Российской Федерации прилагаются активные усилия по парированию указанных угроз, созданию и развертыванию отечественных ИС в защищенном исполнении. Это достигается не только нормативным регулированием со стороны государства деятельности органов, предприятий и организаций в области ЗИ, но и созданием отечественной компонентной базы, программного обеспечения (ПО), отечественных средств и систем защиты информации для ИС.

Вместе с тем обеспечение эффективной защиты информации в ИС, обоснование требований к такой защите, изыскание рациональных путей построения систем защиты связано с необходимостью применения соответствующего методического обеспечения. Сегодня такое методическое обеспечение сводится к системе вербально сформулированных в нормативно-правовых актах уполномоченных федеральных органов исполнительной власти требований, касающихся состава обязательных к применению мер защиты информации для определенного класса (уровня) защищенности ИС и действий по обеспечению ЗИ в части порядка формирования требований к ЗИ и разработки системы защиты информации в ИС, внедрения системы защиты информации, аттестации ИС по требованиям ЗИ, обеспечения ЗИ в ходе эксплуатации аттестованной ИС и при выводе ее из эксплуатации. Такие требования и действия определены на основе обобщения опыта специалистов. При этом какие-либо количественные процедуры их обоснования отсутствуют. Следствием такого подхода является то, что альтернативные варианты построения систем защиты информации, возможно более эффективные, чем предлагаемые к исполнению в нормативных правовых актах, вообще не рассматриваются. Все это обусловливает необходимость существенного совершенствования методического обеспечения защиты информации в ИС.

Данная монография направлена на устранение в определенной мере этого «пробела» в части, касающейся развития методологии оценки эффективности технической защиты информации (ТЗИ) от несанкционированного доступа (НСД), то есть ЗИ некриптографическими

методами. При этом акцент сделан, во-первых, на ТЗИ в ИС от НСД и, во-вторых, – на ТЗИ в так называемых «офисных» ИС, то есть, как названы такие системы, например, в [1], в информационно-организационных системах (расчетных, логико-аналитических, информационно-справочных системах, системах электронного документооборота и т.п.), а также в некоторых классах автоматизированных систем управления технологическими процессами [2–4].

Следует подчеркнуть, что проблема развития методологии оценки эффективности защиты информации в ИС охватывает очень широкий спектр вопросов и в одной монографии, конечно, невозможно раскрыть все, что касается их решения. В частности, в ней не рассматриваются методологические вопросы, связанные, например, с оценкой эффективности новых технологий защиты информации и построения систем защиты, таких как SIEM-системы, SOC-системы, многоагентные системы защиты и др., поскольку даже краткое изложение материала по ним представляет собой самостоятельный труд. Следует заметить, что в монографии не раскрываются также вопросы оценки эффективности ТЗИ в ИС, построенных с использованием новых технологий обработки информации, например, суперкомпьютерных, облачных технологий, защиты от угроз безопасности информации на микропрограммном уровне и вопросы эффективности применения таких стратегий защиты, как стратегия «обмана» (введения в заблуждение), стратегия адаптивной защиты и тем более стратегия тотального контроля операций, выполняемых операционной системой (ОС), выявления и оперативного их пресечения. Это обусловлено тем, что эти вопросы в методологическом плане только начинают прорабатываться как в России, так и за рубежом.

Значительное внимание в монографии уделено развитию методологии оценки возможностей реализации и опасности угроз, а также учету фактора времени при оценке эффективности защиты информации в ИС. Кроме того, в ней впервые раскрываются некоторые подходы к количественному обоснованию требований к эффективности ТЗИ в ИС, указываются те аспекты, которые целесообразно учитывать при таком обосновании.

Книга предназначена для научных работников, занимающихся исследованиями в области ТЗИ в ИС, для аспирантов, студентов и слушателей, обучающихся по данной тематике, а также для специалистов, интересующихся вопросами развития методологии оценки эффективности ТЗИ от НСД.

1. ПРОБЛЕМНЫЕ ВОПРОСЫ ОЦЕНКИ ЭФФЕКТИВНОСТИ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

1.1. Онтологические аспекты проблематики оценки эффективности защиты информации от несанкционированного доступа

Необходимость рассмотрения онтологических аспектов защиты информации в ИС обусловлена рядом обстоятельств, к основным из которых относятся:

во-первых, отсутствие определений ряда применяемых на практике терминов и характеристик взаимосвязей этих терминов в действующих терминологических документах и правовых актах;

во-вторых, разные трактовки или неоднозначность трактовок некоторых терминов и их определений в различных публикациях по тематике защиты информации.

Конечно, в данной монографии невозможно охватить все используемые в данной области термины и определения, однако некоторые из них для материала данной монографии нуждаются в пояснении. К таким терминам относятся, например, защита информации, обеспечение безопасности информации, информационная безопасность и безопасность информации, несанкционированный доступ, эффективность защиты и защищенность информации, политика безопасности информации (или политика информационной безопасности). Ниже рассматривается авторская трактовка определений указанных терминов и их взаимосвязей.

В соответствии с [5] защита информации представляет собой «принятие правовых, организационных и технических мер, направленных:

1) на обеспечение защиты информации от неправомерного доступа, уничтожения, модифицирования, блокирования, копирования, предоставления, распространения, а также от иных неправомерных действий в отношении такой информации;

2) на соблюдение конфиденциальности информации ограниченного доступа;

3) на реализацию права на доступ к информации».

Следует отметить, что «обеспечение защиты информации» здесь рассматривается в узком смысле как совокупность действий по реализации правовых, организационных и технических мер непосредственно по защите информации. В широком смысле обеспечение защиты информации включает в себя финансовое, кадровое, техническое, программное и другие виды обеспечения, которые влияют как непосредственно, так и опосредованно на защиту информации. В [6] дается несколько иное определение защиты информации как «деятельности, направленной на предотвращение утечки защищаемой информации, несанкционированных и непреднамеренных воздействий на защищаемую информацию». При этом защита информации

включает в себя правовую, техническую, криптографическую и физическую защиты, а также защиту от разглашения (рисунок 1.1).

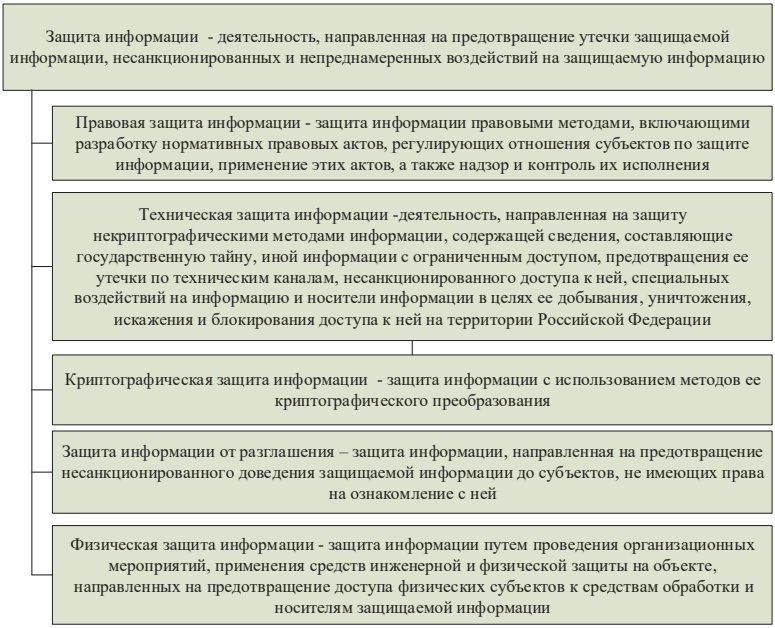


Рисунок 1.1 – Структура понятия «защита информации» [5]

Как было отмечено ранее, в данной работе рассматривается только ТЗИ. В соответствии с определением (см. рисунок 1.1) ТЗИ реализуется некриптографическими методами. Появление данного понятия обусловлено разделением сфер ответственности государственных регуляторов в этой области – ФСБ России (отвечает за криптографическую защиту информации) и ФСТЭК России (отвечает за ТЗИ).

Рассматривая понятие «обеспечение безопасности информации», необходимо подчеркнуть, что здесь так же, как и в предыдущем случае, «обеспечение» рассматривается в узком смысле. Однако здесь это слово привязано к иному понятию – «безопасности информации», под которым в соответствии с [7] понимается «состояние защищенности информации, при котором обеспечены ее конфиденциальность, целостность и доступность».

Таким образом, понятие безопасности информации определено через другое понятие – «защищенности информации», широко применяемое во многих отечественных документах.

Вместе с тем понятие защищенности информации в законах, действующих терминологических документах, нормативных правовых актах органов исполнительной власти сегодня отсутствует.

На наш взгляд, защищенность информации следует понимать как соответствие состава принятых мер по защите информации составу мер, установленному требованиями нормативного документа, в соответствии с которым оценивается эта защищенность.

В связи с этим ИС может быть отнесена к защищенной, если в ней выполняются требования нормативных документов уполномоченных органов исполнительной власти, требования государственных, международных стандартов или стандартов организаций в области защиты информации и информационных технологий, если таковые имеются. Таким образом, понятие защищенности связывается с определенным нормативным документом или некоторым набором таких документов, выполнение требований которых является основанием считать, что информация в ИС защищена в смысле выполнения этих требований. Необходимость учета нескольких нормативных документов одновременно обусловлена, например, тем, что одна и та же ИС может относиться к разным системам (например, к государственной ИС и одновременно к информационной системе персональных данных), требования к которым регламентируются разными нормативными документами, при этом она как система персональных данных может оказаться вполне защищенной, а как государственная информационная система – нет.

Необходимо подчеркнуть, что сегодня для обеспечения дифференцированного подхода к защите информации в отечественных нормативных документах [8–13] введены классы или уровни защищенности ИС, для каждого из которых установлен определенный набор требований. В связи с этим защищенность информации всегда рассматривается в связке с классом защищенности ИС (или уровнем защищенности персональных данных для информационных систем персональных данных). В этом смысле, например, в соответствии с [13] информация в государственной ИС может оказаться защищенной по классу К3, но незащищенной по классу К2.

Кроме того, не все действующие нормативные документы «привязаны» к содержанию парируемых угроз безопасности информации, что приводит к необходимости дополнительного анализа возможностей такого парирования при реализации тех или иных требований по ТЗИ. Без такого анализа зачастую весьма сложно утверждать, достаточен ли набор требований, взятых из выбранных нормативных документов, для парирования актуальных угроз в конкретной ИС, и, следовательно, считать ИС защищенной. Это подчеркивает тот факт, что понятие защищенности ИС является относительным.

С учетом изложенного безопасность информации достигается тогда, когда выполняются требования, установленные нормативным документом для рассматриваемого класса защищенности, или выполняются более жесткие требования по сравнению с установленными. В этом смысле уровень безопасности информации также определяется в зависимости от установленного для ИС класса защищенности.

Необходимость раскрытия соотношения понятий «безопасность информации» и «информационная безопасность» обусловлена следующим. В

современной Доктрине информационной безопасности применяется термин «информационная безопасность Российской Федерации», под которой понимается «состояние защищенности личности общества и государства от внешних и внутренних информационных угроз, при котором обеспечивается реализация конституционных прав и свобод человека и гражданина, достойные качество и уровень жизни граждан, суверенитет, территориальная целостность и устойчивое социально-экономическое развитие Российской Федерации, оборона и безопасность государства» [14]. Аналогичным образом в [15] трактуется понятие «информационная безопасность организации», под которой понимается «состояние защищенности интересов (целей) организации в условиях угроз в информационной сфере», при этом «информационная сфера – это совокупность информации, информационной инфраструктуры, субъектов, осуществляющих сбор, формирование, распространение, хранение и использование информации, а также системы регулирования возникающих при этом отношений». Из изложенного следует, что если безопасность информации касается только самой информации, то информационная безопасность – это значительно более широкое понятие, охватывающее интересы людей, организаций (предприятий) и государства в информационной сфере. Кроме защиты самой информации в данном случае должна обеспечиваться «защита от информации» (от навязывания ложной, тенденциозной, социально вредной и иной неприемлемой для личности, общества и государства информации), защита прав на информированность и др. В данной работе рассматриваются только вопросы, связанные с понятием «безопасность информации».

Одним из основополагающих понятий в области защиты информации является понятие «несанкционированный доступ». В соответствии с [6, 7] под несанкционированным доступом (НСД) понимается «доступ к информации или к ресурсам автоматизированной информационной системы, осуществляемый с нарушением установленных прав и (или) правил доступа».

Такие права и (или) правила устанавливаются обладателем[1] информации для пользователей ИС, для процессов обработки информации, обслуживания ИС, изменения программных, аппаратных (технических) и информационных ресурсов, при этом НСД может быть осуществлен преднамеренно или непреднамеренно.

Поскольку по указанному определению к НСД относится и доступ к аппаратным ресурсам ИС, то следует различать физический НСД, то есть несанкционированный физический доступ к аппаратным ресурсам ИС[2], и виртуальный НСД – доступ к командам, процедурам и функциям операционной среды элементов ИС и с их использованием или путем запуска специально созданных программ, скриптов, команд – к пользовательской

[1] Обладатель информации – лицо, самостоятельно создавшее информацию либо получившее на основании закона или договора право разрешать или ограничивать доступ к информации, определяемой по каким-либо признакам [5].

[2] К аппаратным ресурсам относятся серверы, рабочие станции, аппаратные средства ввода/ вывода информации, коммуникационное оборудование и т.п.

информации (данным или прикладным программам). Необходимо подчеркнуть, что в этой связи к НСД можно относить также:

перехват информации, передаваемой по телекоммуникационной среде, так как такой перехват связан с установкой в узлы телекоммуникационных сетей программных средств перехвата и с запуском в операционной среде этих узлов таких средств перехвата;

реализацию сетевых атак типа «Отказ в обслуживании», направленных на нарушение функционирования ИС, так как при этом нарушаются штатные условия выполнения команд операционной среды, происходит, например, зациклирование вызова этих команд из-за переполнения буферов и т.д.

Виртуальный НСД к информации является состоявшимся, если нарушитель или инициированный им процесс получает возможность выполнения действия (путем вызова команд операционной системы, запуска специально сформированных программ, скриптов и др.) относительно блока защищаемой информации, то есть ее копирования, уничтожения, модификации и т.д.

Следует отметить, что виртуальный НСД может быть непосредственным или удаленным. Непосредственный виртуальный НСД реализуется после получения физического доступа к аппаратным ресурсам ИС путем ввода команд операционной среды с использованием периферийных устройств (например, клавиатуры), а удаленный виртуальный НСД – по информационно-телекоммуникационной среде (сети) с использованием протоколов сетевого (межсетевого) взаимодействия.

Наконец, рассматривая понятие эффективности защиты информации, необходимо отметить, что этот термин достаточно часто встречается в организационно-распорядительных и нормативных документах. (например, «эффективность мер по обеспечению безопасности персональных данных» [16], «эффективность комплекса мер по обеспечению информационной безопасности» [14], «эффективность реализованных в рамках системы защиты мер...» [10], «эффективность использования мер и средств обеспечения безопасности [17], «эффективность требований по защите», «эффективность компонентов защиты», «эффективность механизма защиты» [18] и т.д.).

Более того, например, в [16] даже регламентируется процедура оценки эффективности защиты, где указано, что «оценка эффективности реализованных в рамках системы защиты персональных данных мер по обеспечению безопасности персональных данных проводится оператором самостоятельно или с привлечением на договорной основе юридических лиц и индивидуальных предпринимателей, имеющих лицензию по технической защите конфиденциальной информации».

В соответствии с [6] эффективность защиты информации определяется как «степень соответствия результатов защиты информации поставленной цели». Иногда, наряду с «функциональной» эффективностью, которая фактически имеется в виду в данном определении, в этом понятии учитывают и затраты на применение мер и средств ТЗИ. Так, в [19] «эффективность

обеспечения информационной безопасности» понимается как связь между достигнутым результатом и использованными ресурсами для обеспечения заданного уровня информационной безопасности». В данной монографии эффективность защиты информации понимается в соответствии с [6].

Иногда, понятие «эффективность» подменяют понятием «достаточность», имея в виду, что мера или средство ТЗИ не обладает никакой эффективностью, если не удовлетворяет принятому критерию достаточности. Следует подчеркнуть, что оценка достаточности является бинарной: либо она обеспечивается (показатель оценки равен, например, 1), либо не обеспечивается (показатель равен 0). Достаточность не характеризует степень приближения к заданному уровню защищенности и это понятие отличается от понятия эффективности, характеризующей меру приближения уровня защищенности информации к уровню, определенному в соответствии с установленными требованиями. Оценка эффективности позволяет на основе количественной меры установить, насколько далека защищенность информации в ИС от той, которая соответствует заданному классу защищенности ИС или в более общем представлении – поставленной цели защиты (одной из таких целей может быть выполнение требований, соответствующих заданному в нормативном документе классу защищенности или уровню защищенности персональных данных).

Однако цели ТЗИ, которые ставятся при организации защиты информации в ИС, могут быть существенно разными. Например, цель может состоять в устранении всех известных уязвимостей, в достижении максимально возможной защищенности информации имеющимися средствами, в парировании всех известных угроз безопасности информации, актуальных для данной ИС, в обеспечении устойчивости функционирования ИС в целом или даже в виде необходимости соблюдения некоторых интересов пользователей (предприятия, организации или совокупности предприятий и организаций, эксплуатирующих ИС), например, в недопущении снижения прибыли предприятия ниже установленного порога из-за нарушений безопасности информации, в исключении возможности неприемлемого ущерба от реализации угроз безопасности информации и др. Зачастую такие цели могут быть иерархически упорядоченными (рисунок 1.2). Тогда оценка эффективности по нижестоящей цели выступает как составляющая часть оценки эффективности в соответствии с вышестоящей целью.

Если необходимо оценивать эффективность применения того или иного средства или меры защиты информации, то также имеет место неопределенность данного понятия. Во-первых, эффективность может оцениваться относительно заранее определенной и декларированной цели защиты для данного средства (меры). Во-вторых, оценка может проводиться относительно некоторой общей цели для определенного направления ТЗИ, например, для защиты информации от НСД, защиты информации от несанкционированной передачи во внешние сети и т.д.

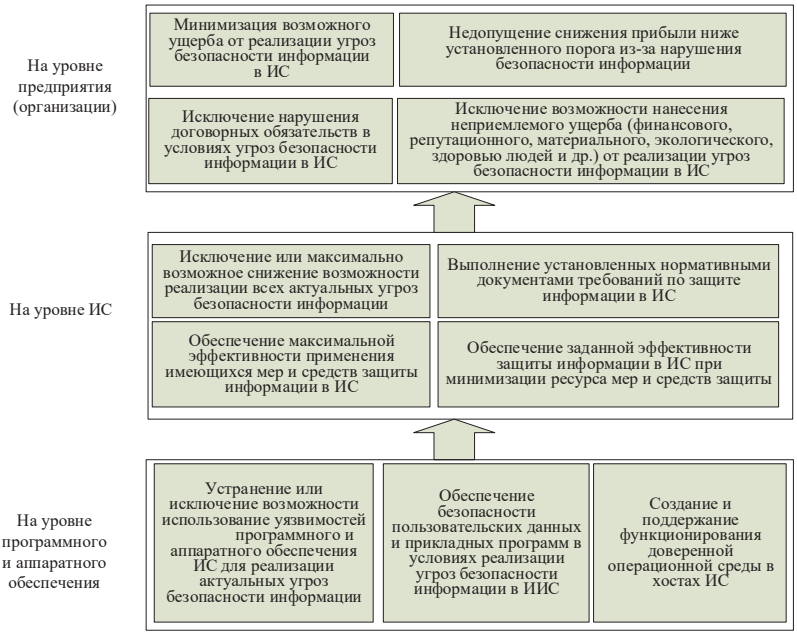


Рисунок 1.2 – Иерархия возможных целей защиты информации в организации (на предприятии)

Чаще всего рассматривается некоторая общая цель защиты, которая формулируется для данной ИС в целом, что создает предпосылки для оценивания вклада каждого средства защиты, применяемого в данной ИС, в результирующую эффективность защиты информации в ней.

В связи с изложенным оценка эффективности защиты оказывается условной или относительной, то есть зависящей от того, каким образом сформулирована цель ТЗИ в конкретной ИС. Это обстоятельство во многом обусловливает тот факт, что сегодня при выборе мер и средств защиты информации, построении систем защиты количественная оценки эффективности ТЗИ, несмотря на востребованность, пока практически не проводится.

Наконец, необходимо остановиться на понятии «политика безопасности информации», которое сегодня весьма широко применяется на практике. В соответствии с [15, 20] политика безопасности информации организации – это совокупность документированных правил, процедур, практических приемов или руководящих принципов в области безопасности информации, которыми руководствуется организация в своей деятельности. Это понятие пришло в Россию из международных стандартов и широко используется во многих национальных стандартах, являющихся аутентичными переводами международных стандартов в области обеспечения безопасности информации.

Сложности с применением этого понятия возникают тогда, когда нужно формировать некий документ с изложением содержания политики безопасности информации в организации (предприятии), поскольку строгой регламентации содержания понятия «политики безопасности информации», по сути, нет. Во многом это содержание пересекается с содержанием концепции защиты и формируется в виде концентрированного выражения правил, приемов, процедур, принципов, которыми руководствуется организация при создании системы защиты информации и которые прямо или косвенно учитываются в такой концепции.

На наш взгляд, изложенные пояснения по рассмотренным терминам и их определениям будут способствовать пониманию излагаемых в монографии подходов к оценке эффективности ТЗИ, помогут исключить их некорректную трактовку и применение в практике оценки эффективности ТЗИ.

1.2. Основные факторы, подлежащие учету при оценке эффективности защиты информации

Оценка эффективности ТЗИ чаще всего проводится на основе сравнения защищенности информации без применения и с применением мер защиты. Кроме того, оценка защищенности информации необходима:

– при выявлении угроз безопасности информации, оценке их актуальности, возможности предупреждения, обнаружения и нейтрализации угроз, при выявлении уязвимостей программного и программно-аппаратного обеспечения ИС;

– при принятии решений на проведение мероприятий по ТЗИ;

– при обосновании требований по защите информации к средствам вычислительной техники и ИС в защищенном исполнении

– при обосновании требований к средствам, комплексам и системам ТЗИ;

– при сравнении между собой и выборе существующих способов, мер и средств ТЗИ в ИС, изыскании и разработке новых способов ТЗИ, оптимизации распределения ресурса мер и средств защиты информации в ИС;

– при проверке соответствия ИС установленным требованиям и аттестации ИС, при решении задач сертификации средств и комплексов защиты информации и т.д.

При этом имеют место множество факторов, влияющих в той или иной мере на результаты оценки, основные из которых приведены на рисунке 1.3.

Суть влияния этих факторов на результаты оценки защищенности информации и эффективности ТЗИ сводится к следующему.

Фактор относительности понятия защищенности информации и эффективности защиты (см. раздел 1.1) обусловливает то, что результаты оценки защищенности информации должны быть «привязаны» к определенному документу или совокупности документов, на соответствие требованиям которых оценивается защищенность информации в ИС, к

установленному для этой ИС классу (уровню) защищенности, а результаты оценки эффективности ТЗИ – к достигаемой цели защиты информации.

Рисунок 1.3 – Основные факторы, влияющие на результаты оценки защищенности информации в ИС и эффективности ТЗИ

При различии целей защиты информации даже в весьма незначительно отличающихся друг от друга ИС (по назначению, составу аппаратного и программного обеспечения и т.д.) результаты оценки эффективности ТЗИ в них оказываются несопоставимыми, что весьма затрудняет регулирование деятельности организаций (предприятий) в этой области со стороны уполномоченных органов исполнительной власти.

Во многом именно из-за этого указанное регулирование пока ограничивается только проверкой защищенности информации на соответствие требованиям нормативных документов и не затрагивают вопросы оценки эффективности ТЗИ.

Факторы неопределенности сведений о возможностях источника угрозы касаются оснащенности, компетенций и мотиваций нарушителя, реализующего угрозы безопасности информации [6, 13, 21], а также типа и функций применяемых вредоносных программ, которыми могут быть инфицированы ИС. Учет этих факторов обусловливает два варианта проведения оценки:

1) применительно к наилучшей оснащенности, наивысшим компетенциям нарушителей и наиболее опасным их мотивациям, а также ко всем возможным функциям вредоносных программ (в том числе программных или программно-аппаратных закладок);

2) с градацией компетенций и мотиваций нарушителя, градацией функциональных возможностей вредоносных программ, что сегодня фактически отсутствует.

Поэтому сегодня применяется только первый из указанных вариантов оценки.

Факторы неопределенности, случайной направленности и содержания несанкционированных действий при реализации угроз безопасности информации обусловлены тем, что, во-первых, несанкционированные (в том числе деструктивные) действия (копирование, уничтожение, перемещение, модификация и др.) при реализации угрозы могут быть направлены на разные блоки защищаемой информации (файлы, каталоги, директории), при этом не известно, какой из блоков представляет первоочередной интерес для нарушителя, будут ли такие действия осуществляться относительно всех блоков защищаемой информации и в какой последовательности и т.д. Во-вторых, сами несанкционированные действия или их совокупность (например, копирование и уничтожение, модификация и перемещение) относительного защищаемого блока информации при оценке неизвестны, поскольку могут выбираться нарушителем в зависимости от цели реализации угрозы заблаговременно. Для парирования такой неопределенности и случайности может быть использовано несколько подходов:

1) ранжирование угроз по вероятности выполнения возможных несанкционированных действий относительно каждого блока защищаемой информации в ИС. Обоснование возможности применения такого подхода связано с набором соответствующей статистики реализации угроз в различных ИС;

2) использование нечетких оценок (с применением аппарата нечетких множеств) возможного состава несанкционированных действий относительно каждого блока защищаемой информации в ИС. Такой подход вполне приемлем, но в результате его применения оценка становится нечеткой, зависящей от установленных для каждого действия функций принадлежности [22–25], что приводит к необходимости или интерпретации нечетких выводов, или корректной дефаззификации (перехода от нечетких к четким показателям оценки);

3) ранжирование угроз по опасности выполнения возможных несанкционированных действий относительно каждого блока защищаемой информации в ИС и выбора наиболее опасных угроз для рассматриваемых блоков (или блока) информации. Данный подход связан с необходимостью оценки и сравнения ущерба от реализации угроз относительно разных блоков защищаемой информации, однако, во-первых, методики количественной оценки прогнозируемого ущерба от копирования, уничтожения, блокирования и т.д. какой-либо информации в ИС сегодня отсутствуют, во-вторых, ущербы могут быть разнородными, что весьма существенно затрудняет их сравнение (см. раздел 3);

4) разделение всего множества угроз безопасности информации, которые могут иметь место в ИС, на следующие группы:

а) угрозы, связанные с проникновением в операционную среду хоста ИС и к файловой системе, то есть к командам (операциям, функциям) операционной системы с последующим их выполнением относительно файлов

системной или пользовательской информации (в том числе файлов баз данных, вплоть до полей записей и самих записей) [26]. После такого проникновения могут выполняться любые действия с файлами – уничтожение, модификация, копирование, запись на съемные носители информации, несанкционированная передача во внешние сети, повышение привилегий доступа путем изменения учетных записей, подбора паролей доступа или изменения записей в системном реестре, нарушения доступности файлов путем модификации дескрипторных таблиц или записей в реестре, создание «черных входов», инфицирование вредоносными программами, в том числе для создания скрытых каналов передачи данных и т.д.;

б) угрозы, связанные с перехватом пакетов информации при их передаче по сетям общего пользования. Такие угрозы могут реализовываться путем, во-первых, несанкционированного изменения маршрутов (таблиц маршрутизации) передачи пакетов с прохождением их через нужные нарушителю хосты, во-вторых, копирования и передачи пакетов (даже без распаковки) на нужный нарушителю адрес, в-третьих, подмены сетевого адреса («IP-spoofing», «DNS-spoofing»), подмены физического адреса сетевой карты («MAC-spoofing») и др.

Далее для угроз, входящих в группу а), оценивается возможность реализации угроз проникновения и выбирается угроза с наибольшей вероятностью проникновения и наиболее опасное с точки зрения наносимого ущерба обладателю информации несанкционированное действие, которое может быть выполнено в результате проникновения в операционную среду хоста ИС. Такой подход позволяет существенно сократить трудоемкость проводимого анализа возможных угроз.

Аналогично для угроз, входящих в группу б), выбирается угроза с наибольшей вероятностью перехвата сообщений и наибольшим ущербом, который может иметь место в результате перехвата. Тем самым парируется сложность анализа, связанная с огромным разнообразием возможных угроз и неопределенностью выбираемого нарушителем способа реализации угроз перехвата.

Факторы неопределенности и случайности последствий реализации угроз безопасности информации обусловлены не только отмеченными неопределенностями и случайностями состава несанкционированных действий при реализации угроз, но и тем, что, во-первых, разными обладателями одной и той же информации результаты несанкционированных действий относительно нее оцениваются по-разному, поскольку последствия для них, как правило, различны. Например, если для одного обладателя информации уничтожение файла и нарушение нормальной работы компьютера оказывается весьма критичным (срыв сроков выполнения заказа, снижение деловой репутации и т.п.), то для другого уничтожение такого же файла будет сведено лишь к необходимости затраты (возможно даже не существенной) времени на восстановление файла или нормальной работы компьютера. Во-вторых, даже для одного и того же обладателя размеры

ущерба от такого воздействия могут оказаться весьма различными в разные периоды времени. Кроме того, размер наносимого ущерба часто является случайным, то есть непредсказуемой точно величиной. При этом для учета такой случайности могут быть использованы как методы теории вероятностей, так и методы теории нечетких чисел. Для парирования сложностей, связанных с данным фактором, чаще всего рассматриваются только самые опасные несанкционированные действия.

Факторы неопределенности, нечеткости сведений, которые предположительно используют источники угроз при попытке их реализации (об операционной среде, о реализуемых мерах ТЗИ и др.), во-первых, обусловливают случайный характер процесса реализации угроз безопасности информации в ИС для защищающейся стороны, во-вторых, вынуждают моделировать процессы реализации угроз, в том числе, возможно, с учетом вариантов осведомленности атакующего об атакуемой ИС и применяемых в ней мерах ТЗИ, в-третьих, приводят к необходимости использовать для учета таких факторов аппараты теории вероятностей и теории нечетких множеств при оценке защищенности информации и оценке эффективности ТЗИ.

Факторы неопределенности, связанные с типизацией ИС, их программных и аппаратных элементов, обусловлены тем, что часто при оценке защищенности информации и оценке эффективности ТЗИ некоторые сведения об установленном программном и аппаратном обеспечении ИС являются или ориентировочными (например, без указания версии операционной системы или прикладной программы), или обобщенными для класса, к которому относится данная ИС, или отсутствуют вообще. В этом случае проводится типизация ИС (см. раздел 1.4), при которой всегда имеются некоторые характеристики ИС, несколько отличающиеся от характеристик оцениваемой ИС. При правильной типизации отклонения в оценке защищенности и тем более в оценке эффективности ТЗИ, как правило, минимальны, а в ряде случаев и несущественны. Некоторые аспекты методологии типизации ИС рассмотрены в разделе 1.4.

Фактор стратегии защиты информации, реализуемой в ИС, нельзя не учитывать при оценке защищенности информации и оценке эффективности ТЗИ. Это обусловлено следующим.

Защита информации в ИС сегодня может осуществляться с использованием разных стратегий. Классификация известных сегодня стратегий приведена на рисунке 1.4.

Краткая их характеристика сводится к следующему.

Стратегия «периметровой» защиты заключается в том, что информация в ИС защищается по периметру от внешних угроз, а также от угроз несанкционированной передачи ее из ИС в сети общего пользования. Это достигается установкой соответствующих программных и программно-аппаратных средств (типа «межсетевой экран», шлюз и т.п.), контролирующих по определенным правилам входящий и выходящий трафик ИС.

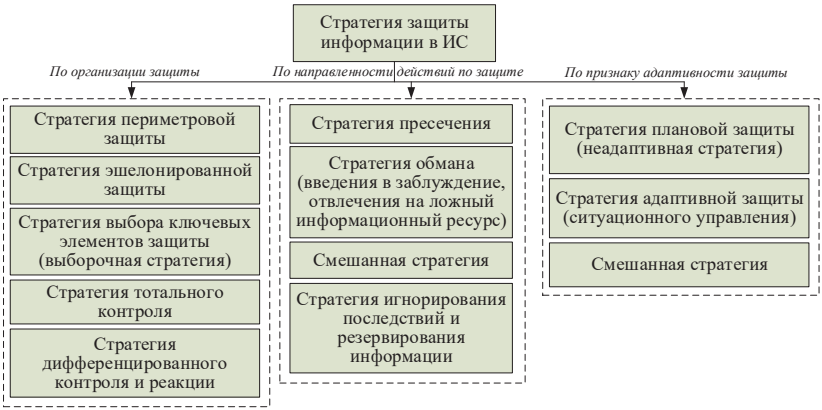


Рисунок 1.4 – Основные стратегии защиты информации в ИС

Стратегия эшелонированной защиты заключается в том, что, наряду с защитой информации в ИС по периметру, принимаются меры защиты на уровне серверов сети, рабочих станций, операционной системы каждого хоста, прикладной программы (например, на уровне системы управления сетевой или локальной базы данных), на уровне архитектуры программно-аппаратной среды каждого компьютера и даже на уровне команд процессора. Состав эшелонов защиты определяется содержанием требований защиты.

Стратегия «ключевых элементов» или выборочная стратегия заключается в том, что защита осуществляется только на наиболее важных элементах (программных или программно-аппаратных) защищаемой ИС. Такая стратегия применяется в случае наличия ограничений на состав применяемых мер и средств защиты.

Стратегия тотального контроля заключается в том, что осуществляется контроль любых действий, операций, команд в хостах ИС. При этом такая стратегия может быть реализована в нескольких вариантах. Первый вариант заключается в том, что осуществляется проверка действий, операций, команд специально выделенным программным (программно-аппаратным) средством – монитором безопасности. Такой контроль может осуществляться или на прикладном уровне (при этом контроль, как правило, не обеспечивает в полной мере выявление возможных нарушений и защита может быть дискредитирована), или на уровне модулей (программных функций), или на уровне команд процессора. В двух последних случаях сложность применения такой стратегии состоит в громоздкости конфигурирования системы защиты. Второй вариант тотального контроля заключается в том, что по всей ИС устанавливаются программные или программно-аппаратные «агенты», осуществляющие выявление любых аномалий (аномалий трафика по периметру сети, нарушений установленных правил разграничения доступа в каждом компьютере сети, сигнатур

вредоносных программ и т.п.). При этом «агенты» могут перепрограммироваться, перенастраиваться, перемещаться по сети в зависимости от складывающейся обстановки. Применение такой стратегии рассматривается для перспективных многоагентных систем защиты информации. В настоящее время системы защиты, основанные на такой стратегии, только начинают разрабатываться [27–29].

Стратегия дифференцированного контроля и реакции заключается в том, что выбираются в соответствии с некоторыми установленными правилами элементы ИС, подлежащие контролю, и определяются дифференцированно применительно к каждому элементу меры защиты в случае выявления нарушений безопасности информации.

Стратегии «периметровой» защиты и тотального контроля относятся к доминантным стратегиям, направленным на защиту любой информации в сети независимо от ее содержания и важности. Все стратегии, разделенные по признаку организации защиты, определяют, по сути, структуру соответствующей системы защиты информации в ИС.

Важным для оценки эффективности ТЗИ является разделение стратегий по направленности действий по защите. При этом действия могут быть направлены на пресечение нарушений, на обман или отвлечение нарушителя на ложные информационные ресурсы ИС, игнорирование последствий (когда пользователь считает, например, экономически более целесообразным восстанавливать информацию, а не применять комплекс дорогостоящих для него мер защиты), а также смешанные стратегии (как правило, в смешанных стратегиях имеются элементы как пресечения, так и введения нарушителя в заблуждение).

Стратегия пресечения является сегодня основной, широко применяемой на практике. Как альтернатива к стратегии пресечения сегодня разрабатываются программные средства защиты, реализующие *стратегию обмана (введения в заблуждение) или отвлечения нарушителя на ложные информационные и аппаратные ресурсы*. Такая стратегия в отечественной практике пока не используется, однако уже предусмотрена в нормативных документах [13] и может привести к серьезному развитию методологии оценки эффективности ТЗИ в ИС. Эта стратегия может быть как «периметровой» (в этом случае уже по периметру ИС устанавливаются средства отвлечения), так и эшелонированной (когда средства отвлечения устанавливаются в нескольких эшелонах защиты) или стратегией ключевых элементов (когда таким образом защищаются только наиболее важные элементы системы).

Стратегия игнорирования последствий и резервирования информации является, как правило, ущербной и ориентация на нее нецелесообразна.

Рассматривая разделение возможных стратегий по признаку адаптивности применяемых мер и средств защиты, необходимо отметить следующее. Элементы адаптивного (ситуационного) управления защитой имеются в любой системе защиты ИС. В современных системах это, как

правило, реализуется в виде возможных, заранее предусмотренных вариантов действий. Например, такие системы могут включать в себя подсистемы обнаружения сетевых атак или вредоносных программ с заранее предусмотренными мерами по блокированию действий нарушителя. В перспективных защищенных ИС могут создаваться специальные системы поддержки принятия решений по обеспечению безопасности информации в ИС в случае развития нештатных ситуаций с выбором необходимых адекватных мер защиты.

Если смотреть стратегии с точки зрения их разделения по признаку организации защиты, то существенным в данном случае является, наряду с сочетанием реализуемых мер защиты, влияние этих мер на защищенность рассматриваемого блока защищаемой информации. Для доминантных стратегий любая мера защиты влияет на защищенность любого блока защищаемой информации. Для остальных стратегий необходимо оценивать указанное влияние.

Принципиально новой является стратегия введения в заблуждение нарушителя (обмана, отвлечения). Эффективность защиты в этом случае, во многом, определяется эффективностью введения нарушителя в заблуждение относительно атакуемого ресурса, в том числе с отвлечением его на ложный информационный ресурс. В данной работе вопросы методологии оценки эффективности защиты информации для данной стратегии не рассматривается.

Факторы неопределенности, нечеткости оценки влияния мер защиты информации на реализацию угроз безопасности информации и снижение возможного ущерба обусловлены тем, что, во-первых, некоторые сведения о характеристиках применяемых программных или программно-аппаратных средств защиты, реализующих ту или иную техническую меру защиты, их настройках и т.п. сегодня, являются или ориентировочными, или обобщенными для класса средств защиты, или отсутствуют вообще. Во-вторых, практически для всех известных мер и средств защиты сегодня отсутствуют количественные оценки их влияния на защищенность информации в ИС (если не считать количественной мерой указанного влияния установленный для средства защиты класс защиты). Как правило, «снижение» возможного ущерба в результате применения той или иной меры защиты оценивается бинарным показателем «да – нет», то есть ущерб есть или его нет, так как размер возможного ущерба не является монотонной функцией от каких-либо характеристик средства защиты, влияющих лишь на возможности реализации угроз, а не на размер наносимого ущерба.

Фактор времени (динамики реализации угроз и динамики применения мер защиты информации) весьма существенно влияет на выводы о возможности реализации той или иной угрозы. Без учета фактора времени оценки возможностей реализации угроз и применения некоторых мер защиты (направленных, например, на опережение процесса реализации угрозы) оказываются или некорректными, или вообще несостоятельными.

1.3. Проблемные вопросы анализа угроз безопасности информации и пути их решения

Важной составляющей оценки эффективности защиты информации в ИС является оценка защищенности системной (технологической, то есть обеспечивающей функционирование средств вычислительной техники, информационных систем и компьютерных сетей) и пользовательской (прикладных программ и данных) информации от возможных угроз несанкционированного воздействия (копирования, модификации, блокирования и т.д.).

В связи с этим и в соответствии с действующими нормативными документами [9–13] должен проводиться анализ угроз безопасности информации, который включает в себя:

определение состава и местонахождения защищаемой системной и пользовательской информации, отнесение ее к определенным видам тайн;

определение возможных источников угроз;

выявление уязвимостей, которые могут эксплуатироваться при реализации угроз;

определение состава возможных угроз безопасности защищаемой информации в ИС;

оценку возможностей реализации угроз в ИС;

оценку ущерба от реализации угроз;

определение перечня актуальных угроз для данной ИС.

На основе такого анализа далее обосновываются необходимые меры и требования по защите (в том числе по построению системы защиты информации в ИС). Однако, как в процессе анализа, так при использовании его результатов возникает большое количество проблемных вопросов, суть которых сводится к следующему.

1. При определении состава и местонахождения защищаемой информации, а значит и при выявлении состава объектов возможного воздействия возникает несколько проблемных вопросов.

Во-первых, объемы информации в компьютерах и тем более в информационных системах исчисляются в гигабайтах- и даже терабайтах. «Вручную» просмотреть всю информацию на предмет отнесения ее к защищаемой практически невозможно (за исключением грифованной информации), а средств быстрого анализа сегодня для этого нет. Возможно появятся «умные» программные средства на основе искусственного интеллекта, которые позволят проводить такой анализ, по крайней мере, относительно пользовательской информации (текстовой, графической, видео и др.). Но сегодня для парирования этой сложности всю информацию агрегируют в группы, сходные по используемым для несанкционированного доступа к информации приемам и способам.

Сначала всю информацию в компьютере разделяют на системную и пользовательскую. Как правило, к большинству файлов системной информации, нарушение целостности или доступности которых приводит к

нарушениям в работе средства вычислительной техники, можно получить доступ только с правами администратора. Именно эти файлы и относят, в первую очередь, к защищаемой системной информации. Если в хостах ИС установлена одна и та же операционная система, то выявив файлы системной информации в одном из них, можно распространить результаты анализа состава защищаемой системной информации на другие хосты ИС. Кроме этого, к защищаемой системной информации относят некоторые файлы файловой системы, обеспечивающие возможность доступа к файлам пользовательской информации. К ним относятся файлы с парольной информацией и дескрипторные таблицы, в том числе для пользовательской информации, при этом нарушение дескрипторных таблиц приводит или к невозможности доступа к соответствующим файлам, или к резкому увеличению сроков получения доступа.

Применительно к пользовательской информации сначала формируют группу, в которые включают файлы, закрытые паролем доступа, затем формируют группы файлов, доступ к которым не закрыт паролем, и в них включаются текстовые, графические, аудио- и видеофайлы в зависимости от вызываемого редактора для просмотра этих файлов. Наконец, формируют третью группу, в которую входят базы данных, легитимный доступ к которым осуществляется через систему управления базами данных (СУБД). Если рассматриваются угрозы копирования текстовых, графических и видеофайлов без их предварительного просмотра, то формируется только одна вторая группа, в которую включаются все эти файлы. В этом случае оценивается возможность реализации угрозы относительно одного файла из группы, а результаты оценки распространяются на все файлы группы с учетом разных времен выполнения относительно них несанкционированных действий (например, копирования или уничтожения).

Во-вторых, состав пользовательской информации, в том числе подлежащей защите, как правило, изменяется ежесуточно, так как разрабатываются документы, проводятся расчеты, готовятся справки и письма, ведутся базы данных и т.д. В этих условиях для выявления состава и местонахождения защищаемых файлов необходимо вести соответствующую регистрацию появления новых файлов и регулярно пересматривать матрицу разграничение доступа к ним (матрицу полномочий, матрицу ролей). В «ручном» режиме это сделать невозможно: необходимо иметь соответствующее программное средство контроля и регистрации с автоматизированной процедурой построения матрицы разграничения доступа. Это позволит оперативно реагировать на изменение состава и местонахождения защищаемой пользовательской информации.

В-третьих, в файловой системе компьютера могут иметься одна или даже несколько копий файла с защищаемой информацией, которые могут находиться в разных каталогах, директориях и логических дисках и, возможно, в архивах. При отсутствии должной регистрации копий файлов с

защищаемой информацией создается угроза утечки защищаемой информации, несмотря на то, что основной файл (оригинал) будет защищен.

2. При определении возможных источников угроз, прежде всего, определяется состав внешних и внутренних источников. При этом в некоторых публикациях вводятся, на наш взгляд, некорректные определения этих понятий. Например, в [21] принято, что внешние нарушители как источники угроз – это субъекты, не имеющие полномочий по доступу к информационным ресурсам и компонентам систем и сетей, а внутренние нарушители – субъекты, имеющие такие полномочия. В этом случае сотрудник организации, выехавший в другой город или даже за границу, имеющий в ходе своей трудовой деятельности полномочия доступа к ресурсам и компонентам корпоративной сети предприятия и не санкционированно проникший в операционную среду хоста этой сети, будет считаться внутренним нарушителем. Если бы в таком определении было бы указано, что полномочия сотрудника сохраняются только в том случае, когда он действует только с разрешенных для него хостов сети предприятия, то такое определение можно было бы принять. Но проще дать иное определение: внешний источник угрозы – это источник, действующий или функционирующий с использованием хоста, расположенного за пределами физической или логической границы защищаемой ИС (сети) предприятия (организации), а внутренний – с использованием хоста, расположенного в пределах физической или логической границы. защищаемой ИС (сети). Если сотрудник действует санкционированно с хоста внешней сети или с хоста ИС другого предприятия (организации), то в этом случае он действует в пределах логической границы защищаемой ИС. Поскольку источником угрозы может быть не только нарушитель, но и, например, съемный инфицированный носитель информации, вредоносная программная закладка, расположенная в оперативной, в постоянной памяти компьютера или в чипсете, или программно-аппаратное закладочное устройство. В этом случае источник угрозы является внутренним, но полномочия по доступу к информационным ресурсам и компонентам здесь ни при чем.

3. При выявлении уязвимостей системного или прикладного программного обеспечения, которые могут эксплуатироваться при реализации рассматриваемых угроз, возникают неоднозначности: во-первых, одна и та же уязвимость может эксплуатироваться при реализации разных угроз, при этом сегодня отсутствуют какие-либо признаки, по которым можно было бы определить, какая из уязвимостей будет вероятнее всего использоваться при реализации конкретной угрозы; во-вторых, процесс реализации большинства угроз является многоэтапным и на каждом этапе, как правило, эксплуатируются свои уязвимости, а в целом при реализации угрозы имеет место эксплуатация нескольких уязвимостей с возможным разветвлением процесса реализации. При этом может быть несколько траекторий[3] процесса

[3] Здесь под траекторией понимается последовательность действий, которые нужно выполнить в процессе реализации угрозы от ее источника до информационного ресурса – объекта воздействия или хищения.

реализации угрозы и при оценке возможностей такой реализации приходится или рассматривать все траектории, или оценивать и выбирать наиболее вероятные из них, однако методическое обеспечение такого выбора сегодня отсутствует.

4. При определении состава возможных угроз безопасности информации необходимо иметь в виду, что угрозы считаются разными, если они различаются хотя бы одним элементом описания: источником, эксплуатируемой уязвимостью, объектом воздействия при реализации несанкционированных (деструктивных) действий, способом (или сценарием) реализации – составом действий, которые нужно выполнить при реализации угрозы. Это обусловливает огромное разнообразие возможных угроз безопасности информации и сложность охвата их при анализе. Основными путями парирования этого «проклятия размерности» являются:

во-первых, пути, описанные при рассмотрении факторов неопределенности, случайной направленности и содержания несанкционированных действий при реализации угроз безопасности информации в разделе 1.2;

во-вторых, поэтапный анализ возможных угроз, при этом, если количество анализируемых элементов описания угрозы невелико (например, количество источников угрозы, количество несанкционированных действий относительно защищаемой информации и др.), то анализируются все элементы описания, общие для всех возможных угроз;

в-третьих, если элементов описания много и охватить их сложно, то осуществляется типизация угроз для рассматриваемого этапа их реализации, то есть объединение угроз в сходные группы и выделение из них одного типового представителя, для которого далее рассматриваются общие для всех угроз данной группы типовые действия на данном этапе, например, способы проникновения в операционную среду хоста, способы перехвата трафика, способы создания ситуаций, приводящих к нарушению функционирования хоста ИС и др. Результаты такого анализа распространяются на всю группу угроз. Некоторые иные аспекты парирования сложностей охвата всего множества угроз рассматриваются в разделе 2.

5. При оценке возможностей реализации угроз в ИС и выявлении возможного ущерба от такой реализации могут иметь место несанкционированные действия (копирование, уничтожение, модификация и т.д.) как относительно одного файла, так и относительно многих файлов (папок, директорий), состав которых, как правило, неизвестен при анализе угроз. Путь парирования этой неопределенности рассматривается в разделе 2.4.2. Весьма проблемным оказывается вопрос оценки ущерба от реализации некоторых угроз. Например, ущерб от реализации угроз, связанных с копированием информации, с экологическими последствиями, со здоровьем людей и т.д., может быть оценен только с истечением некоторого времени. Так, после копирования информации ограниченного доступа ущерб будет зависеть от того, кому и когда попадет эта информация. Вместе с тем, методы

расчета такого ущерба, то есть его прогнозирования, практически отсутствуют. В этой ситуации часто принимается допущение, что возможный ущерб относится к неприемлемому, и анализируются только возможности реализации угроз, приводящих к такому ущербу (см. раздел 3).

6. При определении состава актуальных угроз для данной ИС необходимо иметь определенные критерии, в соответствии с которыми угроза должна относиться к актуальной. Сегодня такие критерии пока являются качественными: угроза считается актуальной, если по оценке обладателя информации угроза реализуема, а ее реализации приводит к существенному для обладателя ущербу. Вместе с тем в перспективе будут разрабатываться системы поддержки приятия решений, необходимых для автоматизации деятельности лиц, ответственных за организацию защиты информации в ИС. Для таких систем нужны формальные алгоритмы отнесения угроз к актуальным. Вопросы анализа угроз, связанные сразработкой таких алгоритмов и касающиеся оценки возможности реализации угроз и наносимого при этом ущерба, рассмотрены в разделах 3–6.

1.4. Методологические основы типизации информационных систем

В связи с огромным разнообразием ИС исследование вопросов ТЗИ, в том числе связанных с разработкой методологии количественной оценки ее эффективности, как правило, осуществляется для типовых систем. Типизация – это процедура, включающая в себя разделение всего рассматриваемого множества объектов на группы эквивалентности в соответствии с выбранной системой признаков типизации и выделение из каждой группы типового представителя, заменяющего без существенных ошибок любой из объектов группы. С учетом приведенного определения понятия «типизация» порядок проведения типизации ИС приведен на рисунке 1.5.

Формально задача типизации может быть записана следующим образом. Пусть все рассматриваемые ИС составляют множество Ω. Каждый ω-й объект из множества Ω описывается набором J существенных характеристик $X_{\omega j}, j=\overline{1,J}$. Из состава этих характеристик сформировано множество R признаков типизации, каждый элемент которого $X_{rj}, r \in R$, представляет собой значение или интервал значений этой характеристики, которым должны соответствовать каждая рассматриваемая характеристика ИС.

Необходимо найти такое подмножество ω_r систем $\left(\bigcup_{r\in R} \omega_r = \Omega\right)$, для которого для всех $\omega \in \omega_r$ выполняется условие $X_{\omega j} \cong X_{rj}$, то есть значения характеристик ИС, входящих в одну группу эквивалентности, совпадает со значением признака типизации (точно равен ему или лежит в установленном интервале значений).

Далее любая из ИС в составе подмножества ω_r может быть принята в качестве типовой для всех систем, входящих в эту группу эквивалентности.

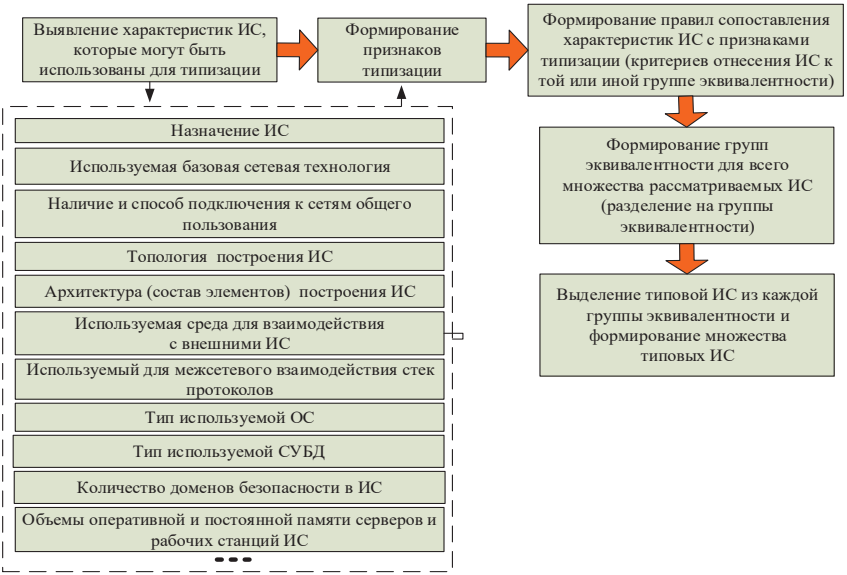


Рисунок 1.5 – Порядок типизации информационных систем

Несмотря на проводимую таким путем типизацию, состав типовых ИС может оказаться весьма большим.

В целях дальнейшего сокращения множества типовых систем в интересах проведения исследований по проблеме ТЗИ разрабатываются обобщенные описательные модели типовых систем. Разработка таких моделей может осуществляться несколькими путями.

Первый и самый простой путь состоит в сокращении перечня признаков типизации.

В этом случае в одно подмножество (кластер) могут попасть системы, зачастую существенно отличающиеся по своим характеристикам, например, с разными используемыми операционными системами, с разными видами внутрисетевой связи (по кабелю, на витой паре, на оптоволокне и др.), с мобильными и немобильными компонентами и т.д. В этом случае нельзя выделить какую-либо систему, которая могла бы быть типовым представителем (эквивалентом) для всех систем данного кластера. Поэтому создается обобщенная модель систем путем расширения интервалов возможных значений его характеристик, не вошедших в перечень признаков типизации. Платой за такой подход является расширение перечня угроз безопасности информации, которые могут быть у обобщенной модели, и вариабельность анализа защищенности информации.

Второй путь состоит в расширении интервалов значений признаков типизации. Этот путь практически равнозначен первому, так как, во-первых,

типовой представитель кластера имеет вариабельное описание (что равносильно вариации характеристик, не вошедших в признаки типизации при первом подходе). Во-вторых, характеристики обобщенной модели формируются путем суммирования или обобщения характеристик систем, входящих в кластер, а не выделением какой-либо реальной системы из этого кластера.

Третий путь состоит в том, что формируются модели сложных типовых ИС, содержащих в себе более простые типовые системы. В этом случае перечень типовых систем может быть значительно сокращен, однако при этом может значительно усложниться анализ защищенности информации.

С использованием изложенных подходов нами была проведена типизация так называемых информационно-организационных («офисных») ИС. В основу типизации положен состав угроз, которые могут иметь место в каждой ИС. При этом в качестве признаков типизации использовались: масштаб системы, ее топология построения, факт и способ подключения к сетям общего пользования. В результате к типовым информационно-организационным системам, использующих одну и ту же базовую сетевую технологию (в подавляющем большинстве случаев это технология Ethernet), в которых не используется беспроводная связь, отнесены:

отдельные автоматизированные рабочие места (АРМ) на базе компьютеров без подключения и с подключением сетям общего пользования;

локальные вычислительные сети (ЛВС) без подключения и с одноточечным или много точечным подключением к сетям общего пользования;

корпоративная компьютерная сеть без подключения и с одноточечным или много точечным подключением к сетям общего пользования;

корпоративная компьютерная сеть с одноточечным и многоточечным подключением к сетям общего пользования.

При проведении исследований некоторые характеристики типовых ИС могут варьироваться, например, тип применяемой ОС; структура и характеристики доменов безопасности; используемый стек протоколов межсетевого взаимодействия и др.

Аналогичным образом типизация может быть проведена и для большинства других ИС (управления технологическими процессами на производстве, управления транспортом и т.д.). Такая типизация позволяет абстрагироваться от характеристик конкретных ИС, обосновывать состав актуальных угроз для типовых ИС и формировать базовые описательные модели таких угроз, разрабатывать математические модели их реализации, позволяющие распространить результаты моделирования на все ИС, определять номенклатуру и содержание требований по защите информации в типовых ИС, составу и характеристика систем защиты информации в них.

В данной работе методологические аспекты оценки защищенности информации и оценки эффективности ТЗИ рассматриваются применительно к типовым ИС различного назначения.

2. ОБЩАЯ ХАРАКТЕРИСТИКА МОДЕЛЕЙ ОЦЕНКИ ЭФФЕКТИВНОСТИ ТЕХНИЧЕСКОЙ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

2.1. Классификация моделей оценки эффективности защиты информации от несанкционированного доступа

Оценивание эффективности защиты информации как меры приближения к желаемому результату защиты (см. раздел 1) должно проводиться на количественной основе, при этом результат может быть далее интерпретирован по определенным правилам в виде качественных суждений, например, «высокая», «средняя», «низкая» эффективность с возможным применением промежуточных модификаторов («очень высокая», «выше средней», «ниже средней», «очень низкая» и т.п.).

Количественная оценка эффективности защиты может применяться в зависимости от задач и применяемых мер защиты как с учетом, так и без учета фактора времени. Кроме того, эффективность может оцениваться без учета затрат на проведение мероприятий (применение мер) по защите информации, то есть так называемая функциональная эффективность, и с учетом таких затрат (иногда используется термин функционально-экономическая эффективность).

С учетом изложенного сегодня известно несколько подходов к оценке эффективности, основными из которых являются:

– функциональный подход, основанный на сравнении состава реализуемых мер защиты с составом, заданным нормативными документами, и определении степени их соответствия (подраздел 2.2);

– подход, основанный на введении оценочных уровней доверия (подраздел 2.3);

– подход, основанный на использовании теории риска *путем сравнения результатов* оценки возможностей реализации совокупности угроз и возможного ущерба от их реализации без применения и с применением мер защиты (подраздел 2.4 и 2.5). Кроме того, если учитывается фактор времени, а ущерб считается неприемлемым, то эффективность защиты может проводиться также на основе оценки возможностей опережения мерами защиты процессов реализации угроз безопасности информации в ИС.

Для реализации этих подходов необходимы соответствующие математические модели.

С учетом множества определений, данных термину «модель» в различных словарях, здесь под моделью (от лат. modulos – мера, образец) понимается образ оригинала (объкта, явления, процесса), отражающий тем или иным способом существенные черты, характеристики оригинала и позволяющий использовать образ для представления оригинала или (и) его исследования. Моделирование рассматривается как метод научного познания действительности, связанный с совершенствованием способов фиксации информации об изучаемых объектах и с получением новых знаний на основе

модельных экспериментов. Модели могут быть информационные (описательные), функциональные, математические (аналитические, логико-лингвистические и др.), имитационные, аналоговые, компьютерные и т.д. В данной монографии рассматриваются математические модели, а также функциональные модели, разрабатываемые в интересах построения математических моделей процессов реализации угроз безопасности информации в ИС.

Математическая модель[4]– это приближённое описание какого-либо класса явлений внешнего мира, выраженное с помощью математической символики. Математическое моделирование в интересах оценки эффективности защиты информации – это, по сути, моделирование процессов влияния мер защиты на возможность реализации угроз и на размер возможного ущерба от такой реализации. По результатам моделирования оценивается эффективность защиты информации в ИС как степень соответствия требуемому результату. В связи с этим такие модели далее названы *моделями оценки эффективности защиты информации* в ИС.

Классификация таких моделей приведена на рисунке 2.1, а их описание и характеристика приводятся далее.

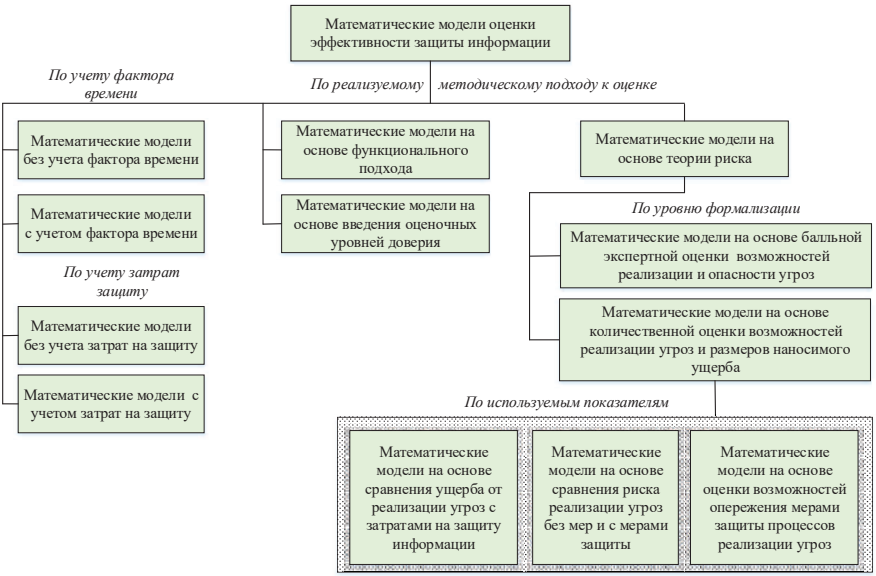


Рисунок 2.1 – Классификация моделей оценки эффективности защиты информации от несанкционированного доступа в информационных системах

[4] Большая советская энциклопедия. http://bse.uaio.ru/

2.2. Модель оценки эффективности технической защиты информации от несанкционированного доступа с применением функционального подхода

Суть функционального подхода состоит в следующем.

Пусть в ИС в соответствии с требованиями нормативного документа для обеспечения заданного уровня (класса) защищенности $k=\overline{1,K}$, должна применяться некоторая совокупность мер защиты F_k, однако фактически в ИС оказались реализованными только f_k мер защиты из совокупности F_k. Все сочетания мер защиты из множества F_k могут быть расположены в порядке возрастания их эффективности, то есть влияния на повышение защищенности информации в ИС. Так, в соответствии с [13], если ИС должна быть защищена по третьему классу защищенности, то в ней должны применяться 63 меры защиты $\left(F_k=63\right)$. Количество сочетаний таких мер N_k составит величину $2^{F_k}-1=2^{63}-1$ По мере перехода от одного сочетания к другому эффективность защиты возрастает.

Тогда в качестве ориентировочного показателя эффективности защиты информации может быть использовано соотношение:

$$\eta_k=\frac{1}{N_k}\cdot\sum_{n_k=1}^{N_k}\prod_{f_k\in n_k}\delta\left(f_k\right), \tag{2.1}$$

где $\delta\left(f_k\right)$ – единичная функция, равная единице, если мера защиты с номером f_k, входящая в сочетание n_k, реализуется в ИС, и нулю – в противном случае.

Иллюстрация зависимости эффективности защиты по третьему классу защищенности от текущего номера сочетания мер защиты n_k, приведена на рисунке 2.2.

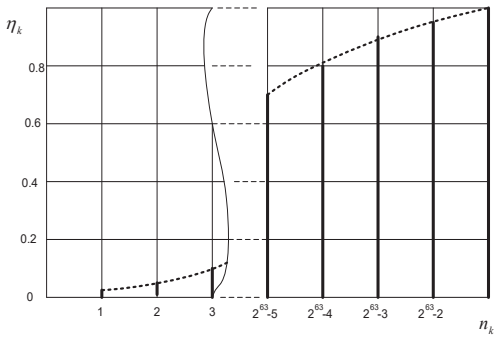


Рисунок 2.2 – Иллюстрация зависимости эффективности защиты информации от сочетания мер защиты по третьему классу защищенности ИС

Для ориентировочной оценки вместо формулы (2.1) может быть использовано соотношение:

$$\eta_k = \frac{2^{n_k^*} - 1}{2^{N_k} - 1}, \tag{2.2}$$

где n_k^* – количество сочетаний мер, в которых все меры, входящие в сочетание, реализуются.

При достаточно большом количестве сочетаний n_n^* расчет можно проводить по приближенной формуле:

$$\eta(v_k) \approx \frac{1}{2^{N_k \cdot (1 - v_k)}}, \tag{2.3}$$

где $v_k = n_k^* / N_k$ – доля реализуемых сочетаний мер защиты от общего количества сочетаний мер защиты, подлежащих реализации в ИС.

Зависимость эффективности защиты информации от доли реализуемых мер защиты информации в ИС третьего класса защищенности приведен на рисунке 2.3.

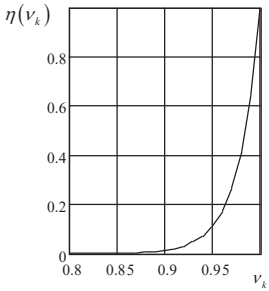


Рисунок 2.3 – Зависимость эффективности защиты информации от доли мер защиты информации в ИС третьего класса защищенности

Если применяется техническая мера f_k, реализуемая с использованием j-го средства защиты ($j = \overline{1,J}$, где J – общее количество рассматриваемых средств защиты для ИС), то вместо коэффициента $\theta(f_k)$ в формуле (2.4) используется коэффициент $\theta_j^{(cp)}(f_k)$. При этом, если средство защиты сертифицировано по заданному классу защиты [9], то полагается, что $\theta_j^{(cp)}(f_k) = 1$. Если применяется несертифицированное средство защиты, то $\theta_j^{(cp)}(f_k) = 1$.

Если класс защиты k^* для сертифицированного средства хуже требуемого $(k^* > k)$, то есть в ИС применено хоть и сертифицированное средство защиты, но не соответствующее требуемому классу защищенности,

то эффективность выполнения меры защиты с применением этого средства может быть оценена дифференцированно по формуле:

$$\theta_j^{(cp)}\left(f_k\right)=\frac{K-k^*+1}{K-k+1}\cdot\delta\left(f_k\right), \tag{2.5}$$

где K – общее количество классов защиты, установленных для данного средства защиты (сегодня $K=6$).

Например, если применяется средство обнаружения вторжений, сертифицированное по 4 классу защиты, а требуется средство по 3 классу, то эффективность выполнения меры защиты этим средством составляет величину $\theta=\frac{6-4+1}{6-3+1}=0.75$.

Если для реализации одной и той же меры защиты независимо применяются одновременно несколько средств защиты, то формула (2.4) преобразуется к следующему виду:

$$\eta_k=\frac{1}{N_k}\cdot\sum_{n_k=1}^{N_k}\cdot\prod_{f_k\in n_k}\delta\left(f_k\right)\cdot\prod_{j=1}^{J(f_k)}\theta\left(f_k\right), \tag{2.6}$$

где $J\left(f_k\right)$ – общее количество средств защиты, которые одновременно могут быть применены в ИС для реализации f_k-й меры.

Достоинство данного подхода в простоте расчетов показателя эффективности ТЗИ, однако этот подход весьма ориентировочный и содержит целый ряд недостатков, таких как:

– наличие лишь опосредованной привязки к составу и содержанию парируемых актуальных угроз безопасности;

– отсутствие сегодня оценок влияния выбираемых мер защиты на эффективности защиты информации в ИС;

– невозможность учета динамики реализации угроз (фактора времени);

– сложности оценок эффективности ТЗИ в случае применения несертифицированных средств защиты, при этом отсутствие сертификации может быть обусловлено отсутствием нормативных документов, содержащих требования к таким средствам.

Вместе с тем, такой ориентировочный расчет дает представление о степени приближения к требуемой защищенности, определенной установленным классом (уровнем) защищенности в соответствии с требованиями действующих нормативных документов.

Следует отметить, что для парирования недостатка, связанного с отсутствием сегодня оценок влияния выбираемых средств защиты на эффективности защиты информации в ИС, необходима разработка соответствующих моделей для расчета коэффициента снижения эффективности средства защиты. В связи с отсутствием таких моделей сегодня приняты следующие правила:

1) в ИС должно применяться сертифицированное средства по классу защиты в соответствии с требованиями нормативного документа;

2) в сертифицированном средстве реализуются все требования нормативного документа;

3) все требования, реализуемые сертифицированным средством защиты, являются абсолютно эффективными.

В этих условиях заранее полагается, что сертифицированное по заданному классу средство защиты всегда эффективно, при этом относительно класса, соответствующего большей защите средство вообще не рассматривается (не должно применяться).

2.3. Модель оценки эффективности технической защиты информации на основе оценочных уровней доверия

Наиболее полно требования доверия и оценочные уровни доверия к безопасности информации в ИС определены в международном стандарте ИСО/ МЭК 15408-3 [30]. В соответствии с этим стандартом доверие – это основа для уверенности в том, что ИС отвечает целям обеспечения безопасности информации. При этом используется шкала оценок в виде 7 уровней доверия, каждый из которых соответствует определенному набору требований доверия. Для упорядочения представлений об этих требованиях все множество требований разделено на классы, семейства, компоненты и элементы доверия (рисунок 2.4).

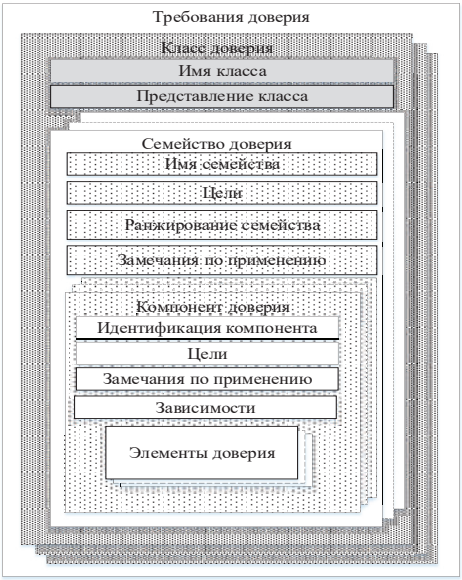


Рисунок 2.4 – Иерархическая структура представления требований доверия: класс-семейство-компонент-элемент

Краткая их характеристика приведена в таблице 2.1.

Таблица 2.1 – Краткая характеристика структурных элементов множества требований доверия

Наименование элемента	Характеристика структурного элемента
Класс доверия	Каждому классу доверия присвоено уникальное имя, которое указывает на тематические разделы, на которые распространяется класс доверия. Предусмотрена также уникальная краткая форма имени класса доверия. Она является основным средством для ссылки на класс доверия и включает в себя латинскую букву «А», за которой следуют еще две буквы латинского алфавита, относящиеся к имени класса. Каждый класс доверия имеет вводный подраздел, в котором изложены состав и назначение класса и содержит, по меньшей мере, одно семейство доверия
Семейство доверия	Каждому семейству доверия присвоено уникальное имя. Это имя содержит описательную информацию по тематическим разделам, на которые распространяется данное семейство доверия. Каждое семейство доверия размещено в пределах класса доверия, который включает в себя другие семейства той же направленности. Предусмотрена также уникальная краткая форма имени семейства доверия. Она является основным средством для ссылки на семейство доверия и включает в себя краткую форму имени класса и символ подчеркивания, за которым следуют три буквы латинского алфавита, относящиеся к имени семейства. Пункт целей семейства доверия представляет назначение семейства доверия. Этот пункт семейства доверия содержит описание имеющихся компонентов и объяснение их отличительных признаков. В семействах доверия, содержащих более одного компонента, выполнено ранжирование компонентов и приведено его обоснование. Необязательный пункт замечаний по применению семейства доверия содержит дополнительную информацию о семействе. Эта информация предназначена непосредственно для пользователей семейства доверия (например, разработчиков профилей защиты). Представление неформально и включает в себя, например, предупреждения об ограничениях использования или областях, требующих особого внимания. Каждое семейство содержит как минимум один компонент доверия
Компонент доверия	Пункт идентификации компонента содержит описательную информацию, необходимую для идентификации, категорирования, регистрации и ссылок на компонент. Каждому компоненту доверия присвоено уникальное имя, содержащее информацию о тематических разделах, на которые распространяется компонент доверия. Каждый компонент доверия входит в состав конкретного семейства доверия, с которым имеет общую цель безопасности. Предусмотрена также уникальная краткая форма имени компонента доверия. Она является основным средством для ссылки на компонент доверия и включает в себя краткую форму имени семейства, после которой ставится точка, а затем – цифра. Цифры для компонентов в пределах каждого семейства назначены последовательно, начиная с единицы Необязательный подпункт целей компонента доверия содержит конкретные цели этого компонента. Для компонентов доверия, которые имеют этот подпункт, он включает в себя конкретное назначение данного компонента и подробное разъяснение целей. Необязательный подпункт замечаний по применению компонента доверия содержит дополнительную информацию для облегчения использования компонента. Для каждого компонента доверия приведен полный список зависимостей от других компонентов доверия. При отсутствии у компонента идентифицированных зависимостей вместо списка указано: «нет зависимостей». Компоненты, иерархичные по отношению к компоненту из списка зависимостей, также могут использоваться для удовлетворения данной зависимости. Каждый компонент доверия содержит набор элементов доверия
Элемент доверия	Элемент доверия – требование безопасности, при дальнейшем разделении которого значимый результат оценки не изменяется. Элемент является наименьшим требованием безопасности. Каждый элемент доверия принадлежит к одному из трех типов: 1) элементы действий разработчика, определяющие действия, которые должны выполняться разработчиком. Этот набор действий далее уточняется доказательным материалом, упоминаемым в следующем наборе элементов. Требования к действиям разработчика обозначены латинской буквой «О» после номера элемента; 2) элементы содержания и представления свидетельств, определяющие требуемые свидетельства и отражаемую в них информацию, а также, при необходимости, – конкретные характеристики, которыми должны обладать либо объект оценки (ИС), либо данные свидетельства доверия. Требования к содержанию и представлению свидетельств обозначены латинской буквой «С» после номера элемента; 3) элементы действий оценщика, определяющие действия, которые должны выполняться оценщиком. Этот набор действий непосредственно включает в себя подтверждение того, что требования, предписанные элементами содержания и представления свидетельств, выполнены, а также – конкретные действия и анализ, выполняемые в дополнение к уже проведенным разработчиком. Требования к действиям оценщика обозначены латинской буквой «Е» после номера элемента: Действия оценщика определяют его ответственность в двух аспектах. Первый аспект – проверка правильности Профиля защиты/ Задания по безопасности в соответствии с требованиями классов APE «Оценка профиля защиты» и ASE «Оценка задания по безопасности». Второй аспект – верификация соответствия ИС функциональным требованиям и требованиям доверия. Демонстрируя, что требования выполняются, оценщик может предоставить основание для уверенности в том. что ИС будет соответствовать поставленным целям безопасности

Все требования задаются в виде подлежащих выполнению вербальных правил, например, эти правила для компонента AVA_VLA.1 – «анализ уязвимостей разработчиком» формулируются следующим образом:

цель – разработчик выполняет анализ уязвимостей, чтобы установить присутствие явных уязвимостей безопасности[5] и подтвердить, что они не могут быть использованы в предполагаемой среде объекта оценки;

элементы действий разработчика –

AVA_VLA.1.1D – разработчик должен выполнить анализ уязвимостей;

AVA_VLA.1.2D – разработчик должен предоставить документацию по результатам анализа уязвимостей;

элементы содержания и представления свидетельств –

AVA_VLA.1.1C – документация анализа уязвимостей должна содержать описание анализа поставляемых материалов объекта оценки, выполненного для поиска явных способов, которыми пользователь может нарушить политику безопасности организации;

AVA_VLA.1.2C – документация анализа уязвимостей должна содержать описание решения в отношении явных уязвимостей;

AVA_VLA.1.3C – документация анализа уязвимостей должна показать для всех идентифицированных уязвимостей, что ни одна из них не может быть использована в предполагаемой среде объекта оценки;

элементы действий оценщика –

AVA_VLA.1.1E – оценщик должен подтвердить, что представленная информация удовлетворяет всем требованиям к содержанию и представлению свидетельств;

AVA_VLA.1.2E – оценщик должен провести тестирование проникновения, основанное на анализе уязвимостей, выполненном разработчиком, для обеспечения учета явных уязвимостей.

Оценочные уровни доверия[6] (ОУД) образуют возрастающую шкалу, которая позволяет соотнести получаемый уровень доверия с затратами на обеспечение безопасности и возможностью достижения этой степени доверия. Каждый последующий ОУД представляет более высокое доверие, чем любой из предыдущих. Увеличение доверия от предыдущего ОУД к последующему достигается заменой компонента доверия иерархичным компонентом из того же семейства доверия и добавлением компонентов из других семейств доверия (то есть добавлением новых требований). Сводка ОУД представлена в таблице 2.2.

Столбцы таблицы представляют собой иерархически упорядоченный набор ОУД. а строки – семейства доверия. Каждый номер в образованной ими матрице идентифицирует конкретный компонент доверия, применяемый в соответствующем ОУД. Каждый ОУД включает в себя не более одного

---

[5] В разделе используется терминология стандарта.

[6] Под оценочным уровнем доверия понимается «набор требований доверия, представляющий некоторое положение на предопределенной шкале доверия…» [20, 30].

компонента каждого семейства доверия, при этом все зависимости каждого компонента от других компонентов доверия учтены.

Таблица 2.2 – Сводка оценочных уровней доверия

Класс доверия	Семейство доверия	Компоненты доверия из оценочного уровня доверия						
		ОУД1	ОУД2	ОУД3	ОУД4	ОУД5	ОУД6	ОУД7
Управление конфигурацией	ACM_AUT				1	1	2	2
	ACM_CAP	1	2	3	4	4	5	5
	ACM_SCP			1	2	3	3	3
Поставка и эксплуатация	ADO_DEL		1	1	2	2	2	3
	ADO_IGS	1	1	1	1	1	1	1
Разработка	ADV_FSP	1	1	1	2	3	3	4
	ADV_HLD		1	2	2	3	4	5
	ADV_IMP				1	2	3	3
	ADV_INT					1	2	3
	ADV_LLD				1	1	2	2
	ADV_RCR	1	1	1	1	2	2	3
	ADV_SPM				1	3	3	3
Руководства	AGD_ADM	1	1	1	1	1	1	1
	AQD_USR	1	1	1	1	1	1	1
Поддержка жизненного цикла	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD				1	2	2	3
	ALC_TAT				1	2	3	3
Тестирование	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	1	2	2	3
	ATE_FUN		1	1	1	1	2	2
	AT_IND	1	2	2	2	2	2	3
Оценка уязвимостей	AVA_CCA					1	2	2
	AVA_MSU			1	2	2	3	3
	AVA_SOF		1	1	1	1	1	1
	AVA_VLA		1	1	2	3	4	4

Краткая характеристика конкретных ОУД сводится к следующему.

Все ОУД условно разделены на две группы: первая – с ОУД1 по ОУД4 и вторая – с ОУД5 по ОУД7. Вторая группа отличается от первой указанием на необходимость применения (в возрастающем объеме) специальных «методов проектирования безопасности».

ОУД1 применяют, если требуется «некоторая уверенность в правильном функционировании, а угрозы безопасности не рассматривают как серьезные». Он будет полезен в случае, если требуется доверие утверждению, что было уделено должное внимание защите информации, например, персональных данных или подобной им информации. ОУД1 обеспечивает оценку в том виде, в каком он доступен потребителю, путем независимого выявления соответствия спецификации и экспертизы представленной документации. Предполагается, что оценка может успешно проводиться без помощи разработчика ИС и с минимальными затратами.

При оценке на уровне ОУД1 предоставляется свидетельство, что ИС функционирует в соответствии с документацией и при этом обеспечивается приемлемая защита от идентифицированных угроз. ОУД1 предоставляет базовый уровень доверия и обеспечивает значимое увеличение доверия по сравнению с неоцененной ИС.

Для ОУД2 предусматривается структурное тестирование и содержит требование сотрудничества с разработчиком для получения информации о проекте и результатах тестирования. Поэтому ОУД2 применяют в случаях, если разработчикам или пользователям требуется независимо подтверждаемый уровень доверия от невысокого до умеренного при отсутствии доступа к полной документации по разработке. ОУД2 также обеспечивает доверие на основе предоставления списка конфигурации объекта оценки и свидетельства безопасных процедур поставки, а также посредством анализа функций безопасности[7] с использованием функциональной спецификации, спецификации интерфейсов, руководств и эскизного проекта создаваемой (модернизируемой) ИС. Анализ поддерживается независимым тестированием функций безопасности, свидетельством разработчика об испытаниях, основанных на функциональной спецификации, выборочным независимым подтверждением результатов тестирования разработчиком, анализом стойкости функций и свидетельством поиска разработчиком явных уязвимостей (например, из общедоступных источников). ОУД2 представляет значимое увеличение доверия по сравнению с ОУД1 за счет тестирования и анализа уязвимостей разработчиком.

Для ОУДЗ предусматривается методическое тестирование и проверка, это позволяет разработчику достичь максимального увеличения доверия путем применения надлежащего проектирования безопасности. ОУДЗ применяют в случаях, если разработчикам или пользователям требуется независимо подтверждаемый умеренный уровень доверия на основе всестороннего исследования ИС. Он обеспечивает доверие путем анализа функций безопасности с использованием функциональной спецификации, спецификации интерфейсов, руководств и эскизного проекта ИС. Анализ поддерживается независимым тестированием функций безопасности, свидетельством разработчика об испытаниях, основанных на функциональной спецификации, выборочным независимым подтверждением результатов тестирования разработчиком, анализом «стойкости функций и свидетельством поиска разработчиком явных уязвимостей (например, из общедоступных источников)». ОУДЗ также обеспечивает доверие посредством использования мер управления средой разработки, управления конфигурацией ИС и свидетельства безопасных процедур поставки. Он представляет значимое увеличение доверия по сравнению с ОУД2 в связи с более полным покрытием тестированием функций и механизмов защиты[8], что дает некоторую уверенность в том, что в ИС не будут внесены искажения во время разработки.

Для ОУД4 предусматривается методическое проектирование, тестирование и углубленная проверка, что позволяет разработчику достичь

---

[7] Под функцией безопасности в соответствии с ИСО/МЭК 15408-2 здесь понимается функция элемента или некоторой совокупности элементов ИС, выполнение которой направлено на корректную реализацию функциональных требований безопасности [18].

[8] Под механизмом защиты понимается именованная совокупность действий, выполняемых с применением определенного набора мер и средств в интересах достижения сформулированной цели защиты.

максимального увеличения доверия путем применения надлежащего проектирования системы защиты и при этом не требуется глубоких специальных знаний и навыков. ОУД4 – самый высокий уровень, на который, вероятно, экономически целесообразно ориентироваться при оценке уже существующих ИС, поэтому ОУД4 применяют, если разработчикам или пользователям требуется независимо подтверждаемый уровень доверия от умеренного до высокого и имеется готовность нести дополнительные затраты, связанные с обеспечением безопасности. ОУД4 обеспечивает доверие посредством анализа функций безопасности с использованием функциональной спецификации, полной спецификации интерфейсов, руководств и эскизного и технического проекта ИС и ее подсистем. Доверие дополнительно достигается применением неформальной модели политики безопасности ИС. Анализ поддерживается независимым тестированием функций безопасности, свидетельством разработчика об испытаниях, основанных на функциональной спецификации, выборочным независимым подтверждением результатов тестирования разработчиком, анализом стойкости функций, свидетельством поиска разработчиком уязвимостей и независимым анализом уязвимостей, демонстрирующим противодействие попыткам проникновения нарушителей с низким потенциалом нападения. ОУД4 также обеспечивает доверие посредством использования мер управления средой разработки, дополнительного управления конфигурацией ИС, включая автоматизацию, и свидетельства безопасных процедур поставки. ОУД4 представляет значимое увеличение доверия по сравнению с ОУДЗ, требуя более детального описания проекта, подсистем ИС и улучшенных механизмов и/или процедур, что дает уверенность в том, что в ИС не будут внесены искажения во время разработки или поставки.

Для ОУД5 предусматривается полуформальное проектирование и тестирование. Это позволяет разработчику достичь максимального увеличения доверия путем проектирования системы защиты информации, поддержанного «умеренным применением узкоспециализированных методов». Поэтому ОУД5 применяют, если разработчикам или пользователям требуется независимо получаемый высокий уровень доверия для запланированной разработки. ОУД5 обеспечивает доверие посредством анализа функций безопасности с использованием функциональной спецификации, полной спецификации интерфейсов, руководств, эскизного и технического проектов ИС и всех ее подсистем. Доверие дополнительно достигается применением «формальной модели политики безопасности» в ИС и полуформального представления функциональной спецификации. Кроме этого, требуется модульное проектирование ИС в защищенном исполнении. Анализ поддерживается независимым тестированием функций безопасности объекта оценки, свидетельством разработчика об испытаниях, основанных на функциональной спецификации, выборочным независимым подтверждением результатов тестирования разработчиком, свидетельством поиска разработчиком уязвимостей и независимым анализом уязвимостей,

демонстрирующим противодействие попыткам проникновения нарушителей с умеренным потенциалом нападения. Анализ также включает в себя проверку правильности анализа разработчиком скрытых каналов. ОУД5 также обеспечивает доверие посредством использования контроля среды разработки, всестороннего управления конфигурацией ИС, включая автоматизацию, и свидетельства безопасных процедур поставки, представляет значимое увеличение доверия по сравнению с ОУД4, в том числе за счет полуформального описания проекта, всех подсистем ИС, более структурированной (и, следовательно, лучше анализируемой) архитектуры, анализа скрытых каналов и улучшенных механизмов и/или процедур, что дает уверенность в том, что в ИС не будут внесены искажения во время разработки.

Для ОУД6 предусматривается полуформальная верификация решений по ЗИ в ИС и тестирование, что позволяет разработчикам достичь высокого уровня доверия путем применения специальных методов проектирования ИС в защищенном исполнении в строго контролируемой среде разработки с целью защиты высоко оцениваемых ресурсов (активов) от значительных рисков. Поэтому ОУД6 применяют при разработке защищенных ИС в ситуациях высокого риска, где такая защита оправдывает дополнительные затраты. ОУД6 обеспечивает доверие посредством анализа функций безопасности с использованием для понимания режима безопасности функциональной спецификации, полной спецификации интерфейсов, руководств эскизного и технического проектов ИС и всех ее подсистем. Доверие дополнительно достигается применением формальной модели политики безопасности в ИС и полуформального представления функциональной спецификации, а также полуформальной демонстрации соответствия между ними. Кроме того, требуется модульное и иерархическое (по уровням) проектирование ИС. Анализ поддерживается независимым тестированием функций безопасности, свидетельством разработчика об испытаниях, основанных на функциональной спецификации, выборочным независимым подтверждением результатов тестирования разработчиком, анализом стойкости функций безопасности, свидетельством поиска разработчиком уязвимостей и независимым анализом уязвимостей, демонстрирующим противодействие попыткам проникновения нарушителей с высоким потенциалом нападения. Анализ включает в себя проверку правильности систематического анализа разработчиком скрытых каналов. ОУД6 также обеспечивает доверие посредством использования структурированного процесса разработки, контроля среды разработки, всестороннего управления конфигурацией ИС, включая полную автоматизацию, и свидетельства безопасных процедур поставки, представляет значимое увеличение доверия по сравнению с 0УД5, требуя всестороннего анализа, структурированного представления реализации (структурных компонентов ИС), более стройной структуры (например, с разбиением на уровни), всестороннего независимого анализа уязвимостей, систематической идентификации скрытых каналов, улучшенного управления конфигурацией и улучшенного контроля среды разработки.

Для ОУД7 предусматривается формальная верификация решений по защите информации в ИС и тестирование. Этот уровень применим для использования в ситуациях чрезвычайно высокого риска и/или там. где высокая ценность ресурсов (активов) оправдывает повышенные затраты. ОУД7 обеспечивает доверие посредством анализа функций безопасности с использованием для понимания режима безопасности функциональной спецификации, полной спецификации интерфейсов, руководств, эскизного и технического проектов ИС, а также структурированного представления реализации (структурных компонентов ИС). Доверие дополнительно достигается применением формальной модели политики безопасности в ИС, формального представления функциональной спецификации для ИС. Кроме того, требуется модульное иерархическое (по уровням) проектирование ИС. Анализ поддерживается независимым тестированием функций безопасности ИС, свидетельством разработчика об испытаниях, основанных на функциональной спецификации, полным независимым подтверждением результатов тестирования разработчиком, анализом стойкости функций, свидетельством поиска разработчиком уязвимостей и независимым анализом уязвимостей, демонстрирующим противодействие попыткам проникновения нарушителей с высоким потенциалом нападения. Анализ также включает в себя проверку правильности систематического анализа разработчиком скрытых каналов. ОУД7, кроме того, обеспечивает доверие посредством использования структурированного процесса разработки средств контроля среды разработки, всестороннего управления конфигурацией ИС, включая полную автоматизацию, и свидетельства безопасных процедур поставки, представляет значимое увеличение доверия по сравнению с ОУД6, связан со всесторонним анализом с использованием формальных представлений и формального соответствия, а также со всесторонним тестированием.

Таким образом, применение оценочных уровней доверия охватывает широкий круг вопросов, касающихся не только функциональной эффективности защиты информации, но и доверия к среде разработки программного и аппаратного обеспечения ИС и ее подсистем, к поставкам оборудования и программного обеспечения и т.д. Вместе с тем, это дает ориентировочное представление о том, насколько можно доверять реализации требований, предъявленных к функциям безопасности объекта оценки – защищаемой ИС. Однако такой подход, на наш взгляд, имеет и заметные недостатки, к которым относятся, прежде всего, такие как:

сложность сравнения вариантов решений по защите информации в ИС при наличии огромного количества вербально определенных требований и правил при отсутствии единого показателя защищенности или оценки эффективности защиты информации;

невозможность строгой оценки влияния отклонений от установленных требований на защищенность информации в ИС;

отсутствие учета временного фактора при формировании оценочных уровней доверия.

В связи с этим более перспективным, на наш взгляд, является развитие методологии оценки эффективности защиты информации в ИС на основе теории риска.

2.4. Модели оценки эффективности технической защиты информации от несанкционированного доступа с использованием теории риска

2.4.1. Общий подход к использованию теории риска при оценке эффективности технической защиты информации от несанкционированного доступа

Применение теории риска [31, 32] сегодня считается приоритетным направлением развития методологии оценки эффективности ТЗИ. Эта теория стала широко развиваться за рубежом во второй половине двадцатого века, а также в России в последнее двадцатилетие. Она позволяет учесть целый ряд "НЕ-факторов" – неопределенность, неточность, неизвестность и (или) неполнота данных об исследуемых процессах, незамкнутость (открытость) множества условий, способствующих реализации угроз.

С термином «риск» обычно связывалась некая качественная или количественная характеристика, определяющая степень потенциальной опасности некоторого рассматриваемого действия. При этом опасность определялась в виде возможного ущерба от реализации такого действия и оценки возможности этой реализации.

С учетом изложенного применялись, по крайней мере, два несколько различающихся подхода к оценке риска.

Первый подход – вероятностный сводится к оценке риска R в виде произведения вероятности $P\left(\zeta_{min} < \zeta \le \zeta_{max}\right)$ того, что при реализации конкретной u -й угрозы будет нанесен ущерб, размеры которого находятся в некотором заданном диапазоне $\left(\zeta_{min}, \zeta_{max}\right)$, и вероятности $P_u\left(t\right)$ реализации такой угрозы за заданное время t:

$$R\left(\zeta_{min}, \zeta_{max}, t\right) = P\left(\zeta_{min} < \zeta \le \zeta_{max}\right) \cdot P_u\left(t\right). \tag{2.7}$$

Сложность такого подхода заключается в том, что необходимо иметь плотности распределения вероятностей $w(u)$ нанесения ущерба, по которой может быть рассчитана вероятность $P(\zeta_{\min} < \zeta \le \zeta_{\max})$.

Для непрерывного закона распределения вероятностей эта вероятность рассчитывается по формуле

$$P\left(\zeta_{min} < \zeta \le \zeta_{max}\right) = \int_{\zeta_{min}}^{\zeta_{max}} w_{\zeta}\left(x\right) dx, \tag{2.8}$$

а для дискретного ($P\left(\zeta_{min} < \zeta \le \zeta_{max}\right) \equiv P\left(\zeta_i\right)$) – по формуле:

$$P\left(\zeta_{min} < \zeta \le \zeta_{max}\right) = \sum_i \zeta_i \cdot P\left(\zeta_i\right) \tag{2.9}$$

для всех i, для которых $\zeta_{min} < \zeta_i \le \zeta_{max}$.

Однако, как непрерывное, так и дискретное распределение вероятностей нанесения ущерба в подавляющем большинстве случаев неизвестно.

В связи с этим нашел более широкое применение *второй подход*, в котором используется оценка среднего размера $\overline{\zeta_u}$ возможного ущерба, наносимого в результате реализации u-й угрозы, и вероятность $P_u(t)$ реализации u-й угрозы. При этом риск R оценивается следующим образом:

$$R_u(t) = \overline{\zeta_u} \cdot P_u(t). \tag{2.10}$$

Однако и в этом случае рассчитать размер среднего ущерба от реализации угроз в конкретной ИС часто не удается. В связи с этим в [33] предложено использовать нормированный риск, определяемый следующим образом. Пусть для предприятия (организации) определен предельный ущерб ζ_{lim}, то есть ущерб, недопустимый для данного предприятия, превышение уровня которого исключено. Тогда нормированный риск определяется по формуле:

$$R_u^{(lim)}(t) = \frac{\overline{\zeta_u}}{\zeta_{lim}} \cdot P_u(t) = \beta_u \cdot P_u(t). \tag{2.11}$$

В [33] отношение $\overline{\zeta_u} / \zeta_{lim} = \beta_u$ названо индексом ущерба при реализации u-й угрозы, а нормированный таким образом риск – коэффициентом опасности угрозы. Далее будет рассматриваться только нормированный риск.

Как правило, в ходе реализации угрозы выполняется только одно несанкционированное действие относительно объекта воздействия (файла, каталога, директории, аппаратного компонента ИС и т.д.), при этом коэффициент опасности угрозы определяется, по сути, коэффициентом опасности этого действия. Нормированный риск от возможной реализации совокупности угроз в этом случае рассчитывается следующим образом:

$$R_\Sigma(t) = \frac{1}{\zeta_{lim}} \cdot \sum_{j=1}^{U} C_U^j \cdot \left(\sum_{u=1}^{j} \overline{\zeta_u} \right) \cdot \prod_{u=1}^{j} P_u(t) \cdot \prod_{k \neq u}^{U} \left[1 - P_k(t) \right]; \; u = \overline{1,U}, j = \overline{1,U}, \tag{2.12}$$

если реализуется хотя бы одна из множества рассматриваемых угроз;

$$R_\Sigma(t) = \frac{1}{\zeta_{lim}} \cdot \left(\sum_u \overline{\zeta_u} \right) \cdot \prod_u \left[P_u(t) \right]; \; u = \overline{1,U}, j = \overline{1,U}, \tag{2.13}$$

если реализуется всё множество рассматриваемых угроз.

Если же в ходе реализации одной u-й угрозы выполняется некоторый набор действий (например, копирование информации, передача скопированной информации на внешний носитель или по заданному сетевому адресу, а затем уничтожение информации в атакованной ИС), то коэффициент опасности определяется следующим образом:

$$R_u(t) = P_{G_u}(t) \cdot \sum_{g=1}^{G_u} \beta_g^{(u)}, \tag{2.14}$$

где $\beta_g^{(u)}$ – индекс ущерба от реализации g-го несанкционированного действия при реализации u-й угрозы;

$P_{G_u}(t)$ – вероятность реализации u-й угрозы в течение времени t с выполнением всей совокупности G_u несанкционированных действий.

В обоих из рассмотренных подходов для оценки риска необходимо определить вероятность реализации угрозы безопасности информации. Для этого необходимо иметь соответствующие математические модели. В данной работе рассматриваются как математические модели оценки возможного ущерба (раздел 3), так и математические модели оценки возможностей реализации угроз безопасности информации (раздел 4).

Нормированный риск далее используется при формировании показателей оценки эффективности ТЗИ в ИС.

Пусть цель защиты заключается в парировании всех выявленных угроз безопасности информации в ИС. В этом случае эффективность защиты оценивается по величине снижения суммарного нормированного риска от реализации совокупности угроз безопасности информации за счет применения мер и средств защиты. Показатели оценки могут быть безотносительные и относительные.

В качестве безотносительного показателя эффективности целесообразно использовать величину отклонения суммарного нормированного риска при реализации совокупности угроз в условиях применения мер защиты $R_{\Sigma}^{(ЗИ)}(t)$ и риска $R_{\Sigma}^{(0)}(t)$ в условиях их отсутствия:

$$\eta_{dif}(t) = R_{\Sigma}^{(0)}(t) - R_{\Sigma}^{(ЗИ)}(t). \tag{2.15}$$

Если в отсутствии мер защиты ущерб соответствует неприемлемому и вероятность реализации угроз за рассматриваемое время близка к единице, то формула (2.15) для расчета такого разностного показателя эффективности имеет вид:

$$\eta_{dif}(t) = 1 - R_{\Sigma}^{(ЗИ)}(t). \tag{2.16}$$

Вместе с тем более широкое применение в исследованиях по ТЗИ, наряду с разностным показателем, находят:

относительный показатель –

$$\eta_{rel}(t) = R_{\Sigma}^{(ЗИ)}(t) \big/ R_{\Sigma}^{(0)}(t),\ R_{\Sigma}^{(0)}(t) > 0; \tag{2.17}$$

относительный разностный показатель –

$$\eta_{reldif}(t) = \left| R_{\Sigma}^{(0)}(t) - R_{\Sigma}^{(ЗИ)}(t) \right| \Big/ R_{\Sigma}^{(0)}(t),\ R_{\Sigma}^{(0)}(t) > 0. \tag{2.18}$$

Если без мер защиты угроза реализуется с вероятностью, близкой к единице, то соотношение для расчета относительного и относительного разностного показателя приводятся к виду:

$$\eta_{rel}(t) = \left[\sum_{u=1}^{U} \zeta_u^{(ЗИ)} \cdot P_u^{(ЗИ)}(t) \right] \Big/ \sum_{u=1}^{U} \zeta_u^{(0)} = \sum_{u=1}^{U} \omega_u \cdot P_u^{(ЗИ)}(t); \tag{2.19}$$

$$\eta_{reldif}(t) = \left[1 - \sum_{u=1}^{U} \zeta_u^{(ЗИ)} \cdot P_u^{(ЗИ)}(t) \right] \Big/ \sum_{u=1}^{U} \zeta_u^{(0)} = 1 - \sum_{u=1}^{U} \omega_u \cdot P_u^{(ЗИ)}(t), \tag{2.20}$$

где $\zeta_u^{(0)}$ и $\zeta_u^{(ЗИ)}$ – размеры ущерба от реализации u-й угрозы без мер и с мерами защиты соответственно;

ω_u – коэффициент, определяющий долю ущерба от реализации u-й угрозы при применении мер защиты по отношению к суммарному ущербу от реализации всей совокупности угроз без применения мер защиты,

$$\omega_u = \zeta_u^{(ЗИ)} \Big/ \sum_{u=1}^{U} \zeta_u^{(0)} . \tag{2.21}$$

В большинстве случаев суммарный ущерб при отсутствии мер защиты оказывается выше предельно допустимого, то есть

$$\sum_{u=1}^{U} \zeta_u^{(0)} > \zeta_{lim} . \tag{2.22}$$

Тогда такую сумму целесообразно ограничить предельно допустимым ущербом, при этом $\omega_u \equiv \beta_u$, где по аналогии с формулой (2.11) β_u – индекс ущерба от реализации u-й угрозы.

Следует отметить, что большинство мер защиты не влияют на размер наносимого ущерба, за исключением некоторых из них, таких как меры восстановления, резервирования информации, сокращения сроков сохранения конфиденциальности и т.п., а влияют лишь на возможность реализации угроз безопасности информации. При этом для каждой из большинства мер защиты $\zeta_u^{(ЗИ)} = \zeta_u^{(0)}$.

Если рассматривать эффективность защиты информации в ИС от каждой актуальной угрозы, то $\omega_u = \beta_u = 1$ и при условии, что $P_u^{(0)}(t) \approx 1$, формулы для расчета относительного и относительного разностного показателей существенно упрощаются:

$$\eta_{rel}(t) = P_u^{(ЗИ)}(t); \tag{2.23}$$

$$\eta_{reldif}(t) = 1 - P_u^{(ЗИ)}(t). \tag{2.24}$$

Предложенные показатели, во-первых, являются условными, поскольку определяются применительно к заданной совокупности угроз безопасности информации, выявленных для конкретной ИС, и, во-вторых, непосредственно учитывают временной фактор.

В ряде случаев на практике могут быть полезными усредненные во времени показатели эффективности защиты информации:

усредненный разностный показатель –

$$\overline{Э}_{dif} = \frac{1}{T} \cdot \int_0^T \left| R_\Sigma^{(0)}(t) - R_\Sigma^{(ЗИ)}(t) \right| dt ; \tag{2.25}$$

усредненный относительный показатель –

$$\overline{Э}_{rel} = \frac{1}{T} \cdot \int_0^T \frac{R_\Sigma^{(ЗИ)}(t)}{R_\Sigma^{(0)}(t)} dt, \; R_\Sigma^{(0)}(t) > 0, \; t = \overline{0,T} ; \tag{2.26}$$

усредненный относительный разностный показатель –

$$\overline{Э}_{reldif} = \frac{1}{T} \cdot \int_0^T \frac{\left| R_\Sigma^{(0)}(t) - R_\Sigma^{(ЗИ)}(t) \right|}{R_\Sigma^{(0)}(t)} dt,\ R_\Sigma^{(0)}(t) > 0,\ t = \overline{0,T}, \tag{2.27}$$

где T – длительность периода времени, относительно которого проводится усреднение показателей.

В ряде публикаций и документов [19, 34] встречаются предложения по использовании иных показателей эффективности, в частности оценивающих предотвращенный ущерб в сравнении с затратами на меры защиты информации. Если обозначить затраты на выполнение требуемых мер защиты от совокупности угроз в ИС как Z_U, то такой показатель эффективности рассчитывается по формуле:

$$Э_U^{(затр)} = R_\Sigma(t) / Z_и . \tag{2.28}$$

Вместе с тем следует отметить, что на практике далеко не всегда удается оценить возможные затраты на организацию и проведения мероприятий по защите информации, а тем более спрогнозировать такие затраты[9].

Далее в работе будет в основном использоваться относительный разностный показатель, рассчитываемый без усреднения по времени по формулам (2.18) или (2.20).

2.4.2. Парирование сложностей, связанных с неопределенностью исходных данных и большой размерностью задачи оценки эффективности защиты информации

При оценке эффективности защиты информации в ИС возникают сложности, связанные, во-первых, с неопределенностью исходных данных, необходимых для такой оценки (см. раздел 1.2), таких как:

а) состав блоков (файлов, каталогов, директорий, записей в базе данных и др.) системной и пользовательской информации, конфиденциальность, целостность и (или) доступность которых должна быть обеспечена. При этом, если состав пользовательской информации конфиденциального характера может быть установлен экспертным путем без проблем, хотя и с возможными значительными временными затратами, то выявление состава информации, особенно системной, целостность и доступность которой должна обеспечиваться в условиях реализации возможных угроз, оказывается далеко не тривиальной задачей, особенно в динамике эксплуатации ИС, когда зачастую ежедневно меняется состав и местонахождение пользовательской информации;

б) содержание деструктивных действий, которые будут выполняться в конкретной ИС относительно конкретных блоков защищаемой информации, таких как:

[9] Далее в рамках данной работы этот подход не рассматривается.

уничтожение или модификация системных файлов (например, файлов с расширением *sys, *.prf, *.ebd, *.hdmp, *.reg, *.dat и других файлов ОС, в частности, в ОС семейства Windows всего имеется более 70 форматов системных файлов) с целью вывода из строя средства вычислительной техники в составе хостов ИС или ИС в целом (отказа функционирования или нестабильной работы);

уничтожение или модификация локальных дескрипторных таблиц или таблиц прерываний с целью нарушения доступности пользовательских файлов или исполняемых файлов прикладных программ;

уничтожение или модификации самих исполняемых файлов прикладных программ (например, исполняемых файлов офисных программ), файлов системы управления базой данных и др.;

уничтожение, модификация или копирование пользовательских файлов (текстовых, графических, видео- и т.д.);

несанкционированный запуск приложений в системе, в том числе с целью получения результатов исполнения приложений.

При этом выбор того или иного действия является прерогативой нарушителя и неизвестен администратору безопасности ИС;

в) пути и способы реализации угроз безопасности информации в ИС, в том числе в условиях применения тех или иных мер защиты. Если две угрозы различаются путями реализации, то, как правило, и время реализации таких угроз различается, а отсутствие учета фактора времени может привести к неправильному выбору и некорректной оценке эффективности мер защиты от таких угроз (например, мер, направленных на опережение процесса реализации угроз). Кроме того, не всегда известен состав уязвимостей программного обеспечения ИС, которые могут эксплуатироваться при реализации конкретных угроз, особенно когда эксплуатируются так называемые «уязвимости нулевого дня» (уязвимости, не зарегистрированные в соответствующих базах данных);

г) сведения о динамике реализации угроз в конкретных ИС и их программных и программно-аппаратных элементах, что обусловлено составом и структурой ИС, применяемым системным и прикладным программным обеспечением, производительностью аппаратных элементов ИС (процессоров серверов и рабочих станций, видеокарт, чипсета материнской платы и т.д.), загрузкой элементов ИС расчетными и иными задачами пользователей и др.

Во-вторых, значительные сложности при оценке эффективности защиты информации в ИС возникают также в связи с так называемым «проклятием размерности» задачи оценки. Это обусловлено:

большим объемом как системной, так и особенно пользовательской защищаемой информации, который может достигать сотен гигабайтов;

количеством подлежащих учету уязвимостей программного обеспечения и возможных угроз безопасности информации, вариантов

способов реализации угроз, а также разнообразием мер и количеством вариантов построения системы защиты информации в конкретной ИС и др.

В интересах парирования сложностей, связанных с неопределенностью исходных данных о составе блоков системной и пользовательской информации и *содержания деструктивных действий относительно них*, могут быть использованы следующие подходы.

1. При отсутствии разграничения доступа к конкретным файлам (что может иметь место, например, когда используется файловая система NTFS (*англ.* New Technology File System – «файловая система новой технологии»), или применяются достаточно продвинутые средства разграничения доступа, такие как Dallas Look, Secret Net и т.п.), для каждой угрозы оценивается *только возможность проникновения в операционную среду сервера или рабочей станции*. При этом полагается, что при таком проникновении может быть выполнено любое действие относительно любого защищаемого файла на сервере или рабочей станции.

Пусть решается задача защиты системных файлов, количество которых в атакуемом хосте ИС составляет величину N_{sys}. Каждый файл имеет разную важность с точки зрения безопасности функционирования хоста, определяемую для каждого n_{sys}-го файла соответствующим коэффициентом $\alpha\left(n_{sys}\right)$, который, по сути, соответствует вероятности выбора файла с номером n_{sys}, при этом $\sum_{n_{sys}=1}^{N_{sys}} \alpha\left(n_{sys}\right)=1$.

Время (τ_u), затрачиваемое на реализацию угрозы нарушения функционирования хоста ИС определяется как сумма времени получения несанкционированного доступа в операционную среду (τ_{acc}) и времени выполнения несанкционированного действия ($\tau_{destr}^{(sys)}$) относительно какого-либо системного файла, что может привести к нарушению функционирования хоста, то есть

$$\tau_u = \tau_{acc} + \tau_{destr}^{(sys)}. \tag{2.29}$$

Тогда плотность распределения вероятности для времени реализации угрозы нарушения функционирования ИС путем проникновения в операционную среду и выполнения какого-либо деструктивного действия (например, или уничтожения – деструктивное действие $g=1$, или модификации – деструктивное действие $g=2$, или блокирования – деструктивное действие $g=3$ и т.д.) определяется из соотношения:

$$w_u\left(x\right)=\int_0^x w_{acc}\left(\tau\right)\cdot w_{destr}^{(sys)}\left(x-\tau\right)d\tau, \tag{2.30}$$

где $w_{acc}\left(\tau\right)$ – плотность распределения вероятности для случайного времени τ получения нарушителем доступа в операционную среду хоста;

$w_{destr}^{(sys)}(x-\tau)$ – плотность распределения вероятности для случайного времени выполнения деструктивного действия относительно какого-либо из системных файлов с номером за время $x-\tau$ при условии, что нарушителем получен доступ в операционную среду.

Пусть вероятность того, что в ходе реализации угрозы будет выбрано деструктивное действие $g\left(g=\overline{1,G}\right)$, составляет величину γ_g, $\sum_{g=1}^{G}\gamma_g=1$. Тогда вероятность того, что за время t будет выполнено любое из рассматриваемых деструктивных действий относительно любого системного файла, рассчитывается следующим образом:

$$P_{destr}^{(sys)}(t)=\sum_{n_{sys}=1}^{N_{sys}}\alpha\left(n_{sys}\right)\sum_{g=1}^{G}\gamma_g\cdot P_{destr}^{(sys)}\left(g,n_{sys},t\right), \tag{2.31}$$

где $P_{destr}^{(sys)}\left(g,n_{sys},t\right)$ – вероятность выполнения g-го деструктивного действия за время t относительно системного файла с номером n_{sys}, которая по своим свойствам соответствует функции распределения (непрерывная, монотонная, равная 0 в точке $t=0$ и 1 на бесконечности).

Тогда плотность распределения определяется по формуле:

$$w_{destr}^{(sys)}(t)=\frac{d}{dt}P_{destr}^{(sys)}(t)=\sum_{n_{sys}=1}^{N_{sys}}\alpha\left(n_{sys}\right)\cdot\sum_{g=1}^{G}\gamma_g\cdot w_{destr}^{(sys)}\left(g,n_{sys},t\right), \tag{2.32}$$

где $w_{destr}^{(sys)}\left(g,n_{sys},t\right)$ – плотность распределения вероятности для времени выполнения g-го деструктивного действия относительно системного файла с номером n_{sys}.

Если все системные файла равнозначны и вероятности выбора деструктивных действий при реализации угроз примерно одинаковы, то формула (2.32) значительно упрощается:

$$w_{destr}^{(sys)}(t)=\frac{1}{G}\sum_{g=1}^{G}w_{destr}\left(g,n_{sys},t\right), \tag{2.33}$$

а вероятность реализации угрозы нарушения функционирования ИС с учетом соотношения (2.30) и (2.32) или (2.33) определяется традиционно:

$$P_u(t)=\int_0^t w_u(x)dx. \tag{2.34}$$

С учетом соотношения (2.31) риск выполнения деструктивных действий при реализации такой угрозы рассчитывается следующим образом:

$$R_u(t)=\sum_{n_{sys}=1}^{N_{sys}}\alpha\left(n_{sys}\right)\sum_{g=1}^{G}\gamma_g\cdot\beta\left(g,n_{sys}\right)\cdot\int_0^t\left\{\int_0^x w_{acc}(\tau)\cdot w_{destr}\left(g,n_{sys},x-\tau\right)d\tau\right\}dx, \tag{2.35}$$

где $\beta\left(g,n_{sys}\right)$ – индекс ущерба при выполнении g-го деструктивного действия относительно системного файла с номером n_{sys}.

Если все системные файла равнозначны, вероятности выбора деструктивных действий при реализации угроз примерно одинаковы и допустимо экспоненциальное приближение для плотностей распределения вероятностей, то формула (2.35) значительно упрощается:

$$R_u(t) = \beta_{sys} \cdot \left\{ \frac{\overline{\tau_{destr}^{(sys)}}}{\overline{\tau_{destr}^{(sys)}} - \overline{\tau_{acc}^{(u)}}} \cdot \left[1 - e^{-\frac{t}{\overline{\tau_{destr}^{(sys)}}}} \right] - \frac{\overline{\tau_{acc}^{(u)}}}{\overline{\tau_{destr}^{(sys)}} - \overline{\tau_{acc}^{(u)}}} \cdot \left[1 - e^{-\frac{t}{\overline{\tau_{acc}^{(u)}}}} \right] \right\}, \qquad (2.36)$$

где $\overline{\tau_{acc}^{(u)}}$ и $\overline{\tau_{destr}^{(sys)}}$ – математические ожидания времен проникновения в операционную среду при реализации u-й угрозы ИС (хоста ИС) и выполнения несанкционированного действия относительно какого-либо из системных файлов;

β_{sys} – индекс ущерба от воздействия на системный файл (нарушения функционирования хоста ИС или ИС в целом), который, как правило, можно положить равным 1 (ущерб неприемлемый).

В графическом виде полученная зависимость при $\beta_{sys} = 1$ представлена на рисунке 2.5.

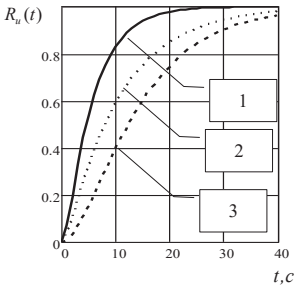


Рисунок 2.5 – Зависимость от времени риска реализации угрозы нарушения функционирования ИС путем воздействия на системные файлы: 1 – при $\overline{\tau_{acc}^{(u)}} = 5c,\ \overline{\tau_{destr}^{(sys)}} = 1c$; 2 – при $\overline{\tau_{acc}^{(u)}} = 10c,\ \overline{\tau_{destr}^{(sys)}} = 1c$; 3 – при $\overline{\tau_{acc}^{(u)}} = 10c,\ \overline{\tau_{destr}^{(sys)}} = 5c$

При наличии ограничений на доступ в операционную среду получить такой доступ нарушитель может с некоторой вероятностью p_{pass}.

В этом случае риск реализации угрозы оценивается следующим образом:

$$R_u(t) = \beta_{sys} \cdot \left\{ \frac{p_{pass} \cdot \overline{\tau_{destr}^{(sys)}}}{p_{pass} \cdot \overline{\tau_{destr}^{(sys)}} - \overline{\tau_{acc}^{(u)}}} \cdot \left[1 - e^{-\frac{t}{\overline{\tau_{destr}^{(sys)}}}} \right] - \frac{\overline{\tau_{acc}^{(u)}}}{p_{pass} \cdot \overline{\tau_{destr}^{(sys)}} - \overline{\tau_{acc}^{(u)}}} \cdot \left[1 - e^{-\frac{t \cdot p_{pass}}{\overline{\tau_{acc}^{(u)}}}} \right] \right\}. \qquad (2.37)$$

В соответствии с [35] закон распределение $w_u(t)$ достаточно близок к экспоненциальному, что позволяет использовать это приближение для весьма существенного снижения сложностей расчета (рисунок 2.6).

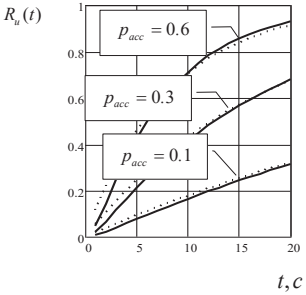


Рисунок 2.6 – Иллюстрация близости закона распределения времени реализации угрозы к экспоненциальному: жирная линия – расчет по формуле (2.37), пунктирная – экспоненциальное распределение; $\overline{\tau_{acc}^{(u)}} = 5c;\ \overline{\tau_{destr}^{(sys)}} = 1c$

Если рассматривать другие угрозы воздействия на системные файлы, отличающиеся способом проникновения в операционную среду, то достаточно оценить дополнительно лишь математическое ожидание времени проникновения $\overline{\tau_{acc}^{(u)}}$, что значительно упрощает анализ возможностей реализации всего множества возможных угроз нарушения функционирования ИС путем воздействия на системные файлы.

Аналогичным образом можно существенно снизить сложность анализа угроз нарушения целостности и доступности файлов пользовательской информации (данных и программ).

Однако применительно к файлам, содержащим конфиденциальную информацию, применяется несколько иной подход, суть которого состоит в том, что предварительно все такие файлы объединяются в группы, в каждую из которых включаются файлы с одинаковым видом тайны, к которому относятся конфиденциальные сведения, с близким уровнем конфиденциальности (или важности) и т.п.

Далее оценка риска реализации угрозы хищения информации, например, путем копирования файла, проводится применительно к одному из файлов группы с последующим распространением результатов на все файлы данной группы. Это также существенно снижает сложность анализа возможностей реализации всего множества угроз относительно файлов с конфиденциальной информацией.

2. При наличии разграничения доступа к конкретным файлам с защищаемой информацией (перечень которых в этом случае, очевидно,

известен), наряду с процедурой получения нарушителем доступа в операционную среду хоста ИС, рассматривается еще и процедура получения доступа к конкретному файлу. В этом случае математическое ожидание времени реализации угрозы рассчитывается следующим образом:

$$\overline{\tau_u} = \frac{\overline{\tau_{acc}}}{p_{pass}} + \frac{\overline{\tau_{acc}^{(f)}}}{p_{acc}^{(f)}} + \overline{\tau_{destr}^{(f)}}, \qquad (2.38)$$

где $\overline{\tau_{acc}}$ – математические ожидания времени доступа в операционную среду хоста ИС в случае отсутствия мер защиты;

$\overline{\tau_{acc}^{(f)}}$, $\overline{\tau_{destr}^{(f)}}$ – математические ожидания времени доступа к файлу с защищаемой информации при условии мер защиты от несанкционированного доступа к файлу и времени выполнения несанкционированного (деструктивного) действия относительно него соответственно;

p_{pass}, $p_{acc}^{(f)}$ – вероятности получения доступа в операционную среду хоста ИС и к защищаемому файлу соответственно.

Плотность вероятности для времени τ_u определяется как свертка плотностей вероятностей случайных величин $\tau_{acc}, \tau_{acc}^{(f)}$ и $\tau_{destr}^{(f)}$, то есть:

$$w_u(t) = \int_0^t w_{acc}(\tau) \cdot \left\{ \int_0^{t-\tau} w_{acc}^{(f)}(y) w_{destr}^{(f)}(t-\tau-y) dy \right\} d\tau, \qquad (2.39)$$

где $w_{acc}^{(f)}(y)$ и $w_{destr}^{(f)}(t-\tau-y)$ – плотности вероятности времени получения доступа к файлу за время y и выполнения относительно него несанкционированного (деструктивного) действия за время $t-\tau-y$ соответственно.

При этом риск реализации угрозы и нанесения ущерба за счет воздействия на защищаемый файл по аналогии с формулами (2.30) – (2.35) определяется из соотношения:

$$R_u(t) = \sum_{g=1}^{G} \gamma_g \cdot \beta_g \cdot \int_0^t \left\{ \int_0^x w_{acc}(\tau) \cdot \left[\int_0^{x-\tau} w_{acc}^{(f)}(y) \cdot w_{destr}^{(f)}(x-\tau-y) d\tau \right] \right\} dx. \qquad (2.40)$$

При равновероятном выборе деструктивного действия и примерно равных последствиях от них при экспоненциальном приближении риск реализации угрозы относительно рассматриваемого файла рассчитывается по формуле:

$$R_u(t) = \beta \cdot \left\{ \begin{array}{l} \dfrac{\overline{\tau_3}^2}{\left(\overline{\tau_3}-\overline{\tau_2}\right)\left(\overline{\tau_3}-\overline{\tau_1}\right)} \left[1 - e^{-\frac{t}{\overline{\tau_3}}} \right] - \dfrac{\overline{\tau_1} \cdot \overline{\tau_3}}{\left(\overline{\tau_3}-\overline{\tau_2}\right)\left(\overline{\tau_3}-\overline{\tau_1}\right)} \left[1 - e^{-\frac{t}{\overline{\tau_1}}} \right] - \\ - \dfrac{\overline{\tau_2}^2}{\left(\overline{\tau_3}-\overline{\tau_2}\right)\left(\overline{\tau_2}-\overline{\tau_1}\right)} \left[1 - e^{-\frac{t}{\overline{\tau_2}}} \right] + \dfrac{\overline{\tau_1} \cdot \overline{\tau_2}}{\left(\overline{\tau_3}-\overline{\tau_2}\right)\left(\overline{\tau_2}-\overline{\tau_1}\right)} \left[1 - e^{-\frac{t}{\overline{\tau_1}}} \right] \end{array} \right\}, \qquad (2.41)$$

где $\overline{\tau_1}$, $\overline{\tau_2}$, $\overline{\tau_3}$ – математические ожидания времен, определяемых как

$$\overline{\tau_1} = \frac{\overline{\tau_{acc}}}{p_{pass}}; \overline{\tau_2} = \frac{\overline{\tau_{acc}^{(f)}}}{p_{pass}}; \overline{\tau_3} = \overline{\tau_{destr}^{(f)}}. \qquad (2.42)$$

Графики полученной зависимости риска реализации угроз от времени приведен на рисунке 2.7 и 2.8. Их анализ показывает, что, как и ожидалось, снижение времен доступа в операционную среду хоста, к файлу и выполнения деструктивного действия существенно снижает риск реализации угрозы.

Парирование сложностей, связанных с отсутствие сведений о путях и способах реализации угроз, а также сведения о динамике реализации угроз в конкретных ИС и их программных и программно-аппаратных элементах может быть достигнуто, во-первых, моделированием процессов реализации угроз, методологические основы которого рассмотрены в разделах 4–6.

Во-вторых, сегодня необходим сбор таких сведений применительно к типовым ИС и основным классам угроз, в том числе в условиях применения известных мер и средств защиты.

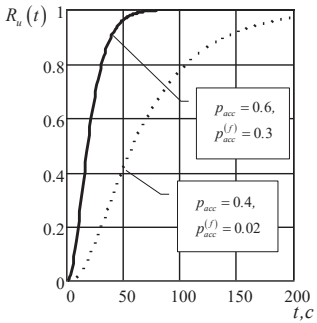


Рисунок 2.7 – График зависимости риска реализации угроз от времени и вероятностей доступа в операционную среду и к файлу: $\overline{\tau_{acc}} = 5c, \overline{\tau_{acc}^{(f)}} = 1c,\ \overline{\tau_{destr}^{(f)}} = 10c$

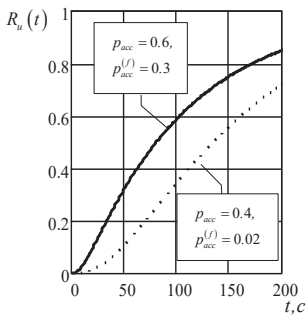


Рисунок 2.8 – График зависимости риска реализации угроз от времени и вероятностей доступа в операционную среду и к файлу: $\overline{\tau_{acc}} = 5c, \overline{\tau_{acc}^{(f)}} = 1c,\ \overline{\tau_{destr}^{(f)}} = 100c$

Следует подчеркнуть важность автоматизации анализа угроз безопасности информации в ИС и оценки эффективности защиты от них для парирования возникающих сложностей при таком анализе и оценке. Это обусловлено как большим объемом, так и значительной трудоемкостью требуемых при этом расчетов.

2.5. Балльный метод оценки эффективности защиты информации

Балльный метод оценки возможностей реализации угроз безопасности информации основывается на экспертном опросе специалистов, обработке результатов опроса и выдаче их в виде баллов, а затем интерпретации полученной балльной оценки в виде суждений о риске реализации угрозы и эффективности защиты [33, 36].

Этот метод реализован в целом ряде международных стандартов и программных продуктов, таких как стандарт ISO 17799[10] и его инструментарий, например, программный продукт COBRA или программный продукт, реализующий метод CRAMM (CCTA[11] Risk Analysis & Management Method – метод CCTA анализа и контроля рисков), программный продукт RiskWatch и др. Наиболее широко в настоящее время используется метод CRAMM, поэтому на его примере ниже характеризуется балльный метод оценки.

В этом методе в первую очередь оценивается ценность ресурсов в ситуациях:

– недоступности ресурса в течение определенного периода времени;

– разрушения ресурса – потери информации, полученной со времени последнего резервного копирования, или ее утраты (полного разрушения);

– нарушения конфиденциальности в случаях НСД;

– модификации данных из-за мелких ошибок ввода, программных ошибок, преднамеренных действий;

– наличия ошибок, связанных с передачей информации, в том числе с отказом в обслуживании, со сбоем в доставке информации, с доставкой по неверному адресу.

В качестве возможных ущербов в коммерческой версии продукта рассматриваются:

– ущерб репутации организации;

– нарушение действующего законодательства;

– ущерб для здоровья персонала;

– ущерб, связанный с разглашением персональных данных;

– финансовые потери от разглашения информации;

– финансовые потери, связанные с восстановлением ресурсов;

– дезорганизация деятельности и др.

Далее формируются балльные шкалы оценки ущерба в интервале значений 0–10 (таблица 2.3).

---

[10] Стандарт ISO/IES 17799 – стандарт по информационной безопасности, опубликованный в 2005 г. организациями ISO и IEC. В 2013 г. сменил название на ISO/IEC 27002:2005 «Информационные технологии. Методы обеспечения безопасности. Практические правила управления информационной безопасностью».

[11] CCTA (Central Computer and Telecommunications Agency) – Центральное агентство по компьютерам и телекоммуникациям Великобритании.

Таблица 2.3 – Пример формирования шкалы оценки ущерба

Вид ущерба	Величина ущерба, соответствующая оценке в баллах				
	2 балла	4 балла	6 баллов	8 баллов	10 баллов
Ущерб репутации организации	Негативная реакция отдельных чиновников, общественных деятелей	Критика в СМИ, не получившая широкого общественного резонанса	Негативная реакция депутатов Государственной Думы или Совета Федерации	Критика в СМИ с последствиями в виде крупных скандалов, парламентских слушаний, широкомасштабных проверок	Негативная реакция на уровне Президента и Правительства
Ущерб для здоровья персонала	Минимальный ущерб, не связанный с госпитализацией или длительным лечением	Средний ущерб, связанный лечением одного или нескольких сотрудников при отсутствии длительных отрицательных последствий	Серьезные последствия (продолжительная госпитализация, инвалидность одного или нескольких сотрудников)	-	Гибель людей
Финансовые потери, связанные с восстановлением ресурсов	Менее 1000 долл.		От 1000 долл. до 10000 долл.	От 10000 долл. до 10000 долл.	Свыше 10000 долл.
Дезорганизация деятельности в связи с недоступностью данных	До 15 минут	От 15 минут до 1 часа	От 1 часа до 3 часов	От 3 часов до 1 суток	Более 1 суток
■■■	■■■	■■■	■■■	■■■	■■■

Затем оцениваются уровни угроз и уязвимостей на основе экспертного опроса по специально составленным анкетам и учета таким образом разнообразных косвенных факторов, таких как:

– статистика по зарегистрированным инцидентам безопасности информации (нарушениям);

– тенденции в статистике по нарушениям;

– наличие в системе информации, представляющей интерес для возможных внутренних и внешних нарушителей;

– моральные качества персонала;

– возможность извлечь выгоду из изменения обрабатываемой в системе информации;

– наличие альтернативных способов доступа к информации;

– количество рабочих мест операторов в системе;

– осведомленность руководства о действиях сотрудников;

– полномочия пользователей и др.

Примеры заполнения ответов и выставления баллов по ответам при оценке угроз выполнения несанкционированных действий в ИС приведены в таблице 2.4, а примеры заполнения ответов и выставления баллов по ответам при оценке уязвимостей по балльной шкале – в таблице 2.5.

Таблица 2.4 – Пример оценки угроз выполнения несанкционированных действий по балльной шкале по ответам экспертов

Вопрос	Ответ	Количество баллов
Сколько раз сотрудники в последние три года пытались получить НСД к информации в системе?	Ни разу	0
	Один или два раза	10
	В среднем раз в год	20
	В среднем чаще одного раза в год	30
	Неизвестно	10
Какова тенденция в статистике такого рода попыток НСД?	К возрастанию	10
	Оставаться постоянной	0
	К снижению	-10
Хранится ли в информационной системе информация, которая может представлять интерес для сотрудников организации?	Да	5
	Нет	0
Известны ли случаи нападения, угроз, шантажа, давления на сотрудников со стороны посторонних лиц?	Да	10
	Нет	0
Есть ли среди персонала лица с недостаточно высокими моральными качествами?	Нет	0
	Есть, но это вряд ли может спровоцировать их на НСД в систему	5
	Есть и они вполне способны совершить действия по НСД	10
Хранится ли в системе информация, несанкционированное изменение которой может принести прямую выгоду сотрудникам?	Да	5
	Нет	0
Имеются ли в системе программные и иные средства, способствующие совершению НСД?	Нет	0
	Да	5
Имеются ли другие способы изменения информации, позволяющие злоумышленнику получить выгоду более простым способом, чем НСД?	Да	-10
	Нет	0
Сколько раз в последние три года сотрудники пытались получить НСД к информации, хранящейся в других системах организации?	Ни разу	0
	Один или два раза	5
	В среднем раз в год	10
	Чаще одного раза в год	15
	Неизвестно	10

Таблица 2.5 – Пример оценки уязвимостей по балльной шкале

Вопрос	Ответ	Количество баллов
Сколько людей имеют право пользоваться системой?	От 1 до 10	0
	От 11 до 50	4
	От 51 до 200	10
	От 200 до 1000	14
	Свыше 1000	20
Будет ли руководство осведомлено, что сотрудники ведут себя необычным образом?	Да	0
	Нет	10
Какие устройства и программы доступны пользователю?	Только терминалы или сетевые контроллеры, обеспечивающие только предоставление и маршрутизацию информации	-5
	Только стандартные офисные устройства и программы, а также управляемые с помощью меню подчиненные прикладные программы	0
	Пользователи могут получить доступ к ОС, но не к среде разработки программ	5
	Пользователи могут получить доступ к среде разработки программ	10
	Да	10

Вопрос	Ответ	Количество баллов
Возможны ли ситуации, когда сотрудники, предупрежденные о предстоящем сокращении или увольнении, имеют доступ к системе?	Нет	0
Каковы в среднем размеры рабочих групп сотрудников в подразделении, имеющих доступ к системе?	Менее 10 человек	0
	От 11 до 20 человек	5
	Свыше 20 человек	10
Станет ли факт изменения информации в системе очевидным для многих сотрудников?	Да	0
	Нет	10
Насколько велики официально предоставленные пользователям возможности по просмотру всех хранящихся в системе данных?	Официальное право предоставлено всем пользователям	-2
	Официальное право предоставлено только некоторым пользователям	0
Насколько необходимо пользователям знать всю информацию, хранящуюся в системе?	Всем пользователям необходимо знать всю информацию, хранящуюся в системе	-4
	Отдельным пользователям необходимо знать всю информацию, хранящуюся в системе	0

Баллы по результатам оценки угрозы суммируются и по количеству баллов оценивается степень опасности угрозы следующим образом:

до 9 – очень низкая;

от 10 до 19 – низкая;

от 20 до 29 – средняя;

от 30 до 39 – высокая;

40 и более – очень высокая.

Аналогичным образом оценивается по баллам степень уязвимости:

до 9 – низкая;

от 10 до 19 – средняя;

20 и более – высокая.

Далее определяется показатель цены потери по вербальной шкале:

N (Negligible) – воздействием можно пренебречь;

Mi (Minor) – незначительные последствия (легко устранимы, с малыми затратами на ликвидацию);

Mo (Moderate) – последствия умеренны, не связаны с крупными затратами, не затрагивают критически важные задачи;

S (Serious) – серьезные последствия со значительными затратами, влияющими на выполнение критически важных задач;

C (Critical) – невозможно решение критически важных задач.

Затем баллы по результатам оценки угрозы суммируются и по количеству баллов оценивается уровень угрозы следующим образом:

до 19 – низкая;

от 20 до 29 – средняя;

от 30 и выше – высокая.

На основе проведенных оценок определяется показатель риска, например, по десятибалльной шкале в соответствии с таблицей 2.6, а затем с его учетом оценивается опасность угрозы (таблица 2.7).

Таблица 2.6 – Показатель риска

Цена потери	Уровень угрозы								
	Н			С			В		
	Уровни уязвимости			Уровни уязвимости			Уровни уязвимости		
	Н	С	В	Н	С	В	Н	С	В
N	0	1	3	1	3	5	2	4	6
Mi	1	2	4	2	4	6	3	5	7
Mo	2	3	5	3	5	7	4	6	8
S	3	4	6	4	6	8	5	7	9
C	4	5	7	5	7	9	6	8	10

*Н – низкий, С – средний, В – высокий

Таблица 2.7 – Оценка опасности угрозы по вербальной шкале

Показатель риска в баллах	Интерпретация оценки риска в вербальной шкале	Вербальная оценка опасности угрозы
1 балл и менее	Очень низкий риск	Опасность очень низкая
2 или 3 балла	Низкий риск	Опасность низкая
4 или 6 баллов	Средний риск	Опасность средняя
7-9 баллов	Высокий риск	Опасность высокая
10 баллов	Недопустимый риск	Опасность чрезвычайная

В этом случае, по сути, оценивается так называемый остаточный риск реализации угроз, который может иметь место даже при применении мер защиты. Тогда эффективность может быть рассчитана, например, при использовании относительно-разностного показателя по формуле (2.18).

Несмотря на то, что балльный метод нашел широкое применение на практике, он имеет ряд недостатков, таких как:

не учитывается фактор времени, который зачастую играет решающую роль при оценке возможности реализации угроз безопасности информации в ИС;

отсутствует мера уверенности в правильности суждений экспертов при выставлении балльных оценок;

не разработаны корректные правила оценки остаточного риска при применении различных мер и средств защиты информации, то есть оценки влияния мер и средств защиты на возможности реализации угроз в ИС.

3. МЕТОДОЛОГИЯ ОЦЕНКИ УЩЕРБОВ ОТ РЕАЛИЗАЦИИ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ

3.1. Классификация ущербов от реализации угроз безопасности информации

Оценка опасности угрозы безопасности информации непосредственно связана с оценкой возможного ущерба от ее реализации. Ущерб возникает в результате нарушения конфиденциальности, целостности или доступности информации, относящейся либо к пользовательской, либо к системной. Нарушения конфиденциальности касаются, в основном, пользовательской информации, нарушения целостности и доступности – как пользовательской, так и системной информации, при этом нарушения целостности и доступности системной информации являются, как правило, непосредственными причинами нарушения доступности пользовательской информации.

По степени опосредованности различаются ущерб, наносимый непосредственно самой защищаемой информации (информационный ущерб), и опосредованный ущерб, обусловленный возникающими последствиями от реализации угроз, включающий в себя финансовый, материальный, экологический, экономический, репутационный, моральный ущербы и ущерб здоровью людей (рисунок 3.1).

Рассматривая *информационный ущерб*, необходимо отметить, что его оценка может осуществляться следующим образом.

1. На основе определения объема (относительного объема) информации, подвергшейся воздействию в результате реализации угрозы. Как правило, при этом рассматривается пользовательская информация (данные или пользовательские программы – исполняемые файлы). В этом случае размер ущерба оценивается, как доля пользовательской информации, относительно которой осуществлены или могут быть осуществлены несанкционированные действия:

$$\Phi_{user} = \frac{1}{V_{user}} \cdot \sum_{j=1}^{J_{user}} \sum_{u=1}^{U_j} \begin{bmatrix} p_{copy}(u,j) \cdot V_{copy}^{(u)} + p_{change}(u,j) \cdot V_{change}^{(u)} + \\ + p_{move}(u,j) \cdot V_{move}^{(u)} + p_{del}(u,j) \cdot V_{del}^{(u)} \end{bmatrix}, \qquad (3.1)$$

где $V_{copy}^{(u)}, V_{change}^{(u)}, V_{move}^{(u)}, V_{del}^{(u)}$ – объемы скопированной, измененной, перемещенной, уничтоженной пользовательской информации при реализации u-й угрозы;

V_{user} – общий объем защищаемой пользовательской информации;

$p_{copy}(u,j), p_{change}(u,j), p_{move}(u,j), p_{del}(u,j)$ – вероятности того, что при реализации u u-й угрозы будет выбрано несанкционированное действие, направленное соответственно на копирование, изменение, перемещение или уничтожение j-го файла, при этом

$p_{copy}(u,j) + p_{change}(u,j) + p_{move}(u,j) + p_{del}(u,j) = 1$;

U_j – общее количество угроз, направленных на j-й файл.

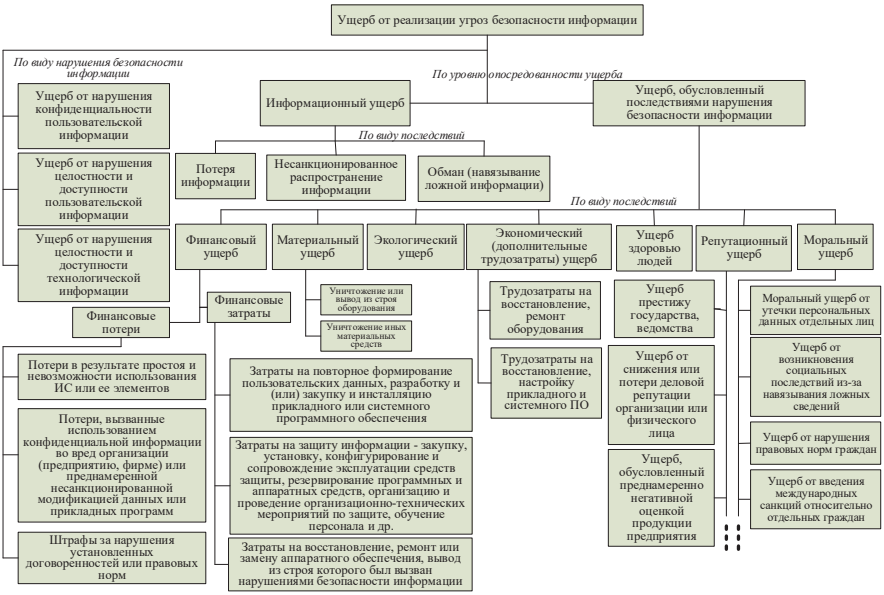


Рисунок 3.1 – Классификация ущербов от реализации угроз безопасности информации

Основными недостатками такого подхода являются, во-первых, невозможность оценки ущерба, связанного со сложностью обнаружения изменения объемов файлов при сохранении их работоспособности (изменением кода исполняемых файлов при сохранении их объема), во-вторых, сложность прогноза состава файлов пользовательской информации, относительно которых будет реализовываться та или иная угроза, в-третьих, неопределенность выбора источником угрозы того или иного несанкционированного действия.

2. На основе оценки ценности информации, подвергшейся воздействию в результате реализации угрозы. Вопросам оценки ценности информации было посвящено много работ, в которых предложены различные подходы, например, основанные [37]:

а) на теории информации К. Шеннона. При этом вводится понятие цели, для достижения которой используется защищаемая информация и полагаются измеримыми вероятности достижения цели [38]. Если P_0 – вероятность остижения цели при отсутствии защищаемой информации, а P_I – вероятность достижения цели при наличии этой информации Z_I, то ее ценность определяется следующим образом:

$$Z_I = log_2 P_I - log\, P_0 = log\left(\frac{P_I}{P_0}\right),\ P_0 > 0.4; \qquad (3.2)$$

б) на энтропийном подходе, при котором полагается, что пользователь без информации действовал в условиях неопределенности, оцениваемой энтропией H, и при этом имел средние потери $R(H)$. С информацией I энтропия стала величиной $H(I)$, а потери – $R\left[H(I)\right]$ [39]. Тогда ценность информации определяется разностью

$$Z_I = R(H) - R\left[H(I)\right]; \tag{3.3}$$

в) на оценке времени устаревания информации [40], когда сравниваются затраты времени на достижения цели функционирования без ее использования T_0 и с использованием T_I, при этом ценность информации определяется из соотношения:

$$Z_I = \left(1 - \frac{T_I}{T_0}\right)^{\alpha}, \tag{3.4}$$

где α – коэффициент динамики старения информации;

г) на «атрибутивном подходе» [37], в котором полагается, что каждая информация обладает некоторым набором измеримых атрибутов («точность», «достоверность», «актуальность» и т.д.), обозначаемых $a_{I1}, a_{I2}, ... a_{In}, ... a_{IN}$. Тогда общая ценность информации определяется как функция $S_I = F\left(a_{I1}, a_{I2}, ..., a_{In}, ..., a_{IN}\right)$, которая при линейной аппроксимации имеет вид:

$$S_I = \sum_{n=1}^{N} \xi_{In} \cdot a_{In}, \tag{3.5}$$

где ξ_{In} – весовые коэффициенты, определяемые экспертным путем, и т.д.

Необходимо отметить, что методы оценки информационного ущерба на основе определения ценности информации пока не находят какого-либо значимого применения в области защиты информации. Это во многом объясняется сложностью увязки такой оценки с содержанием угроз безопасности информации и преимущественным использованием экспертных процедур при проведении таких оценок.

Финансовый ущерб может включать в себя упущенную выгоду, финансовые затраты на защиту информации, восстановление данных и программ, закупку и замену аппаратного обеспечения и т.п., и финансовые потери. Сегодня существуют методы оценки этих составляющих финансового ущерба, например, по расчету затрат на повторное формирование или восстановление пользовательских данных, на разработку и/или инсталляцию, возможно с закупкой, прикладного программного обеспечения, на восстановление, ремонт или замену аппаратного обеспечения, на закупку, установку, конфигурирование и сопровождение эксплуатации средств защиты и т.п., а также по определению штрафов за нарушения установленных договоров или правовых норм. Вместе с тем, такие оценки, как правило, являются апостериорными, то есть проводимыми после реализации угроз безопасности информации, а не прогнозными, что востребовано практикой защиты информации. Это обусловлено и тем, что отсутствует корректная

увязка финансовых потерь с содержанием блоков информации, относительно которых проводятся несанкционированные действия в ходе реализации угроз безопасности информации.

Более подробно некоторые аспекты оценки финансового ущерба рассмотрены в следующем разделе.

Материальный, экономический, экологический ущерб и ущерб здоровью персонала могут, как правило, в весьма ограниченных случаях пересчитываться в финансовый ущерб. В частности, разработан и применяется комплекс методик для оценки материального, экологического, экономического и финансового ущерба в результате возникновения и развития чрезвычайных ситуаций [41]. Однако эти методики не учитывают особенности развития чрезвычайных ситуаций, обусловленных нарушениями безопасности информации в критически важных информационных системах [42]. Это не позволяет их использовать непосредственно в практике оценки эффективности ТЗИ.

Что касается *репутационного* и тем более *морального ущербов,* то методики их корректного пересчета в финансовый ущерб пока отсутствует.

Весьма проблематичной в связи с изложенным является оценка:

ущерба от нарушения конфиденциальности информации, поскольку такой ущерб зависит от того, кто и каким образом воспользуется такой информацией во вред государству, организации, владеющей этой информацией, или во вред конкретной личности, через какое время будет использована информация ограниченного доступа и др.;

суммарного ущерба от реализации угроз безопасности информации.

Кроме того, весьма сложно учесть все факторы, влияющие на размеры ущерба, многообразие последствий от реализации угроз и т.д.

Наконец, значительно сложнее осуществлять прогноз возможного ущерба на некоторый прогнозный период времени, чем оценивать ущерб, когда считается, что в период времени до начала реализации угрозы ничего в условиях ее реализации не изменится. Соответствующая методология прогнозирования возможных ущербов на прогнозный период сегодня отсутствует.

С учетом изложенного следует констатировать, что сегодня официально признанных методик количественной оценки возможных ущербов от реализации угроз безопасности информации в ИС нет.

3.2. Некоторые аспекты методологии количественной оценки финансового ущерба от реализации угроз безопасности информации

Рассмотрим некоторые аспекты методологии количественной оценки финансового ущерба на примере гипотетического варианта оценки финансового ущерба и пересчитываемых в него ущербов от нарушений безопасности пользовательской и системной информации в ИС, когда несанкционированные действия связаны с финансовыми потерями.

Пусть относительно i-го блока информации возможно выполнение g-го несанкционированного действия в результате реализации угрозы с ущербом, составляющим величину Φ_{ij}. Это может привести к тому, что будет нарушен доступ пользователя к некоторым другим блокам информации или станет невозможным использование другой прикладной программы. Такая взаимосвязь блоков информации может быть описана формально в виде матрицы $\Delta_g = \left|\delta_{ij}\right|, (i, j) = \overline{1.J}$, для каждого g-го несанкционированного действия, где δ_{ij} единичная функция, равная единице, если нарушение безопасности i-го блока при g-м несанкционированном действии приводит к нарушению безопасности j-го блока информации, J – общее количество блоков защищаемой информации. Возможные финансовые затраты и потери, которые в данном случае определяют суммарный ущерб от выполнения g-го несанкционированного действия в ходе реализации угроз безопасности информации, приведены на рисунке 3.2.

Рисунок 3.2 – Затраты (потери), обусловливающие суммарный ущерб от реализации несанкционированного действия относительно защищаемого блока информации

Тогда суммарный ущерб относительно i-го блока информации составит величину:

$$\Phi_{ig}^{(\Sigma)}(t) = \Phi_{ig}\left(t, \bigcup\bigcup\bigcup \psi_h\left[z_{ks}(i,g)\right]\right) + \sum_{\substack{j=1 \\ i \neq j}}^{J} \delta_{ijg} \cdot \Phi_{ig}\left(t, \bigcup\bigcup\bigcup \psi_h\left[z_{ks}(i,g)\right]\right), \qquad (3.6)$$

где $\Phi_{ig}\left(t, \bigcup\bigcup\bigcup \psi_h\left[z_{ks}(i,g)\right]\right)$ – затраты, которые могут иметь место за время t при нарушении безопасности i-го блока информации, используемого при

выполнении автоматизируемой функции z_{ks}, от которой зависит показатель функционирования предприятия $\psi_h, h \in H$,

$$\Phi_{ig}\left(t, \bigcup_k \bigcup_s \bigcup_h \psi_h\left[z_{ks}(i,g)\right]\right) = \Phi_{ig}^{\{rep\}}(t) + \Phi_{ig}^{\{rem\}}(t) + \Phi_{ig}^{\{o\}}\left(t, \bigcup_k \bigcup_s \bigcup_h \psi_h\left[z_{ks}(i,g)\right]\right) +$$
$$+\Phi_{ig}^{(mod)}\left(t, \bigcup_k \bigcup_s \bigcup_h \psi_h\left[z_{ks}(i,g)\right]\right) + \Phi_{ig}^{\{conf\}}(t) + \Phi_{ig}^{\{IT\}}(t) + \Phi_{ig}^{(st)}\left(t, \bigcup_k \bigcup_s \bigcup_h \psi_h\left[z_{ks}(i,g)\right]\right), \quad (3.7)$$

где $z_{ks}(i,g)$ – k-я компонента функции, характеризующей деятельность предприятия (организации) в s-й сфере и влияющей на ущерб от реализации g-го несанкционированного действия относительно i-го блока информации (обозначения функций $\Phi_{ig}^{(rep)}, \Phi_{ig}^{(rem)}, \Phi_{ig}^{(o)}, \Phi_{ig}^{(mod)}, \Phi_{ig}^{(conf)}, \Phi_{ig}^{(st)}, \Phi_{ig}^{(IT)}$ приведены на рисунке 3.2).

Аналогичным образом оцениваются затраты, вызванные нарушениями безопасности информации других блоков, обусловленными несанкционированным действием относительно i-го блока информации.

Использование данного подхода связано с разработкой комплекса методик оценки затрат по всем указанным составляющим суммарного ущерба.

По некоторым составляющим сегодня существуют экономические методы оценки (например, по расчету затрат на повторное формирование или восстановление пользовательских данных, разработку и/или инсталляцию, возможно с закупкой, прикладного программного обеспечения, на восстановление, ремонт или замену аппаратного обеспечения, штрафы за нарушения установленных договоров или правовых норм, на закупку, установку, конфигурирование и сопровождение эксплуатации средств защиты и т.п.) или предложены подходы к оценке (например, потерь, вызванных утечкой информации). Однако при этом отсутствует увязка потерь с содержанием блоков информации, относительно которых проводится деструктивное действие.

Сегодня фактически отсутствуют методы оценки потерь, связанных с простоем и невозможностью использования ИС или ее элементов, несмотря на то, что предпосылки для развития таких методов имеются. В частности, в этих целях могут быть использованы так называемые «производственные функции» [43]. Основные из них, наиболее широко применяемые в практике «экономического моделирования», приведены в таблице 3.1.

Анализ этих функций показывает, что используемые в них коэффициенты влияния разнообразных факторов и условий, во-первых, оказываются для каждой ИС уникальными и, во-вторых, подбираются экспертным путем.

Кроме того, сегодня достаточно сложно, в силу огромного разнообразия вариантов, оценить количественно последствия, вызываемые преднамеренной несанкционированной модификацией данных или прикладных программ.

Таблица 3.1 – Функции, применяемые при экономическом анализе предприятий

Наименование функции	Вид функции
Линейная	$$z_{ks} = \sum_{j \in J(s)} \alpha_{wj} \cdot w_{kj} + \sum_{j \in J(p)} \alpha_{pj} \cdot p_{kj} + \sum_{j \in J(\xi)} \alpha_{\xi j} \cdot \xi_{kj} + \sum_{j \in J(d)} \alpha_{dj} \cdot d_{kj} + \sum_{j \in J(s)} \alpha_{\varphi j} \cdot \varphi_{kj},$$ где z_{ks} – k-я компонента вектора критически важных факторов (автоматизируемых функций на предприятии) в s-й области деятельности предприятия; w_{kj}, p_{kj}, ξ_{kj} – автоматизируемые функции на предприятии, параметры которых характеризуют задействованные ресурсы системного, прикладного программного и аппаратного обеспечения соответственно, безопасность которых может быть нарушена; d_{kj}, φ_{kj} – автоматизируемые функции на предприятии, параметры которых характеризуют данные, обрабатываемые в КС, и охраняемые сведения, конфиденциальность которых должна быть обеспечена, соответственно; $\alpha_{wj}, \alpha_{pj}, \alpha_{\xi j}, \alpha_{dj}, \alpha_{\varphi j}$ – коэффициенты влияния факторов на значение автоматизируемых функций $w_{kj}, p_{kj}, \xi_{kj}, d_{kj}, \varphi_{kj}$ соответственно
Функция Леонтьева	$$z_{ks} = min\left\{\begin{matrix} \frac{w_1^{(k)}}{\alpha_{w,1}},...,\frac{w_{r(w)}^{(k)}}{\alpha_{w,r(w)}},...,\frac{w_{R(w)}^{(k)}}{\alpha_{w,R(w)}}; \frac{p_1^{(k)}}{\alpha_{p,1}},...,\frac{p_{r(p)}^{(k)}}{\alpha_{p,r(p)}},...,\frac{p_{R(p)}^{(k)}}{\alpha_{p,R(p)}}; \frac{\xi_1^{(k)}}{\alpha_{\xi,1}},...,\frac{\xi_{r(p)}^{(k)}}{\alpha_{\xi,r(\xi)}},...,\frac{\xi_{R(\xi)}^{(k)}}{\alpha_{\xi,R(\xi)}}; \\ \frac{d_1^{(k)}}{\alpha_{d,1}},...,\frac{d_{r(d)}^{(k)}}{\alpha_{d,r(d)}},...,\frac{d_{R(d)}^{(k)}}{\alpha_{d,R(d)}}; \frac{\varphi_1^{(k)}}{\alpha_{p,1}},...,\frac{\varphi_{r(\varphi)}^{(k)}}{\alpha_{\varphi,r(\varphi)}},...,\frac{\varphi_{R(\varphi)}^{(k)}}{\alpha_{\varphi,R(\varphi)}}, \end{matrix}\right\};$$ $$r(w) = \overline{1,R(w)}; r(p) = \overline{1,R(p)}; r(\xi) = \overline{1,R(\xi)}; r(d) = \overline{1,R(d)}; r(\varphi) = \overline{1,R(\varphi)}$$
Функция Кобба-Дугласа	$$z_{ks} = R \cdot \prod_{j \in J(w)} \left(w_j^{(k)}\right)^{\alpha_{w,j}} \cdot \prod_{j \in J(p)} \left(p_j^{(k)}\right)^{\alpha_{p,j}} \cdot \prod_{j \in J(d)} \left(d_j^{(k)}\right)^{\alpha_{d,j}} \cdot \prod_{j \in J(\xi)} \left(\xi_j^{(k)}\right)^{\alpha_{\xi,j}} \cdot \prod_{j \in J(\varphi)} \left(\varphi_j^{(k)}\right)^{\alpha_{\varphi,j}},$$ где R – коэффициент размерности, а остальные обозначения те же, что и для линейной функции
Функция «с постоянной эластичностью замещения»	$$z_{ks} = R \cdot \left\{\begin{matrix} \sum_{j \in J(w)} \alpha_{w,j} \cdot \left(w_j^{(k)}\right)^{-\rho} + \sum_{j \in J(p)} \alpha_{p,j} \cdot \left(p_j^{(k)}\right)^{-\rho} + \sum_{j \in J(d)} \alpha_{d,j} \cdot \left(d_j^{(k)}\right)^{-\rho} + \\ + \sum_{j \in J(\xi)} \alpha_{\xi,j} \cdot \left(\xi_j^{(k)}\right)^{-\rho} + \sum_{j \in J(\varphi)} \alpha_{\varphi,j} \cdot \left(\varphi_j^{(k)}\right)^{-\rho} \end{matrix}\right\},$$ где ρ -параметр соответствия, остальные обозначения те же, что и для линейной функции
Квадратичная функция	$$z_{ks} = \gamma_1 \cdot K + \gamma_2 \cdot L + \gamma_3 \cdot K \cdot L - \gamma_4 \cdot K^2 - \gamma_5 \cdot L^2,$$ где L, K -объем приложенного труда и капитала; $\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5$ - коэффициенты пропорциональности
Функция Брауна	$$z_{ks} = \varepsilon_1 \cdot \left[K(t)\right]^{a(t)} \cdot \left[L(t)\right]^{b(t)}.$$ Здесь введена зависимость параметров L и K от времени. $a(t), b(t)$ - коэффициенты «эластичности производства» по отношению к величине капитала и к численности занятых. L и K функции от множества параметров так же, как и для квадратичной функции; ε_1 – коэффициент верификации
Функция Солоу-Хилхорста	$$z_{ks} = R \cdot \left[\varepsilon_1 \cdot K^{-\beta_1} + (1 - \varepsilon_1) \cdot L^{-\beta_2}\right]^{\frac{1}{\beta}},$$ где α_1 – коэффициент верификации; β_1, β_2 - коэффициенты «эластичности производства» по капиталу и труду соответственно. Здесь L и K также функции от множества параметров, как и для квадратичной функции
Функция Тинбергена	$$z_{ks} = R \cdot K^{\alpha} \cdot L^{\beta} \cdot exp(\mu \cdot t),$$ где α, β – темпы прироста производства; μ – параметр прогресса. Здесь L и K функции от множества параметров так же, как и для квадратичной функции

Все это обусловливает то, что вид функционала (3.6) даже только для оценки финансового ущерба остается неопределенным. Тем более данный подход не позволяет непосредственно решить основную проблему – проблему

количественной оценки совокупного ущерба в зависимости от параметров нарушения безопасности информации в ИС (с учетом всех видов ущерба, в том числе тех, которые пока не могут быть перечислены в финансовый ущерб).

Необходимо подчеркнуть, что сложность решения проблемы количественной оценки совокупного ущерба от реализации угроз обусловлена и целым рядом других факторов, таких как:

– отсутствие шкал для количественной оценки некоторых видов ущерба от реализации угроз;

– различие в оценках негативных последствий от реализации одних и тех же угроз на разных предприятиях (в разных организациях), то есть в разных ИС;

– наличие целого ряда так называемых "НЕ-факторов" – неточности, неопределенности, неизвестности и (или) неполноты данных об исследуемых процессах, незамкнутости множества условий, способствующих реализации угроз, нелинейности исследуемых процессов реализации угроз и защиты от них [44].

Все это обусловливает необходимость изыскания такого подхода к определению процедуры оценки ущерба, а затем и риска реализации угроз для информации в ИС, который позволил бы парировать указанные сложности.

При этом важным условием является то, что в случае, если будут построены приемлемые математические модели количественной оценки ущерба от нарушения безопасности информациив типовых ИС или найдены приемлемые альтернативные методы такой оценки, не должны меняться не только базовые показатели риска реализации угроз, но и методы их расчета при возможном изменении алгоритмов расчета частных показателей оценки ущербов.

3.3. Шкалы оценок ущерба, парадигма предельного ущерба и универсальная шкала оценок ущербов

3.3.1. Построение универсальной шкалы оценок ущерба для одной информационной системы

Методологии оценки ущербов от реализации угроз безопасности информации развивались преимущественно в рамках теории оценки рисков. Как правило, в разработанных подходах учитывались:

– важность для пользователя, владельца информации или организации той информации (ее носителей или средств обработки), в отношении которой могут быть выполнены несанкционированные действия при реализации угроз;

– область применения (использования) результатов обработки информации (выходной информации);

– содержание несанкционированных действий и виды нарушений безопасности информации;

– вид ущерба;

– затраты, необходимые на восстановление информации, носителей информации;

– время между реализацией угрозы и фактическим нанесением ущерба (в том числе, когда величина ущерба накапливается со временем) и т.д.

Для оценки величины ущербов в общем случае рассматривалась возможность применения как количественных, так и качественных шкал.

Так, в [45], предложена система шкал, основанная на понятиях эмпирической операции и математической структуры, при этом рассматриваются четыре вида шкал: наименований, порядка, интервалов и отношений. К количественным относятся, как правило, три последних.

Тип количественной шкалы обусловливаются видом функции f, осуществляющей допустимые преобразования $\Psi = f(a)$, где a – оцениваемая величина. Если f – монотонная функция, то соответствующая шкала является шкалой порядка; если f – линейная функция, то соответствующая шкала – это шкала интервалов; если f определяет преобразование подобия, то соответствующая шкала – шкала отношений.

Наиболее приемлемыми для количественной оценки ущербов являются так называемые оппозиционные (полярные) шкалы[12]. Это обусловлено тем, что оппозиционные шкалы основаны на установлении, по крайней мере, двух противоположных (в этом смысле полярных) точек на шкале, определяющих крайние противоположные результаты оценки ущерба (например, отсутствие ущерба и неприемлемый ущерб). На таких шкалах могут рассматриваться несколько типов оценок: точечные количественные, интервальные, вербальные (лингвистические), при этом могут применяться метрические или порядковые шкалы. Для всех остальных шкал, как правило, предполагается построение либо эталонных описаний, либо проведение только попарных сравнений, либо формирование специальных функций уверенности или полезности по определенным аспектам, которые так или иначе необходимо сводить к единой шкале оценок. Применение шкал, не относящихся в полярным, не позволяет переходить к комплексным оценкам ущерба от реализации разнородных угроз.

Построение полярной шкалы оценок ущерба проводится путем формирования дистрибутивной решетки Z с введенными на ней бинарными операциями пересечения ($\wedge$) и объединения ($\vee$), удовлетворяющими требованиям:

идемпотентности ($x \wedge x = x,\ x \vee x = x$);

коммутативности ($x \wedge y = y \wedge x,\ x \vee y = y \vee x$);

ассоциативности ($x \wedge (y \wedge z) = (x \wedge y) \wedge z,\ x \vee (y \vee z) = (x \vee y) \vee z$);

дистрибутивности

$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z),\ x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ [44,45].

[12] Шкала может быть полярной независимо от того, к какому из указанных типов она относится.

На такой шкале аксиоматически задаются два семейства унарных операторов – «утверждения» и «отрицания» следующим образом [45].

Пусть Z – решетка, в которой ζ_{min} – минимальный, а ζ_{max} – максимальный элемент, и класс унарных операторов утверждения $m: \mathrm{Z} \to \mathrm{Z}$ удовлетворяет условиям:

1) $m(\zeta_{min}) = \zeta_{min}; m(\zeta_{max}) = \zeta_{max}$ (*ограниченность*);

2) $\forall x, y \in Z, x \le y \Rightarrow m(x) \le m(y)$ (*монотонность*).

Класс унарных операторов отрицания[13] $n: [0,1] \to [0,1]$ удовлетворяет условиям:

1) $n(\zeta_{min}) = 1, n(\zeta_{max}) = 0$ (*ограниченность*);

2) $\forall x, y \in [\zeta_{min}, \zeta_{max}], x \le y \Rightarrow n(x) \ge n(y)$ (*антитонность*).

Идея использования понятия неприемлемого ущерба достаточно широко распространена на практике. В частности, метод балльных оценок основан на установлении верхней границы оценок, то есть, по сути, на введении предельного ущерба [33, 36].

Особенность шкалы с верхней границей в виде неприемлемого ущерба заключается, прежде всего, в ее универсальности относительно видов ущерба. Кроме того, она является условной, поскольку ее верхняя граница обусловлена суждением о неприемлемости ущерба для данного конкретного потребителя.

В качестве максимального элемента решетки вводится максимально возможный или неприемлемый ущерб, а в качестве минимального – отсутствие какого-либо ущерба, что позволяет построить свою метрическую шкалу для оценки ущербов от реализации угроз безопасности информации *для одной ИС*. Подходы к построению шкалы *для множества ИС* рассматривается в следующем подразделе. Это обусловлено тем, что в разных ИС один и тот же ущерб может оцениваться по-разному.

Важным аспектом построения таких шкал для разнородных ущербов является отсутствие непосредственной возможности применения даже таких простейших операторов, как операторы сравнения и сложения разнородных ущербов. Для этого требуется определить процедуры приведения шкал для разнородных ущербов к единой (базовой) шкале оценок, например, шкале финансового ущерба. Однако сегодня практически отсутствуют такие процедуры пересчета ущерба в какую-либо единую числовую шкалу оценок. Парировать эту сложность можно следующим образом. Наиболее часто при построении такой шкалы используют семь градаций оценок. Формально указанная шкала формируется путем задания показателя C с двумя промежуточными лингвистическими оценками – положительной C^+ и

---

[13] Примером тривиального оператора отрицания [48] является оператор $n(x) = 1 - x$. К нетривиальным операторам отрицания относятся, например, оператор квадратичного отрицания $n(x) = \sqrt{1-x}$ и параметрический оператор отрицания Суджено: $n_\lambda(x) = (1-x)/(1+\lambda \cdot x)$, $\forall x \in [0,1]$, $-1 \le \lambda < \infty$. Примером тривиального оператора утверждения является оператор $m_1(x) = x$.

отрицательной – C^-, а также двумя оценками, усиленными с помощью модификатора v – vC^+ (например, модификатор может иметь значение «очень» или «неприемлемо большой») и vC^- (модификатор может иметь значение «очень», «несущественный» и т.п.), таких, что $C^+ \le vC^+, C^- \ge vC^-$ и двумя оценками, ослабленными с помощью модификатора w (например, модификатора «скорее») wC^+ и wC^- таких, что $C^+ \ge wC^+, C^- \prec wC^-$, а также нейтральной (средней) оценкой C^0 [44]. Градации вербальной шкалы упорядочены следующим образом: $vC^- \le C^- \le wC^0 \le wC^+ \le C^+ \le vC^+$.

После построения базовой вербальной шкалы устанавливается ее соответствие каждой из шкал оценок разнородных ущербов. Это позволяет перевести оценки разнородных ущербов в единую шкалу оценок. Затем базовой вербальной шкале ставится в соответствие нормированная решетка C, то есть замкнутый отрезок [0,1], на который проецируется вербальная шкала, пример проекции приведен на рисунке 3.3.

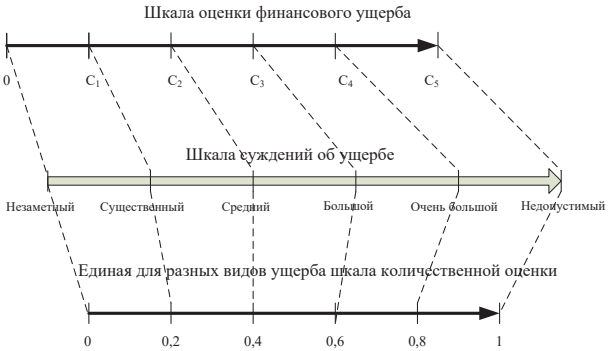


Рисунок 3.3 – Пример приведения разнородных шкал к единой количественной шкале

Таким образом, каждой паре – номеру защищаемого блока информации в ИС и выполняемому относительно него несанкционированному действию (или взаимосвязанному набору таких действий) должен быть поставлен в соответствие определенный ущерб, выраженный в количественной или качественной шкале.

Если имеется количественная шкала и на ней определена величина предельного ущерба, то для приведения к единой шкале оценок величины ущербов необходимо нормировать относительно предельно допустимого ущерба рассматриваемого вида. Так, если ущерб оценивается по вербальной шкале, то на ней определяется точка предельного ущерба и все ущербы выше этой точки рассматриваются как недопустимые.

Далее предельной точке ставится в соответствие верхняя граница числовой шкалы и затем по полученной числовой шкале оценивается данный

вид ущерба. Таким образом по предложенной универсальной шкале может быть оценен ущерб от выполнения как каждого несанкционированного действия, так и совокупности таких действий, выполняемых относительно защищаемого блока информации при реализации множества угроз безопасности информации с различными последствиями. Следует отметить, что сумма ущербов разного вида при реализации совокупности угроз могут значительно превышать предельно допустимый уровень.

Пусть относительно i-го блока защищаемой информации ($i \in I$) может быть выполнено множество $G_i = \{g_i\}$ несанкционированных действий (например, файл сначала копируется, а затем уничтожается), каждое из которых может принести определенный ущерб, оцениваемый по универсальной шкале. Такая оценка представляет собой, по сути, индекс этого ущерба. Тогда суммарный ущерб при воздействии на i-й блок защищаемой информации оценивается следующим образом:

$$\beta_{iG_i} = \begin{cases} 1, \text{если} \sum\limits_{g \in G_i} \beta_{ig} \geq 1, g = \overline{1,\ G_i}\ ; \\ \sum\limits_{g \in G_i} \beta_{ig}, \text{если} \sum\limits_{g \in G_i} \beta_{ig} < 1, \end{cases} \tag{3.8}$$

где β_{ig} – индекс ущерба от реализации g-го деструктивного действия относительно i-го блока защищаемой информации;

G_i – мощность множества несанкционированных дествий, которые могут быть выполнены относительно i-го блока информации.

Для всех блоков защищаемой информации, относительно которых возможна реализация угрозы и выполнение всего множества несанкционированных действие, индекс суммарного ущерба определяется аналогично:

$$\beta_{IG} = \begin{cases} 1, \text{если} \sum\limits_{i \in I} \sum\limits_{g \in G_i} \beta_{ig} \geq 1; \\ \sum\limits_{i \in I} \sum\limits_{g \in G_i} \beta_{ig}, \text{если} \sum\limits_{i \in I} \sum\limits_{g \in G_i} \beta_{ig} > 1. \end{cases} \tag{3.9}$$

Для определения суммарного индекса совокупности разнородных ущербов обладателю может потребоваться учесть разное восприятие им разнородных ущербов. В этом случае следует учесть важность для обладателя разных видов ущерба соответствующим коэффициентом важности V_ϑ вида ущерба ϑ, при этом:

$$\sum_{\vartheta \in \Theta} V_\vartheta = 1,\ \vartheta = \overline{1,\ \Theta}\ , \tag{3.10}$$

где Θ – мощность множества видов ущербов (количество видов ущербов). Тогда суммарный ущерб с учетом важности видов ущерба оценивается сверткой:

$$\beta_\Sigma = \sum_{\vartheta \in \Theta} V_\vartheta \cdot \beta_{IG}^{(\vartheta)}\ , \tag{3.11}$$

где $\beta_{IG}^{(\vartheta)}$ – индекс суммы ущербов ϑ -го вида при выполнения совокупности G несанкционированных действий относительно множества I защищаемых блоков информации.

Для равнозначных шкал функции важности равны 1 и индексы ущерба в формуле (3.11) суммируются обычным образом.

3.3.2. Построение универсальной шкалы оценок ущерба для совокупности информационных систем

Предельный ущерб отражает представление конкретного обладателя информации о важности защищаемой им информации. Однако оценка важности зависит от того, чьи интересы затрагиваются при нанесении ущерба. В связи с этим такая оценка является объективно условной. Для учета и парирования такой условности может быть использован предлагаемый метод взаимосвязанных шкал с их перенормировкой при переходе от одной шкалы к другой. Суть метода сводится к следующему.

Во-первых, для учета уровня пользователя или обладателя информации, чьи интересы затрагиваются при нарушении ее безопасности, вводятся L уровней (градаций) важности, при этом первый уровень соответствует наибольшей, а последний – наименьшей важности. Пример вербальной интерпретации каждого уровня приведен в таблице 3.2.

Таблица 3.2 – Пример вербальной интерпретации уровней важности защищаемой информации и шкал для оценки возможных ущербов

Наименование уровня важности	Номер уровня важности	Краткая характеристика уровня
Федеральный (государственный)	1	Затрагиваются интересы государства, министерств, первых лиц государства. Возможно развитие федеральной чрезвычайной ситуации (ЧС)
Региональный	2	Затрагиваются интересы региона. Возможно развитие региональной ЧС с охватом территории не менее двух субъектов федерации
Территориальный	3	Затрагиваются интересы субъекта федерации. Возможно развитие территориальной ЧС
Местный	4	Затрагиваются интересы города, района, населенного пункта. Возможно развитие местной ЧС
Объектовый, корпоративный	5	Затрагиваются интересы предприятия, организации, фирмы. Возможно развитие объектовой ЧС
Групповой	6	Затрагиваются интересы подразделения (нескольких подразделений) организации, предприятия, первых лиц организации (предприятия).
Персональный (пользовательский)	7	Затрагиваются интересы пользователя (нескольких пользователей)

Для каждого l-го уровня важности строится своя шкала оценок величины возможного ущерба, по которой для пересчитываемого уровня важности m оценивается суммарный индекс ущерба $\beta_{IG}^{(m)}$.

Во-вторых, для каждой шкалы вводится относительный коэффициент ее важности в виде функции $\alpha(m,l)$, лежащий в пределах от 0 до 1, следующим образом:

$$\alpha(m,l) = f(m,l), l = \overline{1,m},\ m = \overline{1,L}.$$

Примеры такой функции приведены в таблице 3.3.

Таблица 3.3 – Примеры функции, определяющей относительную важность шкалы оценок ущерба

Наименование зависимости	Вид функции	Примечания
Линейная	$\alpha(m,l) = \alpha_{min} + \frac{1-\alpha_{min}}{L-l+1} \cdot (L-m+1)$	L - общее количество уровней важности; $\alpha_{\min}$ - коэффициент важности (отношение самого низкого к самому высокому уровню важности); l - номер уровня важности, относительно которого пересчитывается коэффициент опасности для уровня m
Квадратичная	$\alpha(m,l) = a \cdot m^2 + b \cdot m + c,$ $a = \frac{(1-\alpha_{min})}{(L-l)^2};\ b = 2 \cdot \frac{(1-\alpha_{min})}{(L-l)^2} \cdot L;$ $c = \alpha_{min} + (1-\alpha_{min}) \frac{L^2}{(L-l)^2}$	a, b, c – параметры параболы при условии, что $l < L, m \le l$
Степенная	$\alpha(m,l) = \left(\frac{L-m+1}{L-l+1}\right)^n,$ $n = \frac{1}{L-l+1} \cdot ln\left(\frac{1}{\alpha_{min}}\right)$	Обозначения те же, что и для линейной функции
Показательная	$\alpha(m,l) = c \cdot e^{b \cdot m},$ $b = \frac{\alpha_{min}}{L-l};\quad c = e^{-\frac{\alpha_{min} \cdot l}{L-l}}$	b, c - параметры показательной функции при условии, что $l < L, m \le l$

С использованием относительных коэффициентов важности шкала с уровнем меньшей важности может быть нормирована относительно шкалы с уровнем большей важности, то есть оценка индекса ущерба $\beta_{IG}^{(m)}$ по шкале с пересчитываемым уровнем важности m может быть пересчитана на значение индекса $\beta_{IG}^{(l)}$ по шкале с уровнем важности l, при этом:

$$\beta_{IG}^{(l)} = \alpha(m,l) \cdot \beta_{IG}^{(m)} \quad . \tag{3.12}$$

Таким образом, приведенные соотношения позволяют нормировать шкалы, построенные на нижнем уровне важности относительно шкал с более высоким уровнем важности и тем самым парировать условность введения предельно допустимого уровня ущерба.

Достоинствами предложенного подхода являются:

– инвариантность к видам шкал, по которым оцениваются парциальные разнородные ущербы от реализации угроз безопасности информации;

– возможность приведения шкал оценки принципиально разнородных ущербов к единой шкале оценок;

– возможность применения нечетких оценок возможного ущерба от реализации угроз безопасности информации;

– парирование сложности учета разных оценок одного и того же ущерба разными пользователями за счет введения операции нормирования по условному предельно допустимому ущербу;

– возможность оценки суммарного ущерба за счет введения операций суммирования разнородных ущербов с использованием весовых функций каждого вида ущерба.

3.4. Аппарат нечетких множеств и его применение для оценки ущерба от реализации угроз безопасности информации

При вербальных суждениях о размерах ущерба от реализации угроз могут быть достаточно большие отклонения, приводящие к ошибкам, возможно значительным, при оценке опасности угроз и оценке эффективности защиты информации. Применение же статистических методов оценки неопределенных (случайных) размеров ущерба, несмотря на значительно большую проработанность, часто сдерживается из-за невозможности учета огромного разнообразия существенных и при этом слабо формализуемых факторов и отсутствия необходимой статистики.

Для парирования этих сложностей можно использовать два приема. Первый из них основан на составлении вариационного ряда (ряда предпочтений) для последствий реализации угроз безопасности информации. Это позволяет определить весовые коэффициенты для каждого последствия нарушений. В случае отсутствия каких-либо оснований (сведений) для выявления весового коэффициента можно использовать известное правило Фишберна [46], в соответствии с которым, если на множестве рассматриваемых последствий от реализации угроз безопасности установлены отношения предпочтения $r_1 \geq r_2 \geq \ldots \geq r_N$, то значимость n-го ($n = \overline{1,N}$) последствия определяется из соотношения:

$$r_n = \frac{2 \cdot (N - n + 1)}{(N + 1) \cdot N}, \tag{3.13}$$

что соответствует максимуму неопределенности сведений о последствиях.

Введенная таким образом система весовых коэффициентов на всем множестве последствий реализации угроз безопасности информации позволяет построить простую шкалу оценок опасности последствий следующим образом:

$$\beta_n = \frac{r_n}{max(r_n)} = 1 - \frac{n-1}{N}. \tag{3.14}$$

Второй прием основан на нечетких оценках индексов ущербов, формируемых по универсальной шкале оценок (см. раздел 3.3), то есть с использованием аппарата нечетких множеств.

Начало развитию теории нечетких множеств положила статья профессора из США Лотфи Заде в 1965 г. «Fuzzy Sets» («Нечеткие множества»). Дальнейшее развитие его идей привело не только к существенному пересмотру основ математики, появлению новых ее разделов, таких как теория нечетких чисел, нечеткая логика, теория нечетких систем и др., но и к широкому практическому применению. Лидерство здесь захватила Япония, где нечеткую математику стали применять в системах управления движением поездов метрополитена и механических транспортных средств, в водоочистительных сооружениях, в химических реакторах, в сталеплавильных печах, в оборудовании для металлообработки и т.д. Применительно к теории защиты информации вопросы применения этого аппарата начали решаться, в основном, в начале 21-го века в целом ряде работ [24, 33, 47–49]. Рассмотрим некоторые базовые понятия этой теории.

Нечетким множеством A, определенным на некоторой числовой предметной области X, называется множество пар:

$$A = \{\mu_A(x), x\}, \forall x \in X, \tag{3.15}$$

где $\mu_A(x)$ – функция принадлежности (ФП) множеству A каждого элемента $x \in X$, $\mu_A(x) \in [0,1]$.

Значение ФП называют степенью принадлежности. ФП может задаваться в виде графика (рисунок 3.4), формулы, таблицы, вектора степеней принадлежности и даже суммы или интеграла.

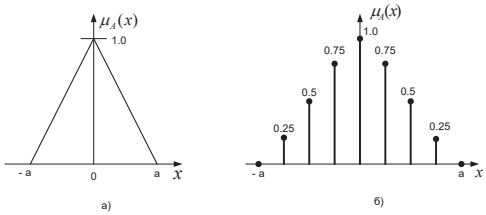


Рисунок 3.4 – Графическая форма представления непрерывной (а) и дискретной (б) функции принадлежности нечеткого числа «примерно ноль»

Рассмотрим для примера нечеткое множество чисел, которое характеризуется понятием «приблизительно 6»

$$A = \left\{ \frac{0}{3}; \frac{0.4}{4}; \frac{1}{6}; \frac{0.8}{7}; \frac{0.3}{8} \frac{0.1}{9}; \frac{0}{10} \right\}.$$

Для этого описания числа 0 и 10 не относятся к нечеткому множеству A. Хисло 6, несомненно, к нему относится, числу 4 соответствует степень принадлежности 0.4, числу 7 – степень 0.8, числу 8 – степень 0.3, а числу 9 – степень 0.1.

Допустимый диапазон значений ФП может не ограничиваться интервалом от 0 до 1. Теоретико-множественные операции над ним не выводят

значение функции принадлежности за пределы этого интервала, однако при выполнении арифметических операций степени принадлежности могут оказаться больше 1. В связи с этим применяют операцию нормирования путем деления исходной ФП на ее максимальное значение $\sup_x \mu_A(x)$, при этом ФП результирующего множества A_N будет принимать значения в интервале [0,1]:

$$\mu_{A_N} = \mu_A(x) / \sup_x \mu_A(x). \tag{3.16}$$

Сегодня применяются весьма разнообразные ФП, множество которых можно разделить на две группы.

К первой группе относятся кусочно-линейные или иногда называемые многоугольными ФП [25], то есть функции, состоящие из прямолинейных участков (рисунок 3.5).

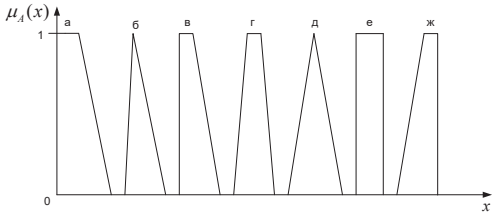


Рисунок 3.5 – Формы используемых на практике кусочно-линейных функций принадлежности:
а – крайняя левая ФП; б – асимметричная треугольная ФП;
в – асимметричная трапециевидная ФП; г – симметричная nрапециевидная ФП; д – симметричная треугольная ФП;
е – прямоугольная ФП; ж – крайняя правая ФП

Существенным преимуществом многоугольных ФП является то, что для их определения требуется наименьший по сравнению с другими функциями объем информации, который ограничивается данными об угловых точках.

Чтобы определить кусочно-линейную (многоугольную) ФП, достаточно задать лишь модальное значение соответствующего нечеткого множества. Для записи математического выражения многоугольной функции используют логические переменные $w_i : \{0,1\}$.

Например, в случае в случае трапециевидной ФП вводятся следующие логические переменные:

$$w_1 = \begin{cases} 1 \text{ для } a \le x \le b, \\ 0 \text{ в другихслучаях;} \end{cases} \quad w_2 = \begin{cases} 1 \text{ для } b \le x \le c, \\ 0 \text{ в других случаях;} \end{cases}$$

$$w_3 = \begin{cases} 1 \text{ для } c \le x \le d, \\ 0 \text{ в других случаях,} \end{cases} \tag{3.17}$$

а ФП имеет вид

$$\mu(x) = w_1\left(\frac{x-a}{b-a}\right) + w_2 + w_3\left(\frac{d-x}{d-c}\right). \qquad (3.18)$$

В случае симметричной треугольной функции требуется ввести только одну логическую переменную

$$w_1 = \begin{cases} 1 \text{ для } (e-a) \le x < (e+a), \\ 0 \text{ в других случаях,} \end{cases} \qquad (3.19)$$

а ФП записывается следующим образом:

$$\mu(x) = w\left(\frac{a-|x-e|}{a}\right), \qquad (3.20)$$

где e – значение x для биссектрисы симметричной треугольной ФП.

Недостатком кусочно-линейных ФП является то, что они не являются непрерывно дифференцируемыми. Это может приводить к усложнению процесса адаптации нечетких моделей [25].

Ко второй группе относятся так называемые «интуитивные» функции принадлежности, которые обладают рядом свойств, определяемых аксиомами Шваба [25]:

аксиома 1: интуитивные ФП $\mu(x)$ являются непрерывными на всей области определения X. Выражаемая человеком качественная оценка параметра x не изменяется скачкообразно ни при каком достаточно малом изменении ее значения;

аксиома 2: первая производная интуитивной ФП $\mu'(x) = \dfrac{d\mu(x)}{dx}$ является непрерывной на всей области определения X, то есть скорость изменения оценки параметра x не меняется скачкообразно при любых малых изменениях самого параметра;

аксиома 3: вторая производная интуитивной ФП $\mu''(x) = \dfrac{d^2\mu(x)}{dx^2}$ является непрерывной на всей области определения X;

аксиома 4: интуитивная ФП имеет минимальное значение кривизны.

Это утверждение означает, что из множества возможных ФП человек выбирает ту, для которой значение максимума второй производной является минимальным среди всех таких функций, то есть

$$\mu(x) : \mu(x) = \arg\min_{\mu}\left|\max\left(\frac{d^2\mu(x)}{dx^2}\right)\right|. \qquad (3.21)$$

Примеры некоторых интуитивных ФП и их краткая характеристика приведены в таблице 3.4.

Сегодня известно достаточно много способов формирования ФП, краткая характеристика некоторых из них приведена в таблице 3.5.

Таблица 3.4 – Примеры интуитивных функций принадлежности

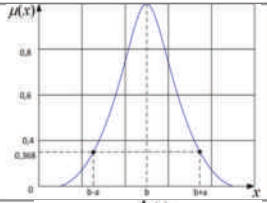
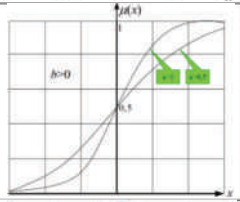
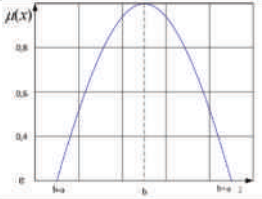
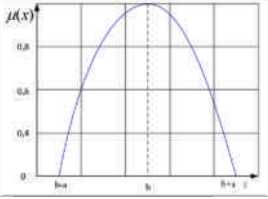
Наименование функции принадлежности	Аналитический вид функции принадлежности	Графический вид функции принадлежности	Недостатки функции принадлежности	Достоинства функции принадлежности
Симметричная гауссова функция	$\mu(x) = \exp\left\{-\frac{(x-b)^2}{2a^2}\right\}$		1. Для ее использования необходимо задание большего, чем, например, для треугольной функции числа параметров, что затрудняет настройку нечеткой модели. 2. Затрудняет получение простых локально линейных поверхностей отклика нечеткой модели	1. Обеспечивают получение гладких, непрерывно дифференцируемых гиперповерхностей отклика нечеткой модели. 2. Дают возможность аналитического анализа нечетких систем
Сигмоидальная функция принадлежности	$\mu(x) = \frac{1}{1+\exp\{-a\cdot(x-b)\}}$		Те же	Те же
Гармоническая функция принадлежности	$\mu(x) = \begin{cases} 0.5\cdot\left[1+\cos(\pi\cdot\frac{x-b}{a})\right] & для\ (b-a)<x\le(b+a); \\ 0 & в\ других\ случаях. \end{cases}$		1. Гармоническая функция имеет ограниченный носитель $[(b-a),(b+a)]$, что позволяет задавать ее параметры экспертным путем. 2. Она хорошо согласуется с аксиомами Шваба [25]	При неравномерном расположении функции принадлежности на области определения симметричность гармонической функции отрицательно сказывается на качестве моделирования на участках, слабо охватываемых функцией принадлежности
Полиномиальная функция принадлежности	$\mu(x) = \begin{cases} 0 & если\ x<(b-a); \\ 1-\left(\frac{x-b}{a}\right)^2 & если\ (b-a)\le x<(b+a); \\ 1 & если\ x>b \end{cases}$		1. Функция является непрерывно дифференцируемой и поэтому более гладкой, чем треугольная. 2. Параметры a и b легко задаются экспертным путем	Функция не удовлетворяет в должной мере аксиомам Шваба: ее производная не обращается в ноль в конечных точках носителя

Таблица 3.5 – Краткая характеристика некоторых методов построения функций принадлежности

Наименование метода	Содержание методы	Краткая характеристика метода
Метод опроса	В группе из N экспертов на вопрос о принадлежности значения характеристики (параметра, функции) к рассматриваемому множеству (например, интервалу значений) N_+ дает положительный ответ, а N_- – отрицательный. При этом функция принадлежности определяется как $\mu_j(x) = N_+ / (N_+ + N_-)$	Метод позволяет сформировать практически все известные классы функций принадлежности: нормальные, субнормальные, выпуклые и невыпуклые, унимодальные и полимодальные, дискретные, непрерывные и непараметрические [25]
Метод усреднения	Каждый эксперт указывает значение функции принадлежности $\mu_{ji}(x)$ для запрашиваемого значения нечеткого параметра x и затем вычисляется среднее для всех экспертов значение функции принадлежности, то есть: $\mu_j(x) = \frac{1}{N}\sum_{n=1}^{N}\mu_{jn}(x)$	То же
Метод частоты использования лингвистического терма	В качестве функции принадлежности принимается оценка частоты использования экспертами значения лингвистической переменной. Составляется таблица значений лингвистической переменной (например, значений вида: очень малый, малый, средний, большой, очень большой) и оценивается частота ответов экспертов с этими значениями. Значения оцениваемого параметра рассматриваются на множестве выделенных интервалов. Функция принадлежности оценивается по формуле $\mu_j(x) = \frac{1}{N}\sum_{n=1}^{N}\mu_{jn}(x)$, где J - количество интервалов значений нечеткой величины x (как правило, одинаковое для всех термов); I - количество термов; c_{ij} - значение частоты использования экспертами i-го терма для интервала значений j	Могут быть построены функции принадлежности: нормальные, выпуклые, унимодальные и полимодальные, дискретные, непрерывные и непараметрические
Метод парных сравнения с определением среднегеометрического значения	Функции принадлежности определяются путем расчета среднегеометрического значения коэффициентов парных сравнений: $\mu_j(x_i) = w_i = \sqrt[J]{\prod_{j=1}^{J} c_{ij}}\,;\ j = \overline{1,J}$	То же
Метод количественного парного сравнения (метод Саати)	Формируется матрица значений значимости оцениваемого параметра по установленной шкале значимости: $C = \|c_{ij}\|$ и находится собственный вектор матрицы $w = (w_1, w_2, \ldots, w_k)$ путем решения уравнения $Cw = dw$, где d - собственное значение матрицы. Функция принадлежности определяется как $\mu_j(x_i) = w_i$	Могут быть построены функции принадлежности: субнормальные, выпуклые, унимодальные и полимодальные, дискретные, непрерывные и непараметрические. С увеличением количества сравниваемых величин существенно возрастает размерность матрицы
Метод сравнения с нахождением частного	Функции принадлежности определяются по коэффициентам парных сравнений следующим образом: $\mu_j(x_i) = c_{ij} \Big/ \sum_{j=1}^{J} c_{ij},\ i = \overline{1,I},\ j = \overline{1,J}$	То же

Наименование метода	Содержание методы	Краткая характеристика метода
Метод парных сравнений на ранговых оценках	Функции принадлежности определяются по коэффициентам парных сравнений следующим образом: 1) определяется лингвистическая переменная и множество, на котором она задается; 2) определяется совокупность нечетких термов $\{T_1, T_2, ..., T_l\}$; 3) формируется для каждого терма T_j матрица: $A = \begin{vmatrix} 1 & \frac{r_2}{r_1} & \frac{r_3}{r_1} & \dots & \frac{r_l}{r_1} \\ \frac{r_1}{r_2} & 1 & \frac{r_3}{r_2} & \dots & \frac{r_l}{r_2} \\ \dots & \dots & \dots & \dots & \dots \\ \frac{r_1}{r_l} & \frac{r_2}{r_l} & \frac{r_3}{r_l} & \dots & 1 \end{vmatrix}$, $r_i = r(x_i)$ – ранг элемента x_i, который характеризует значимость этого элемента в формировании свойств, описываемых термом; 4) вычисляются функции принадлежности для любого терма: $\mu_j(x_i) = \left(\sum_{k=1}^{l} \frac{r_k}{r_i}\right)^{-1}$	Преимущества метода – не требуется решение характеристических уравнений. Могут быть построены функции принадлежности: нормальные, выпуклые, унимодальные и полимодальные, дискретные, непрерывные и непараметрические
Метод назначения	Функция принадлежности для треугольного нечеткого числа формируется путем определения названия параметра, диапазона возможных по мнению эксперта значений, количества лингвистических термов, с помощью которых оценивается параметр, следующим образом: $\mu_f(x) = \begin{cases} 0, \text{ если } x < x_{min}; \\ (x - x_{min})/(b - x_{min}), \text{если } x_{min} \le x \le b; \\ (x_{max} - x)/(x_{max} - b), \text{ если } b \le x \le x_{max}; \\ 0, \text{ если } x > b. \end{cases}$ b - значение функции принадлежности равна 1	Могут быть построены функции принадлежности: нормальные, выпуклые, унимодальные, непрерывные и непараметрические
Метод экспоненциальной функции	Функция принадлежности для нечеткого числа, которое приблизительно равно f формируется в виде экспоненциальной функции $\mu_f(x) = e^{-\alpha(f-x)^2}$, где α – вспомогательный коэффициент, $\alpha = -\frac{4}{\varepsilon^2} ln(0.5)$; ε - расстояние между точками, в которых функция $\mu_k(x)$ приобретает значение 0.5 (этот параметр определяют экспертно по специальному алгоритму [25])	Могут быть построены функции принадлежности: нормальные, выпуклые, унимодальные, непрерывные и параметрические
Метод интервальных оценок	Метод расчета основывается на том, что для ответов экспертов устанавливается интервал возможных значений показателя выбора (например, количества баллов, которые могут быть даны экспертом при ответе на вопрос) $[y_{min}, y_{max}]$, отвечающий пожеланию эксперта выбрать параметр с заданным значением определяющего его терма (например, "высокий", "средний" и т.д.), при этом, если y_x - ответ эксперта, то функция принадлежности находится по формуле: $\mu_f(x) = \begin{cases} 0, \text{ если } y_x \le y_{min}; \\ \frac{y_x - y_{min}}{y_{max} - y_{min}}, \text{если } y_{min} < y_x < y_{max}; \\ 1, \text{ если } y_x \ge y_{max} \end{cases}$	Наличие трех значений функции принадлежности (указанное экспертом, 0 и 1) позволяют определить ее в интервале $[y_{min}, y_{max}]$. Может быть построены функции принадлежности любого из указанных ранее классов

Важным для практики аспектом применения теории нечетких множеств для оценки размеров ущерба является то, что оцененные с использованием универсальной шкалы нечеткие индексы разнородных ущербов могут складываться.

Индекс суммарного ущерба в этом случае будет тоже нечетким числом.

Его функция принадлежности определяется последовательной попарной операцией с функциями принадлежности двух нечетких слагаемых по формуле [23–25]:

$$\mu_{\beta_i \oplus j}(x) = \bigvee_{\beta_i + \beta_j} \left[\mu_{\beta_i}(x) \wedge \mu_{\beta_j}(x) \right] = \max_{\beta_i + \beta_j} \min_{\mu} \left[\mu_{\beta_i}(x), \mu_{\beta_j}(x) \right]; i, j = \overline{1, N}, i \neq j, \quad (3.22)$$

где $\vee$ и $\wedge$ – операторы объединения и пересечения множеств,
или

$$\mu_{\beta_i \oplus j}(x) = \mu_{\beta_i} + \mu_{\beta_j} - \mu_{\beta_i} \cdot \mu_{\beta_j}; i, j = \overline{1, N}, i \neq j. \quad (3.23)$$

Наиболее просто использовать для оценки индексов ущербов треугольные нечеткие числа [23, 25]. В этом случае ФП представляется в виде трех чисел, а само нечеткое числе – следующим образом:

$$\beta_\Sigma = \left(\frac{0}{a}, \frac{1}{b}, \frac{0}{c} \right).$$

При суммировании, например, двух чисел β_1 и β_2, если считать компонентами первого и второго суммируемых чисел (a_1, b_1, c_1) и (a_2, b_2, c_2), то компоненты суммарного числа рассчитываются следующим образом:

$$a_\Sigma = a_1 + a_2, b_\Sigma = b_1 + b_2, c_\Sigma = c_1 + c_2. \quad (3.24)$$

После дефаззификации нечеткого числа β_Σ по формуле [25]:

$$\beta_\Sigma = \left(\int_{X_\mu} x \cdot \mu_{\beta_\Sigma}(x) dx \right) \Bigg/ \left(\int_{X_\mu} \mu_{\beta_\Sigma}(x) dx \right), \quad (3.25)$$

где X_μ – область определения функции принадлежности $\mu_{\beta_\Sigma}(x)$, можно получить четкие значения индекса суммарного ущерба.

Например, для нечетких индексов парциальных ущербов $\left(i = \overline{1, I}\right)$ в виде треугольных чисел $\beta_i = (a_i, b_i, c_i)$ четкое значение индекса суммарного ущерба с учетом формул (3.24) и (3.25) после дафаззификации определяется из соотношения:

$$\beta_\Sigma = \frac{1}{3} \cdot (a_\Sigma + b_\Sigma + c_\Sigma) = \frac{1}{3} \cdot \sum_{i=1}^{I} (a_i + b_i + c_i). \quad (3.26)$$

Пусть треугольные числа для индексов парциальных ущербов формируются E экспертами с некоторым случайным разбросом каждого значения нечеткого индекса $\left(a_i^e, b_i^e, c_i^e\right), i = \overline{1, I}, e = \overline{1, E}$.

При этом распределение значений статистических независимых нечетких индексов подчинено нормальному закону со средними значениями $\left(\overline{a}_i^e, \overline{b}_i^e, \overline{c}_i^e\right)$ и среднеквадратическими отклонениями $\sigma\left(a_i^e\right), \sigma\left(b_i^e\right), \sigma\left(c_i^e\right)$.

Тогда среднее значение суммарного ущерба после дефаззификации может быть определено по формуле, аналогичной формуле (3.26):

$$\overline{\beta_\Sigma} = \frac{1}{3}\sum_{e=1}^{E}\sum_{i=1}^{I}\left(\overline{a}_i^e + \overline{b}_i^e + \overline{c}_i^e\right), \tag{3.27}$$

а его среднеквадратическое отклонение – следующим образом:

$$\overline{\sigma_\Sigma} = \frac{1}{3}\cdot\sqrt{\sum_{e=1}^{E}\sum_{i=1}^{I}\left(\sigma_a^2 + \sigma_b^2 + \sigma_c^2\right)}, \tag{3.28}$$

где $\sigma_a \equiv \sigma\left(a_i^e\right), \sigma_b \equiv \sigma\left(b_i^e\right), \sigma_c \equiv \sigma\left(c_i^e\right)$.

В гипотетическом случае, когда все среднеквадратические отклонения по индексам парциальных ущербов примерно одинаковы и равны σ_I^E, формула (3.28) преобразуется к виду:

$$\sigma_\Sigma = \sigma_I^E\sqrt{\frac{E\cdot I}{3}}. \tag{3.29}$$

Приведенное соотношение позволяет ориентировочно оценить влияние разброса мнений экспертов о размерах парциальных ущербов на оценку индекса суммарного ущерба.

4. МОДЕЛИ ПРОЦЕССОВ РЕАЛИЗАЦИИ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ С ДИСКРЕТИЗАЦИЕЙ СОСТОЯНИЙ БЕЗ УЧЕТА ФАКТОРА ВРЕМЕНИ

4.1. Классификация моделей процессов реализации угроз безопасности информации от несанкционированного доступа

Моделирование процессов реализации угроз проводится при оценивании возможностей такой реализации в интересах выявления актуальных угроз и обоснования выбора адекватных мер защиты от них [10–13, 26]. Вместе с тем такое оценивание является, во-первых, многоаспектным и многоэтапным, а, во-вторых, качественным (экспертным) или количественным (с применением количественных методов).

Рассматривая многоаспектность оценки возможностей реализации угроз, необходимо отметить, что такая оценка, как правило, включает в себя решение комплекса взаимосвязанных задач анализа, в том числе таких как:

выявление наличия, расположения в ИС и определение категории (важности, грифа конфиденциальности и др.) защищаемой информации, относительно которой рассматривается комплекс угроз;

выявление состава и расположения источника каждой угрозы;

выявление уязвимостей в системном и прикладном ПО, в настройках, архитектуре построения и применяемых в ИС информационных технологиях, которые могут быть использованы для реализации угроз;

определение способов реализации угроз, связанных с проникновением в операционную среду ИС;

определение способов реализации угроз, связанных с перехватом трафика ИС и всех ее хостов;

оценка возможностей преодоления мер защиты, уже принятых в ИС и ее хостах;

выявление возможности реализации несанкционированных действий относительно защищаемой информации в ИС (копирования, модификации, блокирования, перемещения, уничтожения и т.д.).

Содержание указанных аспектов анализа обусловливает то, что для решения некоторых задач анализа, как правило, не требуется разработка каких-либо моделей (например, при выявлении защищаемой информации, за исключением, возможно, случаев создания средств поиска и мониторинга защищаемых информационных ресурсов, при выявлении состава и расположения источников угроз) или не требуется учет фактора времени (например, при выявлении уязвимостей), а для других задач без наличия таких моделей оценка возможностей реализации угроз оказывается или необоснованной, а часто и некорректной, или отражающей некоторые частные вопросы анализа (например, оценка принципиальной возможности доступа к защищаемому файлу или каталогу).

Следует подчеркнуть, что сегодня моделирование процессов реализации угроз безопасности информации в ИС в целом или парциальных процессов,

анализ которых необходим для оценки возможностей реализации угроз, применяется или должно применяться при решении большинства задач ТЗИ.

Модель процесса реализации угрозы безопасности информации в ИС – это формальное его представление, позволяющее по определенным показателям и критериям провести оценку возможности реализации угрозы безопасности информации. Сегодня известно большое количество методов и подходов к разработке таких моделей, однако многие из них или направлены на достижение весьма узких целей моделирования (например, на доказательство принципиальной возможности доступа нарушителя к тому или иному блоку информации), или недостаточно состоятельны, так как не учитывают ряд весьма существенных факторов, влияющих на возможности реализации угроз (например, фактор времени).

Общая классификация моделей, встречающихся в практике исследований в области ТЗИ или которые, на наш взгляд, могли бы быть использованы в этой области, приведена на рисунке 4.1. Краткая характеристика указанных моделей сводится к следующему.

Для моделирования процессов реализации угроз безопасности информации или составляющих их процессов (при решении частных задач анализа) применяются как аналитические, так и имитационные модели. Вместе с тем, что в связи со значительной трудоемкостью разработки и узкой, как правило, направленностью имитационным моделям уделяется значительно меньше внимания. Это в полной мере относится к моделям, построенным с использованием аппарата сетей Петри, в том числе стохастических сетей Петри (см. раздел 4.5). К другим имитационным моделям могут относиться, например, имитационные модели на основе комбинаторных методов, эпидемиологические модели (как стохастические, так и нестохастические), функциональные модели (с моделированием процессов выполнения определенного набора функций) и т.п.

И аналитические, и имитационные модели могут быть как детерминированными (не учитывающими случайные факторы, которые могут иметь место в моделируемом процессе) и стохастическими (учитывающими случайные факторы).

В предметной области защиты информации детерминированные модели применяются, как правило, для решения достаточно узких задач. Например, к таким моделям относятся формальные модели безопасности [50, 51], логико-лингвистические модели и модели ситуационного управления [52–54], модели, основанные на исчислении высказываний и теории предикатов [54–56] и др. Более подробно характеристика некоторых из этих моделей применительно к решению задач анализа возможностей реализации угроз безопасности информации в ИС дается в последующих разделах.

Наконец, важным фактором, учет которого принципиально меняет содержание моделей и применяемые при этом методы, является время. При этом необходимо иметь в виду следующее.

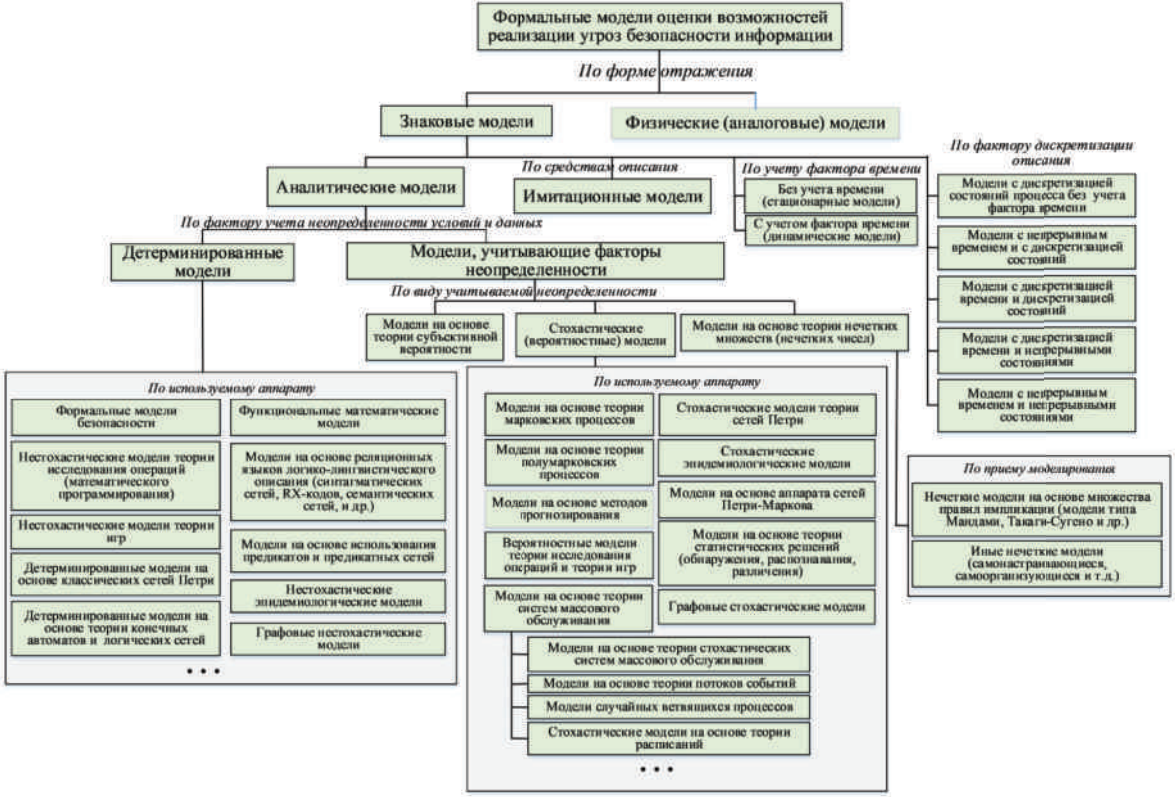


Рисунок 4.1 – Классификация моделей, которые находят или могут найти применение для оценки возможностей реализации угроз безопасности информации от НСД в ИС

В зависимости от того, непрерывное или дискретное множество представляет собой область значений моделируемого процесса и область определения времени, различают 4 вида процессов (рисунок 4.2) [57]: дискретно-дискретный, дискретно-непрерывный, непрерывно-дискретный и непрерывно-непрерывный.

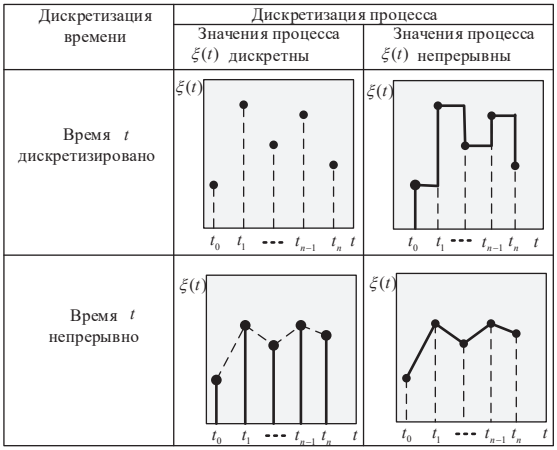


Рисунок 4.2 – Виды моделируемых процессов в зависимости от наличия дискретизации времени и значений процесса

Кроме приведенных на рисунке 4.2 видов процессов, могут быть также сложные, смешанные виды случайных процессов, например, когда значения процесса зависят не только от времени, но и от других, возможно случайных, параметров, которые могут быть как непрерывнозначными, так и дискретными [57].

Дискретные значения моделируемого процесса, как правило, определяются в соответствии с семантически значимыми его состояниями.

Поэтому такие модели часто называют моделями с дискретным временем и дискретными состояниями. К ним относятся, в частности, модели, основанные на теории конечных автоматов, некоторые логико-лингвистические модели, модели на основе логических сетей и сетей Петри (см. раздел 5) и др.

В ряде задач моделирования учет времени оказывается излишним, не влияющим на результаты моделирования.

Так, в формальных моделях безопасности [50, 51] выявляется принципиальная возможность несанкционированного доступа к защищаемой информации (к каталогу, к файлу, к записи в базе данных и т.п.), возможный путь или/и возможность пресечения такого доступа. В графовых моделях угроз сетевых атак формируются возможныепути распространения сетевых

атак (траектории атак) в интересах обоснования возможности реализации и вероятных сценариев атак, а также для обоснования состава мер и целесообразных мест установки в ИС средств защиты [58–62]. При этом временной фактор игнорируется.

Вместе с тем при решении большинства задач такое игнорирование невозможно, так как приводит к некорректным выводам. Как правило, при отсутствии учета фактора времени применяется дискретизация состояний моделируемого процесса.

Иногда для учета фактора времени дискретные состояния привязываются к тактам времени, однако такой учет времени является условным, так как корректное определение длительности тактов определяет необходимость разработки соответствующих моделей процессов, реализуемых в пределах каждого такта, и неизбежно приводят к применению иных методов моделирования.

4.2. Формальные модели безопасности и их применение при оценке возможностей реализации угроз безопасности информации

При анализе угроз безопасности информации в ИС важным аспектом является доказательство принципиальной возможности при их реализации доступа к защищаемой информации. В ряде случаев, особенно при применении мер и средств защиты, такая возможность может вообще отсутствовать. Это привело к необходимости поиска таких доказательств на основе формальных моделей безопасности. Количество таких моделей весьма велико. Как отмечено в [50], это обусловлено тем, что такие модели могут быть использованы:

– для анализа безопасности существующих и перспективных ИС, особенно в случаях, когда требуется получение гарантий защищенности обрабатываемой в них информации;

– как основа для разработки более совершенных моделей, позволяющих более точно описывать и исследовать механизмы защиты информации в современных ИС.

Формальные модели разрабатывались для различных «политик управления доступом» [50, 51], к основным из которых относятся:

дискреционная политика, основными формальными моделями безопасности для которой являются модель управления доступом Харрисона-Руззо-Ульмана, модель Take-Grant, модель типизированной матрицы доступа (модель ТМД);

мандатная (полномочная) политика, основными формальными моделями безопасности для которой являются модель Белла-ЛаПадуллы, мандатные ДП-модели (модели Девянина П.Н. [50]), модель безопасности переходов и т.п.;

политика ролевого управления доступом, основными формальными моделями безопасности для которой являются базовая ролевая модель, мандатная ролевая модель и модель ролевого администрирования.

Кроме этого, сегодня определенное внимание уделяется также политике безопасности информационных потоков и политике изолированной программной среды.

Ниже дается краткая характеристика некоторых из указанных моделей, а также модели Кларка-Вилсона и модели Биба для обеспечения целостности информации.

В основе подавляющего большинства формальных моделей безопасности лежит теория детерминированных конечных автоматов. Использование теории конечных автоматов для анализа защищенности автоматизированных систем обусловлено некоторой схожестью данной проблематики с проблемами, решаемыми в теории надежности, где модели на основе конечных автоматов достаточно широко использовались и используются при синтезе таких систем, обосновании их функциональной надежности и др. В формальных моделях безопасности теория конечных $A=\{a_v\}, v=\overline{1,V},$ автоматов используется без учета фактора времени для выявления возможных состояний ИС, в которых могут иметь место нарушения безопасности обрабатываемой в ИС информации, хотя в самой теории конечных автоматов время рассматривается как важный параметр [52, 63–66].

Конечным автоматом называется совокупность пяти объектов:

1) конечного непустого множества входных данных $A=\{a_v\}, v=\overline{1,V},$ называемого входным алфавитом автомата, а его элементы называют входными символами-буквами;

2) конечного непустого множества выходных данных $B=\{b_l\}, l=\overline{1,L}$, называемого выходным алфавитом автомата, а его элементы называют выходными символами-буквами;

3) конечного непустого множества $S=\{s_z\}, z=\overline{1,Z}$ состояний автомата;

4) переходной функции состояний f, отображающих множество всех упорядоченных пар a_v, s_z в множество S;

5) выходной функции g, отображающих множество всех упорядоченных пар a_v, s_z в множество B.

В классических формальных моделях безопасности, основанных на теории конечных автоматов, постулируется, что все возможные нарушения безопасности информации могут относиться к одной из трех групп: нарушения конфиденциальности, нарушения целостности и нарушения доступности. Эту триаду впервые предложили использовать Зальцер и Шредер в 1975 г. Для ссылки на триаду – конфиденциальность, целостность, доступность иногда используют аббревиатуру КЦД [51]. Как правило, в формальных моделях используется детерминированный автомат, в котором переходная функция f

является детерминированной. Однако возможно использованием и стохастических автоматов, когда указанная функция является вероятностной.

Автомат может находиться только в одном из своих возможных состояний, каждое из которых определяется конкретным сочетанием элементов его описания. Количество таких сочетаний конечно и, в общем случае, среди них есть как безопасные, так и небезопасные с точки зрения возможности реализации угроз. Состояние считается безопасным, если в нем не нарушаются требования политики безопасности. Если начальное состояние автомата является безопасным и функция его переходов такова, что при поступлении любых возможных входных воздействий осуществляется переход в безопасное состояние, то такой автомат моделирует безопасную систему.

Исторически первыми появились формальные модели для дискреционного принципа управления доступом, наиболее известной из которых стала *модель Харрисона-Руззо-Ульмана.* Она формально описывает управление доступом субъектов к объектам системы, то есть к файлам, каталогам, директориям, полям и записям в базах данных и т.п., и контроль распространения прав доступа (сокращенно ее называют «модель ХРУ»). В рамках этой модели система обработки информации представляется в виде совокупности активных сущностей – субъектов (множество S), пассивных сущностей – объектов (множество O), содержащих защищаемую информацию, и конечного множества прав доступа $R=\{r_1,r_2,...,r_n\}$, означающих полномочия на выполнение соответствующих действий. При этом принято считать, что все субъекты одновременно являются объектами (для включения в область действия модели отношений между субъектами), то есть $S \subset O$. В модели не учитывается временной фактор. Как и в большинства таких моделей поведение системы моделируется с помощью понятия состояния, при этом пространство состояний – это произведение $O \times S \times R$. Текущее состояние системы Q в этом пространстве определяется тройкой, состоящей из множества субъектом, множества объектов и матрицы прав доступа M, то есть $Q=(S,O,M)$. Строки матрицы соответствуют субъектам, а столбцы объектам. Любая ячейка $M[s,o]$ содержит набор прав доступа субъекта s к объекту o. Поведение системы моделируется переходами между состояниями. Переход осуществляется путем внесения изменений в матрицу M с помощью команд следующего вида:

$command\ \alpha(x_1,x_2,...,x_k)$

$if\ r_1\ in\ M\left[x_{s_1},x_{o_1}\right]\ and\ (условия\ выполнения\ команды)$

$r_2\ in\ M\left[x_{s_2},x_{o_2}\right]\ and\ (условия\ выполнения\ команды)$

...

$r_m\ in\ M\left[x_{s_m},x_{o_m}\right]\ and\ (условия\ выполнения\ команды)$

$then...op_1,op_2,...op_n\ (операции,\ составляющие\ команду)$

Здесь α – имя команды, x_i – параметры команды, являющиеся идентификаторами объекта и субъекта, s_i и o_i – индексы субъектов и объектов в диапазоне от 1 до k; op_i – элементарные операции. Элементарные операции выполняются только в том случае, если все условия, означающие присутствие указанных прав доступа в ячейках матрицы M, являются истинными. В классической модели допускаются только следующие операции (примитивные операторы):

Enter r *into* $M[s,o]$ (добавление субъекту s права r для объекта o);

Delete r *from* $M[s,o]$ (удаление у субъекта s права r для объекта o);

Create subject s.....(создание нового субъекта s);

Create subject o.....(создание нового бъекта o);

Destroy subject s.....(удаление нового субъекта s);

Destroy object o.....(удаление нового объекта o).

Модель не налагает каких-либо ограничений на содержание прав доступа и их использование в условиях команд, считая их равнозначными. Поэтому те условия, которые участвуют в выполнении команд, фактически представляют собой не столько права доступа, сколько права управления доступом. То есть модель описывает не только доступ субъектов к объектам, но и распространение прав доступа в системе. На ее основе можно построить формальные процедуры определения возможности перехода системы в опасное для информации состояние, в состояние с нарушением прав доступа, если таковое возникнет в результате модификации матрицы. Харрисон, Руззо и Ульман доказали, что в общем случае не существует алгоритма, который может для произвольной системы, ее начального состояния Q_0 и общего права r решить, является ли данная конфигурация безопасной. Однако для частных случаев такие алгоритмы существуют. Все подобные модели неадекватно оценивают уязвимость системы к атакам, реализуемым с помощью программ типа «троянский конь», атак типа «Подмена доверенного объекта», поскольку никакое правило дискреционной политики не нарушается. Развитием данной модели является дискреционная модель типизированной матрицы доступа (ТМД), дополненная концепцией типов объектов и субъектов, что позволяет несколько смягчить условия, для которых возможно доказательство безопасности системы. Формальное описание модели ТМД включает в себя элементы:

O – множество объектов системы;

S – множество субъектов системы $S \subseteq O$;

R – множество прав доступа субъектов к объектам;

M – матрица доступов;

C – множество команд;

T – множество типов объектов;

$t: O \to T$ – функция, ставящая в соответствие каждому объекту его тип;

$q = (S, O, t, M)$ – состояние системы;

Q – множество состояний системы.

Команды в модели ТМД имеют тот же смысл, что и в модели ХРУ, при этом для всех параметров команд указывается их тип. Перед выполнением команд проверяются типы фактических параметров и, если они не совпадают с указанными в определении команды, то команда не выполняется. При этом используются шесть видов примитивных операторов, отличающихся от аналогичных операторов модели «ХРУ» только использованием типизированных параметров.

Модель Белла-ЛаПадула, известная как «no read up, no write down», то есть «нельзя читать выше, нельзя записывать ниже», была разработана в 1975 г. и основана на правилах секретного документооборота, принятого во многих странах мира. Суть ее в назначении субъектам и объектам специальной метки, получившей название уровня безопасности. Контроль доступа осуществляется в зависимости от уровней безопасности взаимодействующих сторон на основе следующих правил:

1. Уполномоченное лицо (субъект) имеет право читать только те документы, уровень безопасности которых не превышает его собственный уровень безопасности.

2. Уполномоченное лицо (субъект) имеет право заносить информацию только в те документы, уровень безопасности которых не ниже его собственного уровня.

Система в модели представляется так же, как в модели Харрисона-Руззо-Ульмана, в виде множества субъектов, объектов и прав доступа к выполнению функций read и write. В мандатной модели рассматриваются только эти два вида доступа, хотя она может быть расширена введением дополнительных прав, например, на добавление информации, выполнение программ и т.д. Это объясняется тем, что здесь контролируются не операции, а потоки информации, которые могут быть только двух видов – от субъекта к объекту (запись) и наоборот (чтение). Уровни безопасности задаются с помощью функции уровня безопасности $F : S \bigcup O \to L$, которая ставит в соответствие объекту и субъекту уровень безопасности, принадлежащий множеству L, на котором определена решетка, то есть формальная алгебра $(L, \leq, \bullet, \oplus)$, где L – базовое множество уровней безопасности, оператор $\leq$ определяет нестрогое отношение порядка для элементов этого множества, операторы $\bullet$ и $\oplus$ определяют процедуры нахождения наименьшей верхней и наибольшей нижней границы элементов на решетке. Модель системы $\sum(v_0, R, T)$ состоит из начального состояния v_0, множества запросов R и функции перехода $T : (V \times R) \to V$, которая в ходе множества запросов переводит систему из одного состояния в другое. Множество состояний представляется в виде набора упорядоченных пар (F, M), где F -функция уровня безопасности, а M – матрица доступа, отражающая текущую ситуацию с правами доступа субъекта к объектам, но набор прав ограничен лишь правом чтения и записи. ЛаПадул доказал теорему, носящую его имя, о том, что система безопасна

тогда и только тогда, когда ее начальное состояние безопасно и все состояния, достижимые из него путем применения конечной последовательности запросов из R безопасны. Эта теорема получила название основной теоремы безопасности. Суть ее в том, что при любом переходе системы из состояния в состояние не возникает никаких новых отношений и не сохраняется никаких старых отношений, которые будут небезопасны. Мак-Лин развил модель ЛаПадула, разработав целую решетку моделей ЛаПадула, сформировал критерий безопасности функций перехода, показал, что на множестве мандатных моделей может быть построена формальная алгебра и т.д. При всей развитости теоретических положений на практике все-таки применяют лишь фрагменты различных формальных моделей в основном там, где обрабатывается классифицированная информация.

Модель Биба – это интерпретация модели Белла-ЛаПадула, только в ней под безопасностью информации понимается ее целостность. Она предложена в 1977 г. Кеннетом Биба.

Понятие целостности данных в научной литературе определяется несколькими способами, описанию и сравнению которых посвящены отдельные научные статьи. В одной из наиболее распространенных трактовок под целостностью данных подразумевается «*отсутствие ненадлежащих изменений»,* что более корректно раскрыто в модели Кларка-Вилсона (см. далее).

Модель Биба так же, как и модель Белла-ЛаПадула, оперирует субъектами и объектами, которые тоже разбиваются на иерархические уровни, только не секретности, а целостности. Чем выше уровень целостности объекта (например, документа), тем более заслуживает он доверия, тем выше вероятность, что он содержит точные (неискаженные) данные и тем строже правила, по которым субъекты, допускаются к работе с этими данными. Чем выше уровень, к которому отнесен субъект, тем больше ему доверяют, в том числе по возможности модификации информации, содержащейся в объектах. Субъекты выполняют над объектами операции «читать» и «писать». При этом выполняются два правила, при соблюдении которых система гарантированно будет находиться в безопасном (с точки зрения целостности информации) состоянии.

Первое правило – простая аксиома целостности – субъекту данного уровня целостности запрещено выполнять операции «читать» по отношению к объектам более низкого уровня целостности (правило «no read down»). Субъект, читая данные из объекта с более низким уровнем целостности, рискует испортить данные своего уровня. Сделать их менее достоверными. Зато он может читать достоверную информацию более высокого уровня.

Второе правило – аксиома целостности, суть которого состоит в том, что субъекту данного уровня целостности запрещено выполнять операцию «писать» по отношению к объектам более высокого уровня целостности (правило «no write up»), так как он может внести искажения. Вместе с тем поток данных, направленный вниз, не может ухудшить степень целостности объектов.

Модель Кларка-Вилсона была разработана в интересах доказательства защищенности целостности данных как отсутствия ненадлежащих их изменений.

Смысл понятия «ненадлежащее изменение» раскрывается Д. Кларком и Д. Вилсоном: *ни одному пользователю автоматизированной системы, в том числе и авторизованному, не должны быть разрешены такие изменения данных, которые повлекут за собой их разрушение или потерю.* В соответствии с Кларком и Вилсоном при рассмотрении вопроса целостности часто используют интегрированный подход, включающий в себя 9 абстрактных теоретических принципов, а именно:

1) корректность транзакций;
2) аутентификация пользователей;
3) минимизация привилегий;
4) разграничение функциональных обязанностей;
5) аудит произошедших событий;
6) объективный контроль;
7) управление передачей привилегий;
8) обеспечение непрерывной работоспособности;
9) простота использования защитных механизмов.

Корректность транзакций состоит в том, что пользователь не должен модифицировать данные произвольным образом, а только определенными способами так, чтобы сохранить их целостность. Предполагается, что корректность каждой из таких транзакций может быть некоторым способом доказана.

Принцип аутентификации пользователей заключается в том, что к модификации данных допускаются только аутентифицированные пользователи (подлинность прав которых на изменение данных проверена).

Принцип минимизации привилегий накладывает ограничения на возможности процессов и пользователей и подразумевает, что процессы или пользователи должны быть наделены теми и только теми привилегиями, которые естественно и минимально необходимы. Излишние полномочия создают излишние возможности для модификации данных и нарушения их целостности.

Принцип разграничения функциональных обязанностей подразумевает организацию работ с данными только таким образом, что в каждой из ключевых стадий, составляющих единый критически важный с точки зрения целостности данных процесс, необходимо участие различных пользователей. Этим гарантируется, что один пользователь не может выполнить весь процесс целиком (или даже две его стадии). Данный принцип распространяется и на программы, и на пользователей. Однако реально он применяется в основном для программного обеспечения.

Принцип аудита произошедших событий является превентивной мерой против потенциальных нарушителей, так как направлен на восстановление полной картины происшедшего. Его реализация позволяет выявить нарушителя.

Принцип объективного контроля является одним из краеугольных камней политики контроля целостности. Смысл его в том, что он подчеркивает состоятельность контроля целостности лишь в случае, когда данные отражают реальное положение дел.

Принцип управления передачей привилегий обеспечивает эффективность политики безопасности, так как, если схема назначения привилегий неадекватно отражает организационную структуру предприятия или не позволяет администраторам гибко манипулировать ею, защита становится тяжким бременем и провоцирует попытки обойти ее там, где она мешает нормальной работе.

С некоторыми оговорками в научной литературе в основу контроля целостности закладывают принцип обеспечения непрерывной работы, включая защиту от сбоев, стихийных бедствий и других форс-мажорных обстоятельств. Принцип простоты использования защитных механизмов подразумевает, что самый безопасный путь эксплуатации системы будет также наиболее простым, и наоборот, самый простой – наиболее защищенным.

Модель Кларка-Вилсона появилась в результате анализа реально применяемых методов обеспечения целостности документооборота в коммерческих компаниях. Основные понятия рассматриваемой модели – это корректность транзакций и разграничение функциональных обязанностей. Модель задает правила функционирования компьютерной системы и определяет две категории объектов данных и два класса операций над ними. Все содержащиеся в системе данные подразделяются на контролируемые и неконтролируемые элементы. Целостность первых обеспечивается моделью Кларка-Вилсона. Последние содержат информацию, целостность которой в рамках данной модели не контролируется. Данная модель вводит два класса операций над элементами данных: процедуры контроля целостности (integrity verification procedures – IVP) и процедуры преобразования (transformation procedures – TP). Наконец, модель содержит 9 правил, определяющих взаимоотношения элементов данных и процедур в процессе функционирования системы, показанных в таблице 4.1.

4.3. Логико-лингвистические модели процессов реализации угроз безопасности информации

Огромное разнообразие угроз безопасности информации в ИС и способов их реализации обусловили необходимость создания электронных баз данных для сбора и накопления сведений об угрозах, их актуализации и применения для решения задач защиты информации в ИС.

Сегодня эти сведения представляют собой вербальные описания, что практически исключает их использование в автоматизированном режиме в формальных процедурах и алгоритмах анализа угроз.

Таблица 4.1 – Правила и принципы политики контроля целостности

Правило модели Кларка-Вилсона	Обозначение правила	Принципы политики контроля целостности, реализуемые правилом
Множество всех процедур контроля целостности (IVP) должно содержать эти процедуры для любого элемента данных из множества всех контролируемых данных	C1	Корректность транзакций, объективный контроль
Все процедуры преобразования (TP) должны быть реализованы корректно, то есть без нарушения целостности обрабатываемых контролируемых данных. С каждой процедурой преобразования должен быть связан список контролируемых данных, которые допустимо обрабатывать процедурой	C2	Корректность транзакций
Должна контролироваться допустимость применения TP к контролируемым данным в соответствии со списками, указанными правилами C2	E1	Минимизация привилегий, разграничение функциональных обязанностей
Должен поддерживаться список разрешенных пользователю процедур преобразования с указанием допустимого для каждого TP и данного пользователя набора обрабатываемых элементов данных	E2	Корректность транзакций, аутентификация пользователя, минимизация привилегий,
Список, определенный правилом C2, должен отвечать требованию разграничения функциональных обязанностей	C3	Разграничение функциональных обязанностей
Должна проводиться аутентификация всех пользователей, пытающихся выполнить какую-либо процедуру преобразования	E3	Аутентификация пользователя
Каждая TP должна записывать в журнал регистрации (специальный элемент из состава контролируемых данных, предназначенный только для добавления в него информации) информацию, достаточную для восстановления полной картины каждого применения этой TP	C4	Аудит произошедших событий
Любая TP, обрабатывающая неконтролируемый элемент, должна выполнять только корректные преобразования этого элемента, в результате которых неконтролируемый элемент превращается в контролируемый	E4	Корректность транзакций
Только специально уполномоченное лицо может изменять списки, определенные правилами C2 и E2. Это лицо не имеет право выполнять какие-либо действия, если уполномочено изменять регламентирующие эти действия списки	C5	Разграничение функциональных обязанностей

Вместе с тем уже сегодня крайне востребованы для автоматизации процессов прогнозирования и анализа угроз соответствующие экспертные системы [67].

Переход от вербальных к формальным описаниям угроз безопасности информации связан с необходимостью разработки соответствующих правил формального описания. При этом подлежат учету разнообразные факторы, влияющие на реализацию угроз, что весьма сложно сделать при большом их разнообразии экспертным путем даже специалистам высокого уровня.

Такая формализация позволила бы на единой методологической основе разработать правила формального описания угроз, оценивать возможности их

реализации, сформировать базу знаний для построения систем поддержки принятия решений по отнесению угроз к актуальным для конкретных ИС с применением элементов искусственного интеллекта и др.

Однако для формальных описаний необходимы соответствующие искусственные языки, которых сегодня разработано достаточно много.

В соответствии с [68] в классификационную схему искусственных языков, как правило, включают три группы языков:

1) реляционные языки, для которых характерно введение конечного множества бинарных отношений, с помощью которых передаются бинарные связи между элементами языка;

2) языки ролевых фреймов, в которых каждая единица в предложении описывается через семантические роли, которые она может выполнять;

3) логические языки, которые используются в формальных системах логического типа, таких как исчисление высказываний или исчисление предикатов первого порядка, многозначные логики и модальные исчисления.

При использовании этих языков могут применяться формализмы математической логики, теории нечетких множеств, теории графов и др. Строгость логических отношений между элементами описания может варьироваться в широких пределах: от отношений, выраженных нечеткими оценками, до отношений строго детерминизма. Наиболее часто упоминаются в литературе [68] и используются на практике так называемые *реляционные языки* (от англ. relation – отношение) и языки ролевых фреймов[14].

К реляционным языкам, как правило, относят табличные языки Кодда, контекстно-свободный плекс-язык, тензорный язык Крона RX-коды, синтагматические цепи и семантические сети [69]. Краткая характеристика этих языков приведена ниже, кроме тензорного языка, который оказывается весьма сложным в практическом применении при описании угроз.

В табличном языке Кодда данные о предметной области в виде строки таблицы формально задаются на множестве:

$$H = I \times H_1 \times \ldots \times H_k . \qquad (4.1)$$

где H_k – множества значений атрибутов $k = \overline{1,K}$, а I – множество имен сущностей, данных в виде n-арных отношений ($n \le k$).

Язык Кодда с табличной формой представления информации ранее составлял теоретическую основу разработки всевозможных баз данных. Сегодня этот язык представляет скорее исторический интерес, но на практике он используется в качестве базиса для построения различных баз данных и развития языков более высокого уровня. Язык позволяет описывать данные, но не знания, поэтому его применение сегодня для описания угроз нецелесообразно [53, 54].

[14] Сегодня известно достаточно много и других языков, однако их возможности для описания угроз безопасности информации весьма ограничены и далее не рассматриваются.

Основными компонентами контекстно-свободного плекс-языка являются: плекс-элементы (абстрактный объект, имеющий определенное количество контактов, то есть входов и выходов для соединения с другими плекс-элементами), операция конкатенации и грамматика, определяющая правила соединения плекс-элементов между собой. Плекс-элементы образуют алфавит языка. Операция конкатенации формально задается матрицей $\|\gamma_{ij}\|$, элементы которой отражают связи между контактами i и j. Грамматикой языка называется четверка: $G = \langle v_0, A, V, R \rangle$, где v_0 – начальный символ, $A = \{a_i\}$ – алфавит терминальных плекс-элементов; $V = \{v_k\}$ – алфавит вспомогательных плекс-элементов; R – множество правил вывода [68].

Контекстно-свободный плекс-язык, так же, как и табличные языки Кодда, весьма сложно использовать при логико-лингвистическом описании угроз безопасности из-за недостаточных выразительных возможностей.

Рассматривая широко известные *RX-коды*, необходимо отметить, что в этом языке в качестве правильных синтаксических выражений используются двойки из множества понятий $\{X_j^m\}$ и множества отношений $\{R_j\}$ и конъюнкции двоек. Для интерпретации выражений языка RX-кодов всем понятиям и отношениям предписываются некоторые семантические значения, например, для понятий: X_1^i – «угроза перехвата трафика с использованием «сниффера», относящаяся с номером 1 к i-му классу угроз, объединяющему все угрозы перехвата трафика; $X_1^{(j)}$ – «угроза переполнения буфера операционной системы», относящаяся с номером 1 к j-му классу угроз «Отказ в обслуживании». Семантические значения для отношений могут быть, например, следующие: R_m – «иметь имя» с порядковым номером m; R_k – «повторять» с порядковым номером k; R_n – «быть меньше» с порядковым номером n и т.д.

Приведем пример использования языка RX-кодов для описания угрозы атаки «sin-flood».

Пример 4.1. Суть атаки «syn-flood» заключается в посылке большого количества пакетов сообщений, сформированных по протоколу TCP с запросом на соединение с атакуемым компьютером (TCP-запросов). Атакуемый компьютер по технологии установления соединений направляет на атакующий компьютер хакера квитанцию о получении запроса и согласии на установление виртуального канала, при этом записывает данные запросившего компьютера в буфер.

Атакующий, получив эту квитанцию, должен был бы направить свое уведомление о том, что он начинает передачу, которая на атакуемой стороне является сигналом для очищения буфера и начала приема пакетов. Однако хакер не посылает это уведомление, а направляет следующий запрос, что заканчивается очередной записью данных об атакующем компьютере в буфер атакуемого. Это многократно повторяется, что приводит к переполнению

буфера атакуемого компьютера и его «зависанию». Пусть рассматриваемая угроза обозначается как X_0^1 и в описании угрозы используются следующие понятия: X_1^1 – имеет место класс угроз «Отказ в обслуживании»; X_2^1 – имя угрозы «Переполнение буфера операционной системы»; X_3^1 – источником угрозы является внешний нарушитель; X_4^1 – пакет с запросом на соединение отправлен; X_5^1 – состояние буфера операционной системы атакуемого компьютера нормальное; X_6^1 – количество отправленных нарушителем пакетов с запросом на соединение превышает предельное для буфера атакуемого компьютера значение X_7^1 – состояние компьютера «отказ в обслуживании». Суть отношений между понятиями состоит в следующем: R_1 – имеет место; R_2 – выполняется повтор; R_3 – выполняется условие; R_4 – следствием является. Для расширения возможностей описания могут использоваться пропозициональные связки, такие как операция конъюнкции $\&$ (соответствующая логике «И»), операция дезъюнкции в разделительном смысле $\vee$ (соответствующая логике «эксклюзивное ИЛИ», когда утверждается одно и только одно из высказываний), и др.

С учетом этих обозначений простейшая формальная запись угрозы атаки «syn-flood» с использованием RX-кодов показана на рисунке 4.3.

$$X_0^0 = \underbrace{(R_1X_1^1)(R_1X_2^1)(R_1X_3^1)(R_1X_5^1)}_{\text{Имеют место } (R_1)\text{ класс угроз - «Отказ в обслуживании» } (X_1^1)\text{; имя угрозы -переполнение буфера } (X_2^1)\text{; Источник угрозы - внешний нарушитель } (X_3^1)\text{; Состояние буфера операционной системы до атаки нормальное } (X_5^1)} \& \underbrace{\left\{R_2X_4^1R_3\left[R_1\left(X_5^1\right)\right]\right.}_{\text{Повторяется } (R_2)\text{ отправка хакером пактов с запросом на соединение } (X_4^1)\text{, пока выполняется условие } (R_3)\text{, при котором имеет место } (R_1)\text{ нормальное состояние буфера операционной системы } (X_5^1)} \hat{\vee} \underbrace{\left.\left(R_3X_6^1\right)R_4X_7^1\right\}}_{\text{При наличии условия } (R_3)\text{, когда количество отправленных хакером пакетов превышает предельное для буфера атакуемого компьютера значение } (X_6^1)\text{ следствием } (R_4)\text{ становится отказ в обслуживании } (X_7^1)}$$

Рисунок 4.3 – Пример формальной записи угрозы сетевой атаки «syn-flood» с использованием RX-кодов

Язык *синтагматических цепей* (СЦ) предоставляет значительно больше возможностей для описания угроз, чем язык RX-кодов. Это обусловлено не только тем, что построение синтагм интуитивно понятно, но и тем, что для расширения его возможностей используются квантификаторы, модификаторы и императивы[15].

[15] Императивы – словесные выражения указаний, реализуемых в ИС в интересах, например, парирования угроз или решения других задач защиты. Примерами императивов являются такие слова и словосочетания, как «провести идентификацию», «изменить пароль доступа» и др. Квантификаторы служат для введения количественных и качественных мер для понятий, отношений и императивов. Примерами квантификаторов могут быть слова «всегда», «для всех», «очень редко» и т.п. Модификаторы приписывают понятиям, отношениям и императивам некоторые характеристики, которые в них ранее не фиксировались. Примерами модификаторов могут быть слова «опасная», «аварийное», «зависание» и т.п.

Еще более важным является расширение состава элементов описания процесса реализации угроз такими классами функциональности как модальность[16], что можно использовать в алгоритмах функционирования перспективных систем поддержки принятия решений по ТЗИ.

Рассмотрим описание той же атаки «syn-flood» с использованием языка синтагматических цепей.

Пример 4.2. Введем следующие понятия и отношения:

A_0 – атака «syn-flood»; b_1 – компьютер хакера; b_2 – атакуемый компьютер; b_3 – запрос на соединение; b_4 - буфер атакуемого компьютера; b_5 – квитанция; b_6 – уведомление; n_1 – количество передаваемых на атакуемый компьютер пакетов с запросами на соединение; n_2 – предельно допустимое количество записей в буфере атакуемого компьютера о запросившем соединение компьютере; s_1 – состояние буфера нормальное; s_2 – переполнение буфера (зависание); r_1 – имеет имя; r_2 – передает, r_3 – блокирует; r_4 – получает; r_5 – записывает; r_6 – находится в состоянии; r_7 – повторяет; r_8 – больше; r_9 – стало причиной; i_0 – класс атаки; i_1 – TCP-пакет с запросом на соединение; i_2 – TCP-пакет с квитанцией о согласии на соединение; i_3 – сведения о компьютере, запросившем соединение, записываемые в буфер; k_0 – «Отказ в обслуживании».

Тогда формальная запись данной атаки имеет следующий вид:

$$A_0 = (i_0 r_1 k_0)\left\{b_2 r_7\left[(b_2 r_6 s_1)\left((b_1 r_2 i_1)(b_2 r_4 i_1)\right)(b_2 r_5 i_3)(b_2 r_2 i_2) b_1 r_3 i_2\right] r_9 (n_1 r_8 n_3)\right\} r_9 s_2 . \qquad (4.2)$$

Такое формальное описание угрозы позволяет выделить признаки, по которым можно осуществлять поиск в автоматическом режиме записи о данной угрозе и ее элементах описания в базе данных, при сборе в сети Internet информации в интересах статистического анализа реализации угроз и др.

Семантические сети (СС) представляют собой следующий шаг в развитии машинно-читаемых представлений информации. Они дают возможность получать логические выводы на основании введенных правил.

СС, как и СЦ, формально определяется тремя классами термов [52, 68]:

1) понятиями $X = \{X_1, X_2, ..., X_s\}$;

2) именами $I = \{i_1, i_2, ..., i_m\}$;

3) предикатами (типами отношений) $R = \{\rho, R\ , R_2, ..., R_q\}$, при этом предикат ρ является особым, означающим «иметь имя», между понятиями в виде триплета $(x_k - r_j - x_s)$ и между именами в виде триплета $(i_p - r_g - i_h)$.

Запись на языке СС значительно богаче, так как с его помощью можно выразить сложные совокупности фактов, описывать не только постоянные

[16] Модальность (от лат. Modus – мера, способ) – оценка связи, устанавливаемой в высказывании, данная с той или иной точки зрения. Модальная оценка выражается с помощью модальных понятий: «необходимо», «возможно», «случайно», «доказуемо», «опровержимо», «обязательно», «разрешено», «хорошо» и т.п.

отношения, присущие объекту моделирования, но и ситуативные отношения, а также обобщать ситуации за счет выделения типовых структур отношений [69]. Это позволяет использовать их при создании иерархических баз знаний и разработке формальных процедур ситуационного поиска решений. Применительно к решению задач ТЗИ этот язык пока непосредственно не применялся. В отличие от традиционного языка СС, применяемого чаще всего для поиска семантической информации, от созданного затем языка ситуационного управления (ЯСУ) [52, 69] – для управления процессами и объектами, распознавания ситуаций и др., в данном случае необходимо не только автоматизировать поиск в базе знаний, распознавание угроз по выборочным сведениям о них, но и подготавливать в автоматизированном режиме данные для анализа угроз, оценок возможностей их реализации в конкретных ИС, определения состава актуальных угроз и т.д. Специфика задач ТЗИ обусловливает то, что, как и в интересах ситуационного управления был преобразован язык СС в ЯСУ [52], в интересах решения задач ТЗИ язык СС с учетом ЯСУ тоже должен быть соответствующим образом изменен в части понятий, имен и отношений. Существенными изменениями в языке СС применительно к решению задач, связанных с анализом угроз, могут быть, например, по аналогии с [69], отношения, устанавливаемые между действиями, соответствующими процессам реализации угроз, то есть отношения между предикатами, определяющими такие действия. Таким образом, в этом случает все предикаты, определяющие отношения, разбиваются на два подмножества: во-первых, множество отношения между понятиями и именами и, во-вторых, множество отношений между предикатами, самими определяющими другие отношения. Примеры некоторых понятий, имен и отношений между именами применительно к задачам описания и анализа угроз безопасности информации в ИС приведены в таблице 4.2 и 4.3.

Это позволяет в формальном описании угрозы связывать между собой последовательно выполняемые и обусловленные действия.

Для описания угроз в этом случае используются четыре класса термов: 1) понятия $\{X_s\}$; 2) имена $\{i_m\}$; 3) предикаты, определяющие отношения между понятиями в виде триплетов $(x_k - r_j - x_s)$ и между именами – $(i_p - r_g - i_h)$; 4) предикаты, определяющие отношения между выполняемыми действиями в виде триплетов $(r_u - r_d - r_b)$, к которым относятся, например, отношения следования, временные, каузативные и др.

Семантическую сеть представляют часто в виде ориентированного графа, вершинами которого являются понятия и имена, а дугами – отношения между ними. Пример графа, описывающего семантику понятия угрозы копирования файла с конфиденциальной информацией с использованием файла cookie без раскрытия сценария реализации, приведен на рисунке 4.4.

Таблица 4.2 – Примеры некоторых понятий и имен, используемых для описания угроз с применением семантических сетей

Понятие	Обозначение понятия	Имя понятия	Обозначение имени понятия
Наименование угрозы	X_0	Угроза копирования файла пользовательских данных с использованием файла cookie	i_1
		Угроза уничтожения файла	i_2
		Угроза подмены (модификации) файла	i_3
		Угроза блокирования исполняемого файла	i_4
Класс угрозы	X_1	Угроза проникновения в операционную среду	i_5
		Угроза отказа в обслуживании	i_6
		Угроза перехвата трафика	i_7
Источник угрозы	X_2	Внешний	i_8
		Внутренний	i_9
Объект воздействия	X_3	Каталог	i_{10}
		Файл системный	i_{11}
		Файл cookie	i_{12}
		Текстовый файл	i_{13}
		Графический файл	i_{14}
		Исполняемый файл прикладной программы	i_{15}
Хост	X_4	Хост атакующего	i_{16}
		Хост атакуемого	i_{17}
		Хост-«жертва»	i_{18}
		Хост – «посредник»	i_{19}
		Поисковая система	i_{20}
Способ реализации	X_5	Идентификатор №1	i_{21}
		Идентификатор №2	i_{22}
Состояние	X_6	Нормальное	i_{23}
		Отказ в обслуживании	i_{24}
		Угроза реализована	i_{25}
		Угроза блокирована	i_{26}
Вспомогательное	X_7	Сетевой адрес	i_{27}
		Предельное количество записей в буфер	i_{28}
		Вредоносная программа	i_{29}
		Количество отправленных пакетов	i_{30}
Уровень опасности угрозы	X_8	Недопустимая	i_{31}
		Высокая опасность	i_{32}
		Средняя опасность	i_{33}
		Низкая опасность	i_{34}

Таблица 4.3 – Примеры некоторых отношений между понятиями и именами при формальной записи процессов реализации угроз

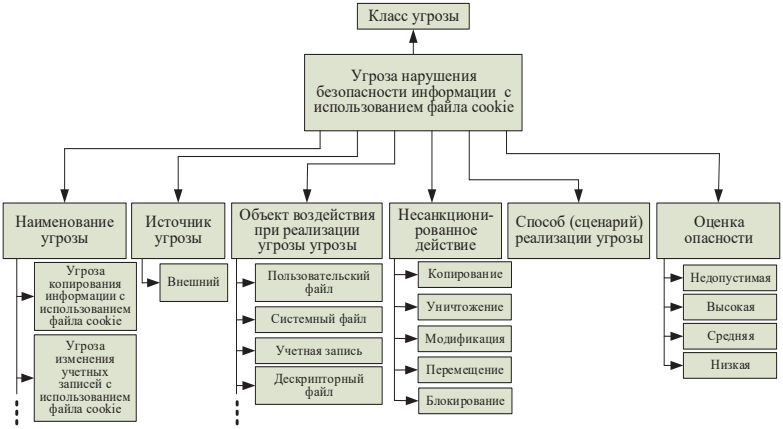
Тип отношения	Содержание отношения	Обозначение
Временные	Быть одновременно	r_1
	Быть позже	r_2
	Быть раньше	r_3
Структурные	Быть в одном сегменте	r_4
	Находиться за пределами сети	r_5

Тип отношения	Содержание отношения	Обозначение
	Содержаться в файле	r_6
	Быть за межсетевым экраном	r_7
Сравнительные	Быть равным	r_8
	Быть больше	r_9
	Быть меньше	r_{10}
Классификационные	Содержать в качестве элемента	r_{11}
	Относиться к чему-либо (например, к категории – виду, классу, типу)	r_{12}
Идентифицирующие	Иметь имя	r_{13}
	Быть чем-либо (например источником, уязвимостью и т.д.)	r_{14}
	Быть актуальной	r_{15}
Каузативные	Быть целью	r_{16}
	Быть причиной	r_{17}
	Быть следствием	r_{18}
	Быть условием	r_{19}
Динамические	Принять что-то (пакет, сообщение, квитанцию и т.д.	r_{20}
	Передать на	r_{21}
	Повторить	r_{22}
	Блокировать	r_{23}
	Копировать	r_{24}
	Перехватить	r_{25}
	Ограничить	r_{26}
	Модифицировать	r_{27}
Прагматические	Служить для	r_{28}
	Находиться в состоянии	r_{29}
	Обеспечить	r_{30}
	Быть дополнением	r_{31}
Следования	Происходить непосредственно перед началом следующего действия	r_{32}
	Происходить непосредственно после завершения предыдущего действия	r_{33}
	Происходить параллельно с другим действием (другими действиями)	r_{34}

Рассмотрим отдельно описание способа (сценария) реализации угрозы копирования нужной нарушителю информации с использованием файла cookie.

Пример 4.3. Введем с учетом таблиц 4.3 и 4.4 следующие обозначения: i_{20} – имя способа реализации угрозы копирования информации с использованием файла cookie; i_{16} – компьютер атакующего; i_{17} – атакуемый компьютер; i_{19} – компьютер со сниффером для перехвата файлов cookies; $i_{21}^{(res)}$ – передача запроса на информационный ресурс в поисковую систему; $i_{21}^{(cook)}$ – передача файла cookie с поисковой машины на атакуемый хост; i_{25} – перехват файла cookie на компьютере-посреднике с установленной программой «сниффер» и передача его на компьютер атакующего; i_{27} – внедрение в файл cookie вредоносного скрипта;

$r_{21}^{(scr)}$ – передача файла cookie, инфицированного вредоносным скриптом, на атакуемый компьютер; $r_{20}^{(scr)}$ – прием и просмотр веб-браузером атакуемого компьютера файла cookie и запуск на исполнение внедренного в него скрипта; r_{24} – копирование файла; $r_{21}^{(f)}$ – передача скопированного файла на хост атакующего.



Формальная запись угрозы с использованием языка семантической сети без раскрытия способа ее реализации

$$(X_0\rho i_1)\,\&\,\left[(X_0 r_{12} X_1)(X_1 r_{13} i_5)\right]\&\left[(X_0 r_{11} X_2)(X_2 r_{13} i_8)\right]\&\left[(X_0 r_{11} X_3)(X_3 r_{13} i_{13})\right]\&$$
$$\&\left[(X_0 r_{11} X_4)(X_4 r_{13} i_{16})\right]\&\left[(X_0 r_{11} X_4)(X_4 r_{13} i_{17})\right]\&\left[(X_0 r_{11} X_4)(X_4 r_{13} i_{19})\right]\&$$
$$\&\left[(X_0 r_{11} X_5)(X_5 r_{13} i_{21})\right]\&\left[(X_0 r_{11} X_6)(X_6 r_{13} i_{25})\right]\&\left[(X_0 r_{11} X_8)(X_8 r_{13} i_{32})\right]$$

Рисунок 4.4 – Семантическая сеть, определяющая описание угрозы копирования файла пользовательских данных, без раскрытия способа ее реализации

Тогда формальная запись способа реализации угрозы на языке СС имеет вид:

$$X_5\rho i_{21}\,\&\,i_{17}r_{21}^{(res)}i_{20}\,\&\,r_{21}^{(cook)}i_{17}\,\&\,i_{19}r_{25}i_{16}\,\&\,i_{16}r_{27}i_{12}\,\&\,i_{16}r_{21}^{(scr)}i_{17}\,\&\downarrow$$
$$i_{17}r_{20}^{(scr)}i_{12}\,\&\,i_{17}r_4 i_{13}\,\&\,i_{17}r_{21}^{(f)}i_{16}. \qquad (4.3)$$

Подобные записи можно использовать в перспективных экспертных системах, предназначенных для автоматического или автоматизированного анализа угроз безопасности информации, в специализированных системах поддержки принятия решений по ТЗИ в ИС, при решении задач сбора и обработки данных об угрозах безопасности информации по результатам мониторинга сети Internet в интересах ведения банка данных угроз и др.

Расширение рассмотренных языков для логико-лингвистического моделирования угроз позволит формально описывать способы применения мер и средств защиты от актуальных угроз безопасности информации в ИС, а также в математических моделях оценки возможностей реализации угроз при автоматизированном формировании исходных данных о них при проведении соответствующих расчетов.

В *фреймовых или ролевых языках* понятия определяются совокупностью семантических (смысловых) ролей, которые выражают их сущность, и

формально задаются строкой, называемой фреймом (от англ. frame – рамка, каркас) [53]:

$$\langle i, \rho_1, \rho_2, ..., \rho_m, \rho_{m+1}, ..., \rho_M \rangle, \tag{4.4}$$

где i – имя понятия;

$\rho_r, r = \overline{1,m}$ – обязательные роли;

$\rho_r, r = \overline{m+1,M}$ – необязательные роли (множество таких ролей может быть пустым).

Обязательные и необязательные роли, в свою очередь, являются сложными понятиями, представляемыми фреймами следующего уровня, называемые слотами, что позволяет строить иерархические фреймовые описания рекурсивного вида:

$$\left\langle \begin{array}{l} i; \rho_1\left(\left\langle i, \rho_1^1 \left\langle i, \rho_1^2 \left\langle ... \rho_1^{Z_1} \right\rangle \right\rangle \right\rangle, \rho_2 \left\langle i, \rho_2^1 \left\langle i, \rho_2^2 \left\langle ... \rho_2^{Z_2} \right\rangle \right\rangle \right\rangle \right), ..., \\ \rho_m\left(\left\langle i, \rho_m^1 \left\langle i, \rho_m^2 \left\langle ... \rho_m^{Z_m} \right\rangle \right\rangle \right\rangle \right) \end{array} \right\rangle \tag{4.5}$$

Существенным недостатком таких описания является дублирование однотипной информации, поэтому в моделях, использующих фреймовые описания, применяется принцип наследования свойств. Суть его состоит в упорядоченной рекурсии с помощью специального графа, в котором вершины соответствуют фреймам, а путь от корней к вершине указывает на порядок рекурсии. При этом вся информация, записанная во фреймах, лежащих на пути от корневой вершины до заданной, автоматически переносится в заданную вершину. Для представления знаний разрабатываются типовые фреймы, такие как фрейм-состав, фрейм-соединение, фрейм-назначение, фрейм-функция, фрейм-параметр, фрейм-сценарий и др. [53, 68].

Это существенно расширяет возможности логико-лингвистического моделирования, что позволяет использовать такие модели для описания функциональных структур различного назначения, формирования эффективных поисковых процедур в базах знаний и поиска рациональных решений по ТЗИ, реализации методов ситуационного управления защитой информации и др. Вместе с тем пока такие модели применительно к задачам ТЗИ не разрабатывались.

Из *предикатных языков* наибольшее применение в практике моделирования нашел язык исчисления предикатов первого порядка. Исходными базовыми элементами этого языка являются:

1) константы – фиксированные элементы из некоторого множества;

2) переменные, обозначаемые обычно малыми латинскими буквами с индексами или без них;

3) функциональные символы, обозначаемые обычно буквами f, g, h и греческими малыми буквами с индексами или без них, служащие для обозначения функциональной зависимости между объектами (иногда для

этого используются слова русского языка, не содержащие заглавных букв, например «опередить», «стереть», «модифицировать» и т.п.);

4) предикатные символы, обозначаемые или заглавными латинскими буквами, или словами русского языка, состоящими из заглавных букв (например, «БОЛЬШЕ», «ОЖИДАЕТ» и т.п.).

Предикат – это по сути формальная символьная запись, состоящая из указанных базовых элементов. Если ли символу предиката и символу предикатной переменной поставить в соответствие определенные слова или словосочетания, то получится высказывание. В этом смысле исчисление предикатов первого порядка является расширением исчисления высказываний в математической логике, называемого пропозициональной логикой [55].

С помощь предикатов первого порядка строятся формальные предложения в виде правильно построенных формул, в результате расчетов по которым получаются результирующие значения «истина» или «ложь».

Такой подход может быть использован, например, при формализации описания процедур выявления уязвимостей в программном коде, при выявлении логических связей между уязвимостью и составом угроз, которые могут реализовываться в данной ИС путем эксплуатации этой уязвимости с учетом структурных и функциональных характеристик ИС и др. Вместе с тем, пока предикатные языки в предметной области ТЗИ находят весьма ограниченное применение.

Следующими уровнями формализации задач, решаемых в предметной области ТЗИ, являются логические модели, в которых естественно-языковые высказывания замещаются на примитивные высказывания – литералы, между которыми устанавливаются отношения, приписываемые формальной логикой (см. раздел 5.3), и алгоритмические модели, реализуемые, преимущественно, на языках программирования (далее не рассматриваются).

4.4. Логические сети и их применение при моделировании процессов реализации угроз безопасности информации

Логические сети являются расширением аппарата теории конечных автоматов, широко применяемого в самых различных сферах, например, в теории кибернетических систем, в телемеханике, информатике и вычислительной технике [63–66, 70] и др. Сама теория конечных автоматов – это частный случай общей математической теории систем [71], позволяющая изучать поведение реальных физических устройств и явлений на основе разработки их абстрактных математических моделей.

Конечный автомат в соответствии с [63–66, 69, 70] определяется следующим образом. Пусть имеется некоторое устройство с n входами и q выходами (рисунок 4.5), на каждый вход которого может поступать сигнал из конечного множества сигналов $X = \{x_1, x_2, ..., x_k\}, k = \overline{1, K}$, называемого входным алфавитом, а совокупность поступивших на n входов сигналов, обозначаемых соответствующими символами, – входным словом.

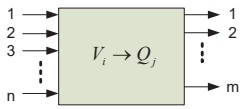


Рисунок 4.5 – Обобщенная схема конечного автомата без памяти

При этом некоторые из элементов такого слова (символов) могут быть пустыми. На выходе устройства появляются выходные слова, составленные из символов выходного алфавита $Y = \{y_1, y_2, ..., y_m\}, m = \overline{1, M}$.

Элементарный такт работы устройства состоит в том, что при появлении на входе устройства входного слова V_i устройство выдает на выходах комбинацию символов, составляющих выходное слово Q_j.

Пусть работа устройства определяется только входным словом, тогда она будет полностью определена, если задать соответствия всех входных слов выходным в виде:

$$\begin{cases} V_1 \to Q_{j_1}; \\ V_2 \to Q_{j_2}; \\ \dots \\ V_{k^n} \to Q_{j_{k^n}}, \end{cases} \tag{4.6}$$

где k^n – количество строк по числу различных входных слов длины n над алфавитом X, состоящим из k различных символов.

Устройство, условия работы которого описываются с помощью указанных соответствий (4.6), называют детерминированным конечным автоматом без памяти [69, 70].

Более сложными являются автоматы, для которых описываются не только входные и выходные слова, но и состояния самого автомата. Пусть задан конечный алфавит $S = \{s_0, s_1, ..., s_c\}, c = \overline{1, C}$, задающий состояния устройства, и работа устройства полностью определяется входным словом и внутренним состоянием в определенный такт работы (рисунок 4.6).

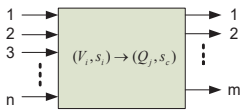


Рисунок 4.6 – Обобщенная схема конечного автомата с памятью

Пара V_i и s_c однозначно определяет выходное слово и внутреннее состояние, в которое перейдет устройство в следующий такт. Работа такого устройства полностью определена, если заданы две таблицы, первая из

которых (таблица 4.5) определяет выходное слово, соответствующее паре V_s, s_i, а вторая (таблица 4.6) – внутреннее состояние устройства в следующий такт [70].

Таблица 4.5 – Соответствие выходного слова входному слову и состоянию устройства

Входные слова	Выходные слов для состояний			
	s_1	s_2	$\cdots$	s_c
V_1	Q_{1_1}	Q_{1_2}	$\cdots$	Q_{1_c}
V_2	Q_{2_1}	Q_{2_2}	$\cdots$	Q_{2_c}
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
V_k	$\cdots$	$\cdots$	$\cdots$	Q_{k_c}

Таблица 4.6 – Соответствие внутреннего состояния устройства входному слову

Входные слова	Выходные слов для состояний			
	s_1	s_2	$\cdots$	s_c
V_1	s_{1_1}	s_{1_2}	$\cdots$	s_{i_c}
V_2	s_{2_1}	s_{i_2}	$\cdots$	s_{2_c}
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
V_k	$\cdots$	$\cdots$	$\cdots$	s_{k_c}

Такие автоматы называют детерминированными конечными автоматами с памятью. При этом параметр c называют глубиной памяти.

В общем случае детерминированные конечные автоматы описываются пятеркой [72]

$$A = (V, S, \delta, s_0, Q), \tag{4.7}$$

где s_0 – начальное состояние автомата как дополнительная его характеристика;

δ – функция, отображающая множество $V \times S$ в множество Q и называемая функцией переходов или переходной функцией.

Естественным обобщением детерминированных автоматов являются автоматы стохастического типа или стохастические конечные автоматы. В таких автоматах вместо однозначного соответствия $V_i \to Q_j$ или $(V_i, s_i) \to (Q_j, s_c)$ задается вероятность соответствия в виде стохастической матрицы. Например, для конечного автомата без памяти такая матрица вероятностей p_{ij} имеет вид, приведенной в формуле (4.8).

$$\begin{array}{c|c|c|c|c} & Q_1 & Q_2 & \dots & Q_{i^m} \\ \hline V_1 & p_{11} & p_{12} & \dots & p_{1i^m} \\ V_2 & p_{21} & p_{22} & \dots & p_{2i^m} \\ \dots & \dots & \dots & \dots & \dots \\ V_{k^n} & p_{k^n 1} & p_{k^n 2} & \dots & p_{k^n i^m} \end{array}, \sum_{j=1}^{i^m} p_{ij} = 1. \tag{4.8}$$

В случае автомата с памятью такие матрицы задаются для каждого столбца обеих таблиц 4.5 и 4.6.

В подавляющем большинстве случаев моделирования систем с помощью теории конечных автоматов символы алфавитов X, Y и S кодируются двоичным кодом, то есть символы принимают значения 0 или 1. Путем такого кодирования автомат с произвольным входным и выходным алфавитом может быть приведен к автомату с двоичным входом и выходом. Это позволяет использовать при анализе таких систем булевы функции [69] и описать работу автомата системой таких функций:

$$\begin{cases} y_1 = f_1(x_1, x_2, ..., x_n); \\ y_2 = f_2(x_1, x_2, ..., x_n); \\ ..... \\ y_m = f_m(x_1, x_2, ..., x_n). \end{cases} \tag{4.9}$$

Вся работа автомата в этом случае или задается в виде конечных таблиц, или в виде аналитической записи логических (булевых) функций.

Вместе с тем теория конечных автоматов в части использования при моделировании в предметной области защиты информации находит пока весьма узкое применение, – чаше всего при моделировании работы некоторых цифровых устройств (например, шифраторов и дешифраторов, буферов памяти, накопителей и т.п.), в том числе в интересах оценки устойчивости их функционирования в условиях деструктивных информационных воздействий. Более широкое применение сегодня находят логические сети.

Логические сети или модели логического вывода применяются в том случае, когда необходимо установить логическую связь между вводимыми в модель параметрами и условиями, определяющими моделируемый процесс, и логическим решением о том, что по завершении процесса произойдет ожидаемое событие (решение «да») или не произойдет (решение «нет»). Логическая сеть определяется следующим образом.

Пусть имеется конечное множество $A = \{1, 2, 3, ..., m\}, m = \overline{1, M}$ и пусть задано множество $B = \{(i, j)\}$, элементами которого являются упорядоченные пары элементов множества A, где i, j – любые из элементов A, $i \neq j$, и множество $F\{f_1, f_2, ... f_m\}, m = \overline{1, M}$, элементами которого являются логические (булевы) функции. Каждому элементу множества *A* сопоставлен только один элемент множества *F*. Совокупность множеств *A* и *B,* совместно с однозначным отображением множества *A* на множестве *F* называется *логической сетью*. Множество входов логической сети обозначается как $C = \{c_1, ..., c_n\}$, а множество выходов как $Y = \{y_1, y_2, ..., y_q\}$. Если множество *C* содержит *n* элементов, а множество *Y* содержит q элементов, то такую логическую сеть называют логическим
(n, q)-полюсником [72].

Геометрической интерпретацией логической сети является ориентированный граф, который строится следующим образом. Сначала на плоскости размещают элементы множества A в виде кружков – вершин графа с нумерацией. Внутри кружка указывается элемент множества F, определяющий логическую функцию, выполняемую этим элементом, затем вершины соединяются ориентированными дугами согласно элементам множества B. Дополнительно в виде жирных точек указывают входы и выходы логической сети (рисунок 4.7).

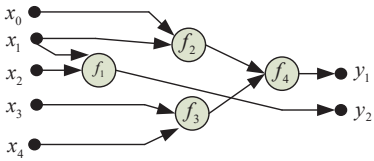


Рисунок 4.7 – Пример графа логической сети с пятью входами и двумя выходами

В качестве булевых функций на приведенном рисунке могут иметь место, например, следующие[17]:

$$f_1 = x_1\overline{x_0},\quad f_2 = x_1x_2, f_3 = x_3 \vee x_4; f_4 = f_1 \vee f_2 . \qquad (4.10)$$

При моделировании процессов реализации угроз, как правило, достаточным оказывается использование $(n,1)$-полюсников или $(n,2)$-полюсников. Пример такой логической сети, описывающей логику реализации атаки проникновения в операционную среду атакуемого хоста «Туннелирование трафика», приведена на рисунке 4.8.

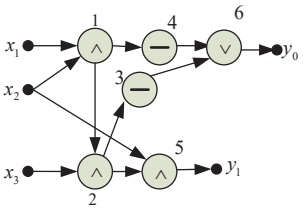


Рисунок 4.8 – Граф логической сети, моделирующей логику выполнения атаки «Туннелирование трафика»: $\wedge$ – знак логического умножения (выполнение логического условия «И»); $\vee$ – знак логического сложения (выполнения логического условия «ИЛИ»), «–» – знак логического отрицания (выполнения логического условия «НЕ»)

[17] Черта над буквой означает отрицание, то есть, если $x_0 = 1$, то $\overline{x_0} = 0$.

В этой сети x_1 – параметр, отражающий передачу пакета на доверенный хост, содержащего инкапсулированный пакет для атакуемого хоста с кодом (скриптом) программы, выполнение которой обеспечивает несанкционированный доступ в его операционную среду с заданного сетевого адреса ($x_1 = 1$ для пакета с инкапсулированной программой); x_2 – параметр, отражающий факт успешной аутентификации атакующего хоста на доверенном хосте, при этом осуществляется прием и распаковка пакета, содержащего инкапсулированный пакет со скриптом для атакуемого компьютера ($x_2 = 1$ при успешной аутентификации и распаковке пакета для доверенного хоста); x_3 – параметр отражающий факт передачи ($x_3 = 1$) сообщения с атакующего хоста на атакуемый хост.

При реализации атаки может быть два исхода $y_0 = 1$ – атака не удалась и $y_1 = 1$ – сообщение (команда, вредоносная программа и т.п.), принятое на атакуемом компьютере исполнено. Логическая функция такой сети для варианта успешной реализации атаки представляется следующим образом:

$$f_1 = \left(\left(x_1 \wedge x_2\right) \wedge x_3\right) \wedge x_3, \tag{4.11}$$

а для срыва атаки –

$$f_1 = \overline{\left(\left(x_1 \wedge x_2\right) \wedge x_3\right)} \vee \overline{x_1 \wedge x_2}. \tag{4.12}$$

Следует подчеркнуть, что далеко не все процессы моделируются с использованием логических сетей даже в тех случаях, когда определяющим в выполнении процессов являются логические условия. Так, как правило, оказывается невозможным использование логических сетей при моделировании циклических процессов, когда выход из цикла осуществляется при выполнении определенного условия, например, переполнении буфера при получении атакуемым компьютером количества пакетов, превышающего предельно допустимое. В этом отношении более продуктивным является применение аппарата сетей Петри (см. раздел 4.4).

Вместе с тем аппарат логических сетей начал успешно применяться при решении задачи логического вывода в ходе доказательства возможности эксплуатации уязвимостей «нулевого дня» при реализации тех или иных угроз безопасности информации в ИС. Кроме того, появились расширения логических сетей, такие как, например, логические сети с предикатными операциями[18] [56].

В соответствии с [56] модель процесса в общем виде может быть представлена системой бинарных предикатов:

$$M^B = \left\{ R_i \middle| \forall R_i \exists B_k, R_i \in M, B_k \in B, k = \overline{1,K}, i = \overline{1,I} \right\}, \tag{4.13}$$

где R_i – бинарные предикаты, а B_k – объекты предметной области.

В этой модели введены два типа узлов логической сети. Вершины первого типа $x \in V^*$ отражают конкретные действия (процедуры) процесса.

[18] Иногда такие сети называют предикатными сетями

С каждой вершиной связаны определенные действия процесса с помощью бинарных предикатов:

$$\forall x_i \in V^* \exists! x_j, R_k(x_i, x_j), (i, j = \overline{1, I}), k = \overline{1, K}\,. \qquad (4.14)$$

Соответственно, модель (4.13) дополняется следующим образом:

$$M^B = \left\{ R_k \mid \forall R_k(x_i, x_j) \in M, x_i \in V^*, (i, j = \overline{1, I}), \exists B_k \in B, k = \overline{1, K} \right\}. \qquad (4.15)$$

Вершина называется вершиной второго типа x′∈V**, если из вершины x′ графа логической сети выходит две дуги.

Адаптация модели к предметной области достигается за счет применения операторов: последовательного выполнения или импликации (знак «→»), эксклюзивного выбора (знак «XOR», исключающее «ИЛИ»), неэксклюзивного выбора (знак «OR» – логическая операция «ИЛИ»), параллельного выполнения (знак «AND» – логическая операция «И») и цикла (знак «Ω»):

$$\rightarrow (R_k(x', x_i), R_i(x', x_j))\,; \qquad (4.16)$$

$$XOR(R_k(x', x_i), R_i(x', x_j)) = R_k(x', x_i) \oplus R_i(x', x_j)\,; \qquad (4.17)$$

$$OR(R_k(x', x_i), R_i(x', x_j)) = R_k(x', x_i) \vee R_i(x', x_j)\,; \qquad (4.18)$$

$$AND(R_k(x', x_i), R_i(x', x_j)) = R_k(x', x_i) \wedge R_i(x', x_j)\,; \qquad (4.19)$$

$$\Omega(R_k(x', x_i), R_i(x', x_j), n_\Omega)\,, \qquad (4.20)$$

где n_Ω – количество повторов в цикле;

В полученной конструкции используются предикатные операции, то есть предикаты второго порядка, поэтому такая сеть в [56] названа логической сетью с предикатными операциями.

Графически, за счет применения предикатных операций, логическая сеть из плоской становится объемной (рисунок 4.9).

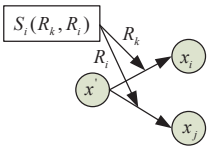


Рисунок 4.9 – Логическая сеть с предикатной операцией $S_i(R_k, R_i)$ для исходящих дуг

Это имеет физический смысл, так как отражает реальное разделение процедур процесса на элементарные базовые действия и операции над этими процедурами.

Для операторов последовательного выполнения и цикла важно задать порядок вершин (слева-направо) вне зависимости от их типа, поскольку порядок вершин задает порядок выполнения действий процесса. В случае

цикла левая вершина соответствует началу цикла, содержащему условие повторения, а правая – собственно операции (группе операций) цикла, после которых происходит переход к новому повторению (рисунок 4.10).

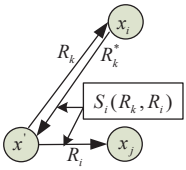


Рисунок 4.10 – Логическая сеть с предикатной операцией цикла

При этом предикатная операция цикла выполняется следующим образом:

$$S_1\left(R_k\left(x',x_i\right),R_i\left(x',x_j\right)\right)=\Omega\left(R_k\left(x',x_i\right),R_i\left(x',x_j\right)\right)= \\ =\left[R_k\left(x',x_i\right)\wedge R_i\left(x',x_j\right)\right]^v\wedge R_i\left(x',x_j\right), \tag{4.21}$$

то есть сначала v раз выполняются предикаты $R_k\left(x',x_i\right)$ и $R_k^*(x_i,x')$, а после завершения циклов – предикат $R_i\left(x',x_j\right)$.

В целом модель процесса представляется в виде иерархии предикатных операций вида (4.16) – (4.20) на верхних уровнях, а также бинарных предикатов на нижнем уровне:

$$M=\langle S_k,R_i\rangle,R_k\left(x_i,x_j\right)\in M,k=\overline{1,K},x_i\in\left(V^*\cup V^{**}\right), \\ \left(i,j=\overline{1,I}\right),S_l\left(R_i,R_j\right)\in S,l=\overline{1,L}. \tag{4.22}$$

Вместе с тем в [56] предикатные операции рассматриваются только для исходящих дуг, что вполне оправдано при моделировании процессов трассировки программ. Однако во многих практически важных случаях необходимо вводить предикатные операции для нескольких дуг, входящих в узел сети (рисунок 4.11).

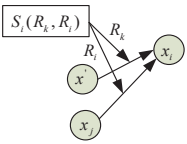


Рисунок 4.11 – Логическая сеть с предикатной операцией для исходящих дуг

В этом случае предикатные операции «ИЛИ», исключающего «ИЛИ», и «И» соответственно определяются следующим образом:

$$XOR(R_k(x',x_i),R_i(x_j,x_i))=R_k(x',x_i)\oplus R_i(x_j,x_i)\,; \tag{4.23}$$

$$OR(R_k(x^{'}, x_i), R_i(x_j, x_i)) = R_k(x^{'}, x_i) \vee R_i(x_j, x_i)\ ; \qquad (4.24)$$

$$AND(R_k(x^{'}, x_i), R_i(x_j, x_i)) = R_k(x^{'}, x_i) \wedge R_i(x_j, x_i). \qquad (4.25)$$

Исходя из общих представлений о процессе, полная модель процесса обязательно должна иметь одно начальное и одно конечное состояние, а также содержать пути достижения конечного состояния.

Пример 4.4. Пусть необходимо смоделировать логику реализации атаки «Подмена доверенного объекта», направленную на проникновение в операционную среду атакуемого хоста от имени доверенного хоста (с его сетевым адресом) с предварительным выводом из строя доверенного хоста путем атаки «шторм TCP-запросов» [26].

Логическая сеть с предикатными операциями, моделирующая логику выполнения такой атаки, приведена на рисунке 4.12.

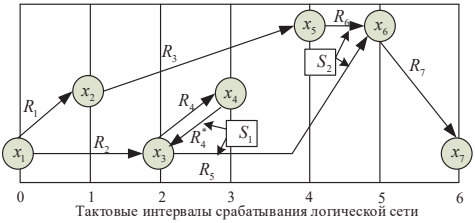


Рисунок 4.12 – Логическая сеть с предикатными операциями, моделирующая логику реализации атаки «Подмена доверенного объекта»

Принятые на рисунке 4.12 обозначения приведены в таблице 4.7.

Следует подчеркнуть, что в моделях процессов на основе логических сетей с предикатными операциями может условно учитываться и фактор времени, если установить время выполнения каждого такта, однако на практике это не применяется, так как при этом требуется разработка временных моделей выполнения предикатных операций. Поэтому такие модели применяются тогда, когда нужно:

– обосновать путем логического вывода принципиальную возможность реализации моделируемого процесса;

– оптимизировать выбор вариантов реализации исследуемого процесса на формальной основе, когда экспертным путем решение такой задачи оказывается невозможным из-за большого количества таких вариантов, громоздкости их описания, сложностей сопоставления и др.

Таблица 4.7 – Обозначения, принятые в логической сети с предикатными операциями, моделирующей логику реализации атаки «Подмена доверенного объекта» (рисунок 4.12)

Обозначение	Наименование, формула	Значение
x_1	Действие	Нарушитель определяет целевой хост, в операционную среду которого собирается проникнуть, состав доверенных для него хостов и выбирает доверенный хост-«жертву», от имени которого собирается атаковать целевой хост
x_2	То же	Нарушитель определяет параметры сетевого трафика целевого хоста (номера пакетов, динамику выдачи квитанций на запросы о сеансе связи и др.)
x_3	–	Нарушитель направляет на хост-«жертву» «шторм TCP-запросов на соединение» и фиксирует факт прекращения поступления квитанций от него, свидетельствующей о «зависании» его операционной системы
x_4	–	Хост-«жертва» на каждый пакет с запросом выдает квитанцию о готовности к соединению (созданию виртуального канала) и записывает указанные в пакете данные о компьютере нарушителя в свой буфер памяти до тех пор, пока буфер не переполнится из-за отсутствия подтверждения со стороны нарушителя и невозможности в связи с этим очищения буфера
x_5	–	Нарушитель посылает целевому компьютеру запрос на соединение от имени вышедшего их строя доверенного хоста
x_6	–	Нарушитель получает доступ в операционную среду целевого компьютера и дает, например, команду на предоставление режима удаленного терминала для компьютера с нужным ему сетевым адресом
x_7	–	Реализация угрозы выполнения несанкционированных действий с защищаемой информацией (копирование, модификация, уничтожение и т.д.)
$R_1(x_1, x_2)$	Предикат $R_1(x_1, x_2) = \exists b_c \in B_k, k = \overline{1, K}, \rightarrow X_{b_c}$	Определяет следующее утверждение: «существует хост b_c, который выбран в качестве целевого, и для него выявлены характеристики X_{b_c}, необходимые для реализации атаки «Подмена доверенного объекта»
$R_2(x_1, x_3)$	Предикат $R_2(x_1, x_3) = \forall b_j, j = \overline{1, K_j}, K_j \in K,$ $b_j \in B_k, k = \overline{1, K}, (\exists b_g, g \in K_j \rightarrow X_{b_g})$	Определяет следующее утверждение: «из всех хостов $b_j, j = \overline{1, K_j}$, взаимодействующих с целевым хостом, выбран хост-«жертва» b_g, от имени которого будет осуществляться атака на целевой хост и для которого определены параметры X_{b_g}, необходимые для атаки «шторм TCP-запросов» для вывода его из строя
$R_3(x_2, x_6)$	Предикат $R_3(x_2, x_6) = \exists X_{b_c} \rightarrow d_{b_c}(\psi_1)$	Определяет утверждение, что «имеются необходимые характеристики X_{b_c} целевого хоста и ему отправлен пакет $d_{b_c}(\psi_1)$, содержащий информацию ψ_1 первого типа – запрос на соединение»

Обозначение	Наименование, формула	Значение
$R_4(x_3,x_4)$	Предикат $R_4(x_3,x_4)=\exists X_{h_y} \to d_{h_y}(\psi_1)$	Определяет утверждение, что «имеются необходимые характеристики X_{h_y} хоста-«жертвы» и ему отправляются пакеты $d_{h_y}(\psi_1)$, содержащие информацию ψ_1 первого типа – запрос на соединение в цикле, пока количество пакетов превысит допустимое значение, при котором буфер хоста-«жертвы» переполняется и его операционная система «зависает»
$R_4^*(x_3,x_4)$	Предикат $R_4^*(x_4,x_5)=\exists d_{h_y}(\psi_1) \to d_{h_y}(\psi_2)$	Определяет утверждение, что «получен пакет с запросом на соединение $d_{h_y}(\psi_1)$ хостом-«жертвой» и им отправлен хосту нарушителя пакет, содержащую информацию ψ_2 второго типа – квитанцию о готовности к формированию виртуального канала». Отправка квитанций выполняется в цикле по получении пакетов с запросами на соединение
$R_5(x_3,x_4)$	Предикат $R_5(x_3,x_4)=(v_{h_y} > v_{cr})$	Определяет утверждение, что «хост-«жертва» выведен из строя», так как количество пакетов с запросами на соединение от хоста нарушителя v_{h_y} больше критического значения v_{cr}, при котором количество записей данных о запросившем соединение хосте переполняет буфер
$R_6(x_4,x_5)$	Предикат $R_6(x_6,x_7)=d_{sh_1}(\psi_1) \to d_{h_y*}(\psi_2) \to d_{sh_1}(\psi_3)$	Определяет утверждение, что «виртуальный канал между целевым хостом и якобы функционирующим доверенным хостом» установлен, то есть проведена полная процедура троекратного «рукопожатия», при этом целевой хост получил запрос на соединение $d_{sh_1}(\psi_1)$, отправил квитанцию о готовности на доверенный хост $d_{h_y*}(\psi_2)$ и получил квитанцию от нарушителя от имени доверенного хоста $d_{sh_1}(\psi_3)$ о начале взаимодействия
$R_7(x_3,x_7)$	Предикат $R_9(x_7,x_8)=\exists d_{sh_1}(\psi_3) \to d_{sh_1}(\psi_4)$	Определяет утверждение, что «нарушитель получил доступ в операционную среду целевого хоста и имеет возможность выполнить несанкционированные действия с защищаемой информацией
S_1	Предикатная операция $S_1=\Omega(R_4(x_3,x_4),R_5(x_3,x_6),v_{cr})=$ $=(R_4(x_3,x_4),R_4^*(x_4,x_5))^{v_{cr}} \wedge R_5(x_3,x_6)$	Предикатная операция цикла, реализуемая следующим образом: сначала не менее v_{cr} раз выполняется цикл, включающий в себя передачу нарушителем пакетов на соединение с хостом-«жертвой» и передачи хостом-«жертвой» пакетов с ответной квитанцией о готовности к соединению. После завершения цикла выполняется переход к вершине x_6 в соответствии с предикатом $R_5(x_3,x_6)$
S_2	Предикатная операция $S_2=R_4(x_3,x_4) \wedge R_5(x_3,x_5)$	Предикатная операция логического умножения (логическая операция «И»)

4.5. Сети Петри и их применение при оценке возможностей реализации угроз безопасности информации

Дальнейшим развитием теории конечных автоматов стал переход к методам анализа и синтеза дискретных параллельных систем, среди которых выделились методы, основанные на использовании аппарата сетей специального вида, предложенных Карлом Петри [73] и названных в честь него сетями Петри.

Характерными примерами таких не описываемых в рамках классической теории автоматов параллельных систем, в которых отдельные компоненты функционируют, в основном, независимо, взаимодействуя друг с другом время от времени, являются многопроцессорные вычислительные системы, мультипрограммные операционные системы, асинхронные электронные схемы и т.п.

Следует подчеркнуть, что реальные системы функционируют во времени, то есть события происходят в некоторые моменты времени и длятся некоторое время. В дискретных системах события явно привязаны к определенным моментам и интервалам времени, но в ряде таких систем события могут происходить внутри больших интервалов времени, при этом часто выходом может служить отказ от введения времени и тактированных последовательностей изменения состояний моделируемой системы с заменой их причинно-следственными связями между событиями. Модели такого типа получили название асинхронных. Замена временных связей причинно-следственными дает возможность более наглядно описать структурные особенности функционирования систем [74].

Процесс взаимодействия в больших асинхронных системах, как правило, имеет сложную динамическую структуру. Для упрощения их описания в моделях указываются не непосредственные связи между событиями, а те ситуации (определенные сочетания условий), при которых данное событие может реализоваться.

В связи с этим дискретные системы представляются как структуры, образованные из элементов двух типов – событий и условий.

В сетях Петри события и условия представлены абстрактными символами из двух непересекающихся алфавитов, называемых соответственно множеством переходов (обычно переходы обозначаются буквой t) и множеством мест или позиций (обозначается буквой p).

Позиции и переходы связаны отношением непосредственной зависимости (непосредственной причинно-следственной связи), которая изображается с помощью направленных дуг. Позиции, из которых ведут дуги на переход, называются входными позициями данного перехода, а позиции, к которым ведут дуги из данного перехода, называются выходными для данного перехода

Выполнение условия отображается разметкой соответствующей позиции с помощью помещения в нее маркеров (фишек), при этом количество маркеров в позиции указывает на емкость условия. В процессе

функционирования сети Петри происходит смена разметок позиций как результат срабатывания ее переходов. Сеть останавливается, если ни один из ее переходов не может сработать.

Графическим представлением сети Петри служит двудольный ориентированный граф с двумя типами вершин, при этом вершины-позиции изображаются кружочками, а вершины-переходы – барьерами (планками). В сети Петри допускается существование кратных дуг, в этом случаем графическим ее представлением является мультиграф (рисунок 4.13).

Формально сеть Петри определяется набором $C = \{P, T, F, M\}$, где P – непустое множество позиций сети, T – непустое множество переходов сети, F – отношение инцидентности, то есть отображение переходов в комплекты позиций и наоборот $F \subseteq P \times T \cup T \times P$, M_0 – функция начальной разметки сети.

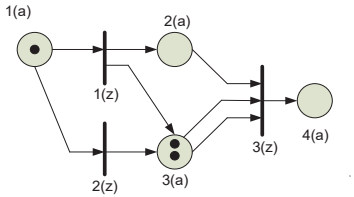


Рисунок 4.13 – Пример графа сети Петри[19]

Функционирование сети Петри описывается формально с помощью множества последовательностей срабатываний переходов и множества достижимых в сети разметок. Переход может сработать при некоторой разметке M, если каждое входное место перехода t имеет разметку с количеством маркеров (фишек), не меньшим, чем кратность дуги, соединяющей позицию p и переход t. Формально это записывается следующим образом:

переход $t_j \in T$ в маркированной сети Петри с маркировкой $\mu \in M$ разрешен, если для всех $p_i \in P$ выполняется условие $\mu(p_i) \geq w(p_i, I(t_j))$, где w – количество дуг (с учетом кратности), входящих в переход t_j из позиции p_i.

При этом переход запускается удалением всех разрешающих фишек из его входных позиций (то есть разметка каждой его входной позиции p_i уменьшается на величину кратности дуги, соединяющей позицию p_i и переход t_j) и последующим помещением в каждую из его выходных позиций по одной фишке для каждой дуги. Сеть Петри считается ординарной, если кратность всех ее дуг не превышает единицы.

[19] Обозначение позиций и переходов соответствует принятым для описания сетей Петри-Маркова (см. раздел 6).

Разметка M' достижима из разметки M, если существует последовательность разметок $M, M_1, M_2, ..., M'$ и слово $\tau = t_1 t_2 ... t_k$ в алфавите T такие, что имеет место отношения $[\ \rangle$ непосредственного следования $M[t_1\rangle M[t_2\rangle ... [t_k\rangle M'$.

Множество всех последовательностей срабатываний сети Петри называют свободным языком сети Петри, при этом рассматриваются все разметки сети, достижимые из начальной разметки M_0 и составляющие множество $R(C, M_0)$. Язык сети Петри используется при ее анализе в интересах выявления характеристик моделируемой системы.

В ходе такого анализа часто важным для практики оказывается выявление наличия таких свойств сети как ограниченность и безопасность, достижимость и живость (активность переходов). Достаточно полно характеристика этих свойств приведена в [74].

Аппарат сетей Петри быстро привлек внимание специалистов, занимающихся вопросами моделирования систем. Однако скоро стало понятно, что классические сети Петри имеют определенные ограничения, не позволяющие адекватно моделировать многие системы. В результате появились многочисленные модификации этого аппарата и разнообразные подклассы сетей Петри. Краткая характеристика некоторых из них приведена в таблице 4.8.

Аппарат сетей Петри нашел применение и при решении вопросов технической защиты информации в ИС. Сегодня уже сформировалось несколько направлений использования этого аппарата в данной области, к которым относятся следующие.

1. Анализ программных кодов на предмет выявления в них уязвимостей, которые могут быть использованы для реализации угроз безопасности информации. В частности, уже в первых работах [74, 75] по практическому применению сетей Петри было предложено их использование при моделировании процессов анализа кодов в интересах выявления недопустимых конструкций кода, тупиков, непредусмотренных циклов, приводящих к зависанию программы и т.д., а в [76] изложен один из подходов к применению сетей Петри для решения задач анализа структурной динамики параллельных программ. Вместе с тем разработка методов анализа программного обеспечения с использованием сетей Петри в интересах оценки возможностей реализации угроз безопасности информации в ИС находятся в самой начальной стадии.

2. Моделирование процессов обнаружения сетевых атак на ИС. Так, в [77] предложено использовать для реализации алгоритма обнаружения сетевых атак формализм раскрашенных сетей Петри. В такой модели раскрашенная сеть Петри описывает сценарий атаки, при этом каждая сигнатура вторжения соответствует экземпляру сети Петри, которая может иметь более одного начального состояния, что обеспечивает частичную

Таблица 4.8 – Краткая характеристика некоторых подклассов сетей Петри

Подкласс сети Петри	Определение сети	Достоинства сети	Недостатки сети
Ординарная сеть Петри	Ординарной называют сеть Петри, у которой кратность любой позиции равна 0 или 1, то есть выполняется условие: $w(p_i, I(t_j)) \le 1$ и $w(p_i, O(t_j)) \le 1$ и для нее количество фишек в любой позиции не превышает 1. Если в сети Петри никакая позиция не может быть одновременно входом и выходом моделируемого процесса, то такую сеть называют сетью без петель или нерефлексивной сетью. Если ординарная сеть является нерефлексивной, то ее называют *простой сетью* Петри.	Позволяет достаточно просто моделировать многие процессы	Не позволяет моделировать процессы со сложными условиями, пересекающимися ветвями, стохастические процессы и учитывать фактор времени
Автоматная сеть Петри	Простая сеть Петри, в которой каждый переход имеет одно входное и ровно одно выходное место и через него может переместиться только одна фишка, то есть выполняется условие $\forall t \in T: w(I(t)) = 1$ и $w(O(t)) = 1$. Автоматная сеть безопасна тогда и только тогда, когда ее начальная разметка содержит ровно одну фишку	Автоматная сеть допускает моделирование так называемого конвейерного параллелизма, когда несколько фишек независимо перемещаются по сети, а также отображать конфликтные ситуации, когда позиция является входной или выходной для нескольких переходов	Не позволяет моделировать процессы со сложными условиями, стохастические процессы и учитывать фактор времени
Сеть Петри «Синхронизированный граф»	Простая сеть Петри, в которой в каждую позицию входит только одна дуга и из каждой позиции выходит только одна дуга, то есть выполняется условие: $w(p_i, I(t_j)) = 1$ и $w(p_i, O(t_j)) = 1$	Позволяет описывать несложные параллельные процессы	Не позволяет моделировать конфликтные ситуации, стохастические процессы и учитывать фактор времени
Свободные сети Петри (сеть Петри со свободным выбором)	Сеть Петри, в которой любая дуга, ведущая от позиции к переходу, или начинается позицией, из которой не выходит ни одной другой дуги, или заканчивается переходом, в которые не ведет никакая дуга. Класс свободных сетей является подклассом ординарных сетей и строго включает в себя классы автоматных сетей и синхрографов	Позволяет исследовать на безопасность многие несложные процессы, учитывать: наличие *ловушек* – позиций, которые одновременно являются входными и выходными для перехода (если ловушка размечена, то есть содержит хотя бы одну фишку, то при функционировании сети она всегда остается размеченной, так как любая фишка, попавшая в ловушку, не может исчезнуть из нее); наличие *тупиков* –позиций, помещая в каждую из которых фишку, соответствующий переход должен забирать по крайней мере одну фишку из нее (если тупик пуст, то соответствующий переход мертв). В сильно связных сетях позиция может быть одновременно и ловушкой, и тупиком	Те же, что и для ординарных сетей
Сеть Петри «маркированный граф»	Это сеть Петри, в которой каждая позиция является входом для точно одного перехода и выходом точно оного перехода. Двойственна автоматной сети Петри, так как в автоматной сети переходы имеют один вход и один выход, в маркированном графе позиции имеют один вход и один выход	Те же, что и в автоматных сетях Петри	Те же, что и в автоматных сетях Петри

Подкласс сети Петри	Определение сети	Достоинства сети	Недостатки сети
Ингибиторная сеть Петри	Это сеть Петри, дополненная специальной функцией инцидентности $F_i : P \times T \to \{0,1\}$, которая вводит ингибиторные дуги («сдерживающие дуги» [74]). Она обеспечивает возможность проверки на нулевую разметку, что не позволяют делать обычные сети. Ингибиторная дуга обозначается не стрелкой на конце, а кружком. Переход t может сработать, если каждая его входная позиция p, соединенная с t обычной дугой с кратностью $w(p,t)$, содержит не менее $w(p,t)$ фишек, а каждая входная позиция, соединенная с переходом t ингибиторной дугой (ее кратность всегда равна 1), имеет нулевую разметку, что формально записывается следующим образом: $\forall p_i, t_j : M(p_j) \geq F(p_i, t_j) \wedge M(p_i) \cdot F_i(p_i, t_j) = 0$	Значительно расширяет возможности моделирования процессов, так как позволяет проверять позиции на нулевую разметку, управлять срабатываем переходов путем ввода условий их срабатывания	Не позволяет учитывать фактор времени
Сеть Петри с приоритетами	Это сеть Петри, дополненная частично упорядоченным множеством приоритетов PR, каждый из которых связан с определенным переходом $pr(t) \in PR$. Переход t может сработать, если для любого другого перехода t', который может сработать по стандартному условию, выполняется соотношение $pr(t) \geq pr(t')$	Позволяет моделировать функционирование дискретных систем, в которых необходим неодновременный запуск нескольких устройств со строго определенной последовательностью. Равносильна по возможностям моделирования ингибиторным сетям	То же
Временная сеть	Это сеть Петри, в которой переходы обладают весом, определяющим продолжительность их срабатывания (задержку). Равносильна сети с приоритетом и ингибиторной сети. Разновидностью временных сетей являются стохастические сети, в которых задержки в срабатывании переходов являются случайными	Позволяет моделировать функционирование систем с учетом временного фактора	Большая громоздкость учета в модели временного фактора, сложность учета временных параметров парциальных процессов, составляющих моделируемый процесс. Невозможность учета времени срабатывания логических стохастических переходов
Раскрашенная (цветная) сеть Петри	Это сеть Петри [74], фишкам которой приписывают дополнительные атрибуты, которые названы цветами. Условия срабатывания переходов и правила изменения разметки сети задаются специальными таблицами, учитывающими цвета фишек. В каждой такой таблице столбцы, связанные с входными позициями перехода, содержат сочетания конкретных цветных фишек, при которых переход может сработать. Столбцы, связанные с выходными позициями, указывают цвета фишек, которые добавляются в выходные позиции для каждого входного сочетания цветных фишек	Раскрашенная сеть равномощна ингибиторным сетям и сетям с приоритетом	Те же, что и для ингибиторных сетей

Подкласс сети Петри	Определение сети	Достоинства сети	Недостатки сети
Самомодифицируемая сеть Петри	В соответствии с [74] в сетях данного класса каждой дуге (p,t) и (t,p) приписана модифицируемая кратность. В качестве кратности может выступать не только число, как в определении сетей Петри, но и символ (например, q) некоторой позиции данной сети, в этом случае кратность дуги переменна и равна текущей разметке $M(q)$ позиции q. Поскольку разметка позиции q динамически меняется при работе сети, число фишек, перемещаемых одним и тем же переходом, динамически меняется при разных срабатываниях этого перехода	Введение модифицируемой кратности позволяет достаточно просто моделировать ингибиторные сети и сети с приоритетами	Те же, что и для ингибиторных сетей
Иерархическая сеть Петри	Эта сеть, наряду с неделимыми (атомарными) компонентами, содержит составные компоненты, сами представляющие собой системы.	Является существенным обобщением сетей Петри. Позволяет моделировать процессы в иерархических структурах	В практике моделирования процессов реализации угроз в ИС не применялась
WF-сети	WF-сети – это подкласс сетей Петри, называемый также сетями потоков работ. Сеть Петри называется сетью потоков работ (WF-сетью), если выполняются следующие условия: существует только одна исходная позиция, такая что отсутствуют входящие переходы, инцидентные этой позиции; существует только одна конечная позиция, такая что отсутствуют выходящие переходы, инцидентные этой позиции; каждый узел данной сети расположен на пути от указанных исходной и конечной позиций [78]	WF-сети позволяют проверять графы потоков работ на наличие структурных конфликтов, таких как наличие тупиков, что имеет место, если: конечная позиция достижима при любой последовательности переходов от исходной позиции; WF-сеть не содержит лишних позиций (которые никогда не будут выполнены); при достижении конечной позиции не должно оставаться фишек в промежуточных позициях	То же
E-сеть	E-сети (от англ. Evaluation – «вычисления», «оценка») – «оценочные сети», относящиеся к подклассу сетей Петри [79]. В E-сетях: – имеются несколько типов вершин-позиций: простые позиции, позиции-очереди, разрешающие позиции; – фишки могут снабжаться набором признаков (атрибутов); – с каждым переходом может быть связана ненулевая задержка и функция преобразования атрибутов фишек; – введены дополнительные виды вершин-переходов; – в любую позицию может входить не более одной дуги и выходить также не более одной	Применяются для оценки производительности систем, моделирования динамики выполнения процессов	Не позволяет в должной мере исследовать динамику процесса, поскольку временные его характеристики вводятся как исходные данные, а не вычисляются по результатам срабатывания сети

упорядоченность, и единственное конечное состояние. Сигнатура считается распознанной при достижении конечного состояния сети. С переходом могут быть связаны условные выражения, определяющие логические условия его срабатывания. С сетью Петри связано множество переменных.

Каждый маркер поддерживает собственную копию переменных в связи с тем, что каждый маркер может связать различные переменные в процессе распознавания (в терминах раскрашенной сети Петри каждый маркер является цветным; при этом цвет может рассматриваться как кортеж, состоящий из n строк, если сигнатура имеет n переменных).

Хотя использование данного метода позволяет обнаруживать широкий спектр сетевых атак, он требует значительных вычислений (время распознавания экспоненциально по отношению к размеру раскрашенных сетей Петри), и пока не может использоваться в системах обнаружения атак реального времени.

3. Моделирование межсетевых протоколов передачи сообщений в компьютерных сетях, в том числе в интересах обеспечения безопасности передаваемых по сети пакетов. Так, в [80] описано использование аппарата сетей Петри для выявления ошибок в протоколах, наличия тупиков, анализа выполнимости процедур проверки пакетов с неверными номерами, наличие недопустимых циклов обработки пакетов и т.д.

4. Моделирование процессов защиты информации с применением разнообразных мер и средств ТЗИ. Так, в [81] сети Петри использовались для имитационного моделирования и исследования эффективности ложных информационных систем, используемых для отвлечения нарушителя на ложный информационных ресурс. Такие модели могут создаваться также в интересах исследования программных способов разграничения доступа к процедурам, функциям, модулям программ, файлам, каталогам и директориям, полям и записям в базах данных, к программно-аппаратным элементам ИС.

Вместе с тем, аппарат сетей Петри в основном применяется при имитационном моделировании ряда процессов, связанных с реализацией угроз безопасности информации, с возможностью применения некоторых мер защиты, с получением с использованием имитационных моделей статистических данных, необходимых для оценки эффективности средств защиты и др.

4.6. Метод анализа возможностей реализации угроз на основе теории графов

Одним из вариантов решения задачи оценки достижимости объекта в составе ИС при реализации угрозы безопасности информации является применение аппарата теории графов, с помощью которых могут быть представлены возможные сценарии процесса реализации угроз. Чаще всего этот аппарат применяется сегодня при анализе сетевых атак в ИС [58, 82].

Анализ существующих методов [58–62], которые применяются для моделирования атак на ИС, показал, что для выявления возможных сочетаний действий, приводящих к воздействию на защищаемую информацию в ИС, наиболее приемлемым оказывается графовый метод. В этом случае не учитывается временной фактор, который, несомненно, существенно влияет на возможность проведения атак, однако для его учета сначала нужно определить всю совокупность действий, которая должна быть выполнена для проведения атаки по тому или иному сценарию в соответствии со структурно-функциональными особенностями защищаемой ИС.

При этом должно учитываться следующее.

1. Компьютерные атаки на ИС могут быть выполнены:

– во-первых, путем несанкционированного доступа в операционные среды хостов ИС;

– во-вторых, путем нарушения (изменения, блокирования) трафика, передаваемого в ИС или из ИС во внешние сети.

При несанкционированном доступе в операционные среды хостов ИС циркулирующая в них системная и прикладная информация (команды, функции, программы, данные, используемые протоколы обмена данными и др.) может быть уничтожена, модифицирована, подменена, заблокирована.

Изменение или блокирование трафика достигается заменой пакетов с передаваемыми по ИС данными и управляющей информацией, если такая передается, перегрузкой линий передачи данных и буферов обмена ими в хостах ИС (в том числе в коммуникационных элементах).

2. Доступ в операционную среду хостов ИС возможен из другого хоста (в том числе коммуникационного элемента), если предусмотрено взаимодействие между ними в ходе функционирования ИС. Это дает возможность несанкционированного доступа к атакуемому хосту через цепочку других хостов, в том числе тех, которые непосредственно не взаимодействуют с атакуемым хостом, при этом осуществляется несанкционированное проникновение в их операционные среды.

3. Доступ в операционную среду хоста ИС возможен из внешней сети (например, сети Internet), если имеется подключение к ней.

Далее полагается, что две атаки являются разными, если они различаются хотя бы одним из элементов описания: объектом атаки (атакуемым хостом), источником угрозы атаки, исходным хостом, с которого начинается атака (исходной точкой атаки), используемой уязвимостью и способом ее эксплуатации, а также содержанием несанкционированного (деструктивного) действия с защищаемой информацией.

При формировании графа, определяющего совокупность действий при выполнении атаки, сначала определяются возможные объекты атаки – хосты ИС. Далее выявляется защищаемая информация (системные файлы операционной системы, прикладные программы и данные и т.п.), относительно которой в ходе атаки могут быть выполнены несанкционированные (деструктивные) действия (например, уничтожение

системного файла, изменение исполняемого файла прикладной программы), в результате чего может нарушиться функционирование хоста или системы в целом. Затем определяется, какой источник угрозы атаки может иметь место: внешний или внутренний (см. раздел 1.3).

После определения указанных элементов описания атаки формируется путь или маршрут атаки, то есть последовательность хостов, которые могут быть задействованы при атаке. Для этого анализируется структурно-функциональная схема сети и выявляются хосты, которые могут быть подвержены атакам. Пример схемы офисной ИС, относительно которой могут иметь место атаки из сети Internet, приведен на рисунке 4.14.

Наиболее удобным и наглядным является представление маршрута атаки в виде ориентированного графа. Алгоритмы формирования графов маршрутов атак на хосты ИС в случаях, когда источник угрозы атаки является внешним или внутренним, приведены соответственно на рисунках 4.15 и 4.16.

При построении дерева маршрутов компьютерных атак на k-й хост ИС a_k, учитываются все атаки, возможность которых обусловлена наличием функциональной взаимосвязи атакуемого хоста с другими хостами ИС.

Такая взаимосвязь может быть представлена в виде ориентированного графа $G\left[G^{(1)},G^{(2)}\right]$, где $G^{(1)}=\left\{g_k^{(1)}\right\},k=\overline{1,K_1}$ – множество вершин графа (K_1 – количество вершин графа), $G^{(2)}=\left\{g_{nj}^{(2)}\right\},k=\overline{1,K_2},j=\overline{1,K_2}$ – множество ориентированных дуг (последовательность индексов n и j указывает направление дуги), при этом взаимосвязи определяются необходимостью приема и передачи сообщений (команд, данных, иной информации) между хостами ИС в ходе обеспечения ее функционирования (K_2 – количество ориентированных дуг в графе).

Применительно к ИС, приведенной на рисунке 4.14, дерево маршрутов атак с внешнего источника угрозы показано на рисунке 4.17, при этом полагается, что объектами атаки могут быть и межсетевой экран, и маршрутизаторы.

Далее в соответствии с графом $G\left[G^{(1)},G^{(2)}\right]$ выявляются все смежные элементы, из которых может быть получен доступ к хосту a_k – объекту атаки, с учетом выявленных уязвимостей.

Для этого используется матрица смежности $M_G=\left\{m_{kn}\right\},k=\overline{1,K},n=\overline{1,K}$ для графа $G\left[G^{(1)},G^{(2)}\right]$, где K – количество хостов ИС, с которых может проводиться атака. Пример графа и его матрицы смежности приведен на рисунке 4.18.

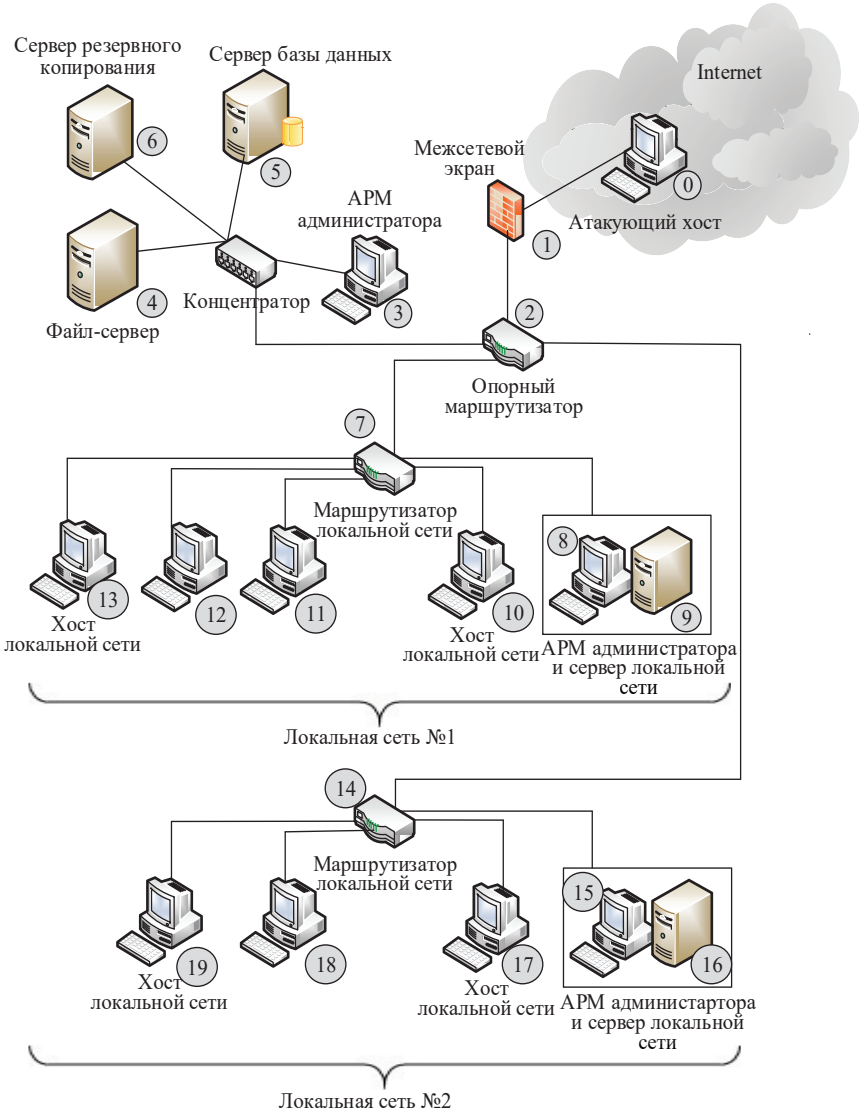


Рисунок 4.14 – Пример фрагмента офисной сети организации, атакуемой из глобальной сети Internet

Рисунок 4.15 – Алгоритм формирования дерева маршрутов атак на хосты ИС в случае внешнего источника угрозы атак

Рисунок 4.16 – Алгоритм формирования деревьев маршрутов атак на хосты ИС в случае внутреннего источника угрозы атаки

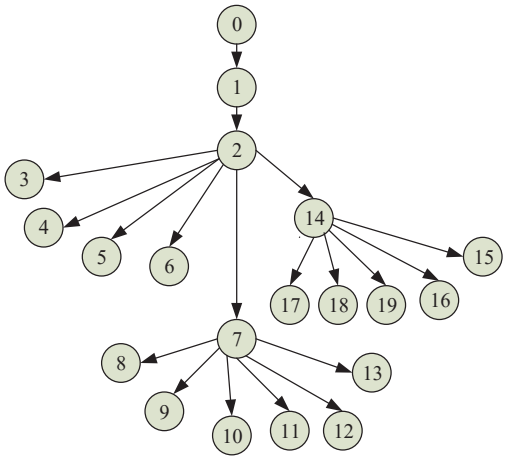


Рисунок 4.17 – Дерево маршрутов атак с хоста внешнего нарушителя на хосты ИС в соответствии со схемой, приведенной на рисунке 4.14

Формально правило, в соответствии с которым доступ в операционную среду хоста ИС из смежного хоста считается возможным, записывается следующим образом:

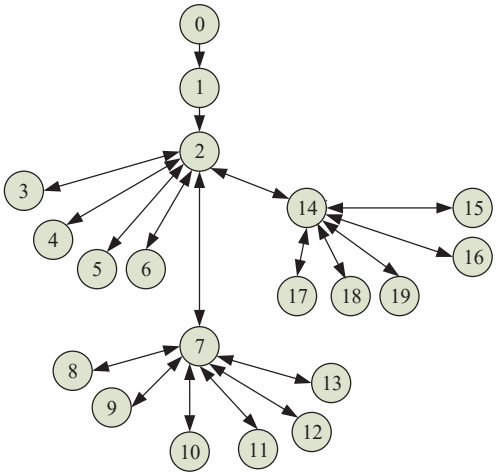
$$\exists a_n \left[m_{kn} \neq 0 \,\&\, \left\{ \exists h_k \left(h_k \in H_k, a_n \Rightarrow a_k \right) \right\} \right], n, k = \overline{K}, \tag{4.26}$$

где h_k уязвимость k-го хоста a_k из множества уязвимостей H_k (например, его системного или прикладного программного обеспечения), эксплуатация которой позволяет получить доступ из смежного хоста a_n в операционную среду хоста a_k.

То есть доступ возможен, если, во-первых, имеется связь двух хостов $\left(m_{kn} \neq 0\right)$ во-вторых, существует уязвимость в атакуемом хосте, позволяющая получить доступ в его операционную среду $\left(a_n \Rightarrow a_k\right)$ и имеется возможность эксплуатации этой уязвимости (использования соответствующего эксплойта, штатной программы или заблаговременно внедренной вредоносной программы).

На основе графа взаимосвязей для каждого хоста строится граф достижимости, полагая, что две вершины a_n и a_k являются связанными, если на ориентированном графе существует маршрут длиной $s\left(a_n, a_k\right)$, в котором концами являются вершины a_n и a_k, то есть имеется путь непосредственно из вершины a_n в вершину a_k.

Порядок построения графов достижимости для всех хостов сводится к следующему:



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1		1																	
2			1	1	1	1	1							1					
3		1																	
4		1																	
5		1																	
6		1																	
7		1						1	1	1	1	1	1						
8							1												
9							1												
10							1												
11							1												
12							1												
13							1												
14		1													1	1	1	1	1
15														1					
16														1					
17														1					
18														1					
19														1					

Рисунок 4.18 – Граф взаимосвязей и соответствующая ему матрица смежности хостов ИС, фрагмент схемы которой приведен на рисунке 4.14

1) для вершины a_k (соответствующей атакуемому хосту) выбираются все смежные вершины с номерами a_n, для которых элементы матрицы смежности отличны от нуля, то есть имеется путь к вершине a_k: $m_{nk} \neq 0$;

2) для каждой вершины a_n, смежной с вершиной a_k, аналогичным образом находятся свои смежные вершины $a_l, l = \overline{1, K}$, с выполнением условия $m_{nk} \neq 0$, при этом $a_l \neq a_k$;

3) с завершением охвата всех вершин, из которых есть путь к вершине a_k, строится граф достижимости для элемента, соответствующего этой вершине, представляющей дерево («обратное дерево»), в котором корнем является вершина a_k, а ветви соответствуют цепочкам – маршрутам от вершин – источников угроз к корневой вершине;

4) процедура повторяется для всех вершин графа взаимосвязей, в результате чего формируется множество деревьев, называемое лесом.

В качестве примера на рисунке 4.19 приведено одно из деревьев такого леса маршрутов для графа взаимосвязей, показанного на рисунке 4.18.

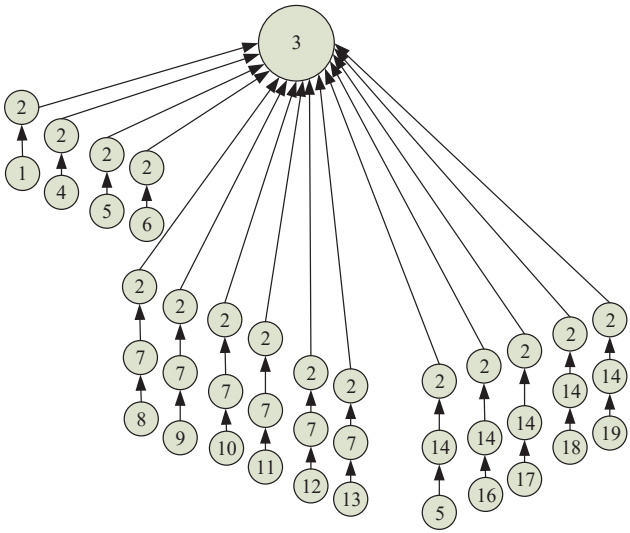


Рисунок 4.19 – Граф достижимости для хоста с номером 3
в соответствии со схемой ИС, приведенной на рисунке 4.14

Важным в графовом методе описания угроз проведения атак является:

– наглядность представления последовательности действий при реализации угроз;

– связь проводимых действий с аппаратными элементами ИС, через которые проходит атака и исключение из рассмотрения тех элементов, через которые она не проходит;

– возможность аналитической записи дерева и леса атак и достаточно простая программная реализация такой записи.

5. МОДЕЛИ ДИНАМИКИ РЕАЛИЗАЦИИ УГРОЗ С ИСПОЛЬЗОВАНИЕМ АППАРАТА МАРКОВСКИХ ПРОЦЕССОВ

5.1. Марковские цепи и их применение при моделировании динамики реализации угроз безопасности информации в информационных системах

Моделирование динамики процессов в предметной области ТЗИ, в том числе динамики реализации угроз, – это разработка моделей, в которых непосредственно учитывается фактор времени. Для такого моделирования сегодня имеется достаточно много методов (см. рисунок 4.1).

Однако наибольшее применение нашли методы теории марковских [35, 57] и полумарковских процессов [35, 83–85], построенные на их основе методы теории массового обслуживания [83, 86–88] и ее составной части – теории потоков [89, 90], а также аппарат теории сетей Петри-Маркова [91, 35] и его расширений.

Следует подчеркнуть, что каждая угроза безопасности информации в ИС реализуется по-своему и поэтому для каждой угрозы приходится разрабатывать свою математическую модель. Из-за этого, во-первых, сегодня отсутствует приемлемый по полноте набор таких моделей, во-вторых, постоянное расширение спектра угроз приводит к необходимости разработки все новых моделей. А это не только затруднительно, но и приводит к значительным сложностям их практического применения, так как обусловливает необходимость, наряду с созданием приемлемых описательных моделей, разрабатывать программные средства моделирования процессов их реализации, поскольку без автоматизации расчетов количественно оценить возможности реализации угроз крайне сложно. В данном разделе рассматриваются наиболее часто применяемые методы для моделирования динамики реализации угроз безопасности информации, основанные на аппарате марковских и полумарковских процессов.

Марковские процессы являются частным видом случайных процессов и впервые были весьма глубоко исследованы в трудах отечественных ученых А.А. Маркова и А.Н. Колмогорова. Достаточно полно теория марковских процессов изложена в [57], именно на основе этого труда В.И. Тихонова и М.А. Миронова далее приводятся основные положения этой теории.

В соответствии с [57] случайный процесс $\xi(t)$ называют марковским, если для любых n моментов времени $t_1 < t_2 < ... < t_n$ из отрезка $[0,T]$ условная функция распределения последнего значения $\xi(t_n)$ при фиксированных значениях $\xi(t_1),\xi(t_2),...,\xi(t_{n-1})$ зависит только от $\xi(t_{n-1})$, то есть при заданных значениях справедливо соотношение:

$$P\{\xi(t_n) \le \xi_n / \xi(t_1) = \xi_1, \xi(t_{12}) = \xi_2, ..., \xi(t_{n-1}) = \xi_{n-1}\} = P\{\xi(t_n) \le \xi_n / \xi(t_{n-1}) = \xi_{n-1}\}, \qquad (5.1)$$

где $P\{\bullet\}$ – вероятность события, указанного в фигурных скобках.

Как и другие случайные процессы, в которых множество состояний и область определения времени реализации процесса могут быть как непрерывными, так и дискретными различают марковские процессы с дискретным множеством состояний и с дискретным временем (дискретные процессы с дискретным временем, получившие название марковских цепей), марковские процессы с дискретным множеством состояний и с непрерывным временем (дискретные процессы с непрерывным временем) и марковские процессы с непрерывным множеством состояний и с непрерывным временем (непрерывные процессы с непрерывным временем). Последние пока не нашли применения при моделировании процессов в предметной области защиты информации и далее не рассматриваются.

Для марковских цепей количество фазовых состояний $\{\vartheta_k\}, k=\overline{1,K}$, моделируемой системы конечно, то есть составляет величину K, и система в дискретные моменты времени $t_1 < t_2 < \ldots$ случайным образом меняет свое фазовое состояние $\theta_0 \to \theta_1 \to \theta_2 \to \ldots$, при этом полное описание поведения системы за n шагов задается совместными вероятностями:

$$P(\theta_0, \theta_1, \theta_2, \ldots \theta_n) = P(\theta_0 = \vartheta_{k_0}, \theta_1 = \vartheta_{k_1}, \theta_2 = \vartheta_{k_2}, \ldots, \theta_n = \vartheta_{k_n}) \tag{5.2}$$

при разных n и θ_n

Если ввести условную вероятность перехода системы из состояния в состояние на μ-м шаге:

$$\pi_\mu\left(\vartheta_{k_\mu} / \vartheta_{k_0}, \vartheta_{k_1}, \ldots, \vartheta_{k_{\mu-1}}\right) = P\left(\theta_\mu = \vartheta_{k_\mu} / \theta_0 = \vartheta_{k_0}, \theta_1 = \vartheta_{k_1}, \ldots, \theta_{\mu-1} = \vartheta_{k_{\mu-1}}\right), 1 \le \mu \le n, \tag{5.3}$$

то поведение марковской цепи описывается соотношением:

$$P(\theta_0, \theta_1, \theta_2, \ldots \theta_n) = P(\theta_0) \cdot \prod_{\mu=1}^{n} \pi_\mu(\theta_\mu / \theta_0, \theta_1, \theta_2, \ldots \theta_{n-1}), \tag{5.4}$$

где $P(\theta_0)$ – вероятность нахождения системы в начальном состоянии θ_0 в момент времени t_0.

Для простой цепи Маркова вероятность фазового состояния в момент времени t_n зависит лишь от того, в каком состоянии система находилась в непосредственно предшествующий момент времени t_{n-1}, поэтому марковские цепи, как и все марковские процессы, для которых выполняется это свойство, называют процессами без последействия. Для таких процессов

$$\begin{cases} \pi_n\left(\theta_n / \theta_0, \theta_1, \ldots, \theta_{n-1}\right) = \pi_n\left(\theta_n / \theta_{n-1}\right); \\ P(\theta_0, \theta_1, \theta_2, \ldots \theta_n) = P(\theta_0) \cdot \prod_{\mu=1}^{n} \pi_\mu(\theta_\mu / \theta_{\mu-1}). \end{cases} \tag{5.5}$$

Вырожденным является случай, когда вероятность нахождения системы в фазовом состоянии θ_n в момент времени t_n не зависит от того, в каком состоянии она находилась в предшествующий момент времени t_{n-1}, при этом:

$$\begin{cases} \pi_n\left(\theta_n / \theta_0, \theta_1, ..., \theta_{n-1}\right) = P_n\left(\theta_n\right); \\ P(\theta_0, \theta_1, \theta_2, ...\theta_n) = \prod_{\mu=0}^{n} P_\mu(\theta_\mu). \end{cases} \tag{5.6}$$

Если обозначить безусловную вероятность того, что на n-м шаге система перейдет в k-е состояние как $P\left(\theta_n = \vartheta_k\right) = p_k\left(n\right)$ и вероятность перехода из j-го состояния на μ-м шаге в k-е состояние на n-м шаге как $P\left(\theta_n = \vartheta_k / \theta_\mu = \vartheta_j\right) = \pi_{jk}\left(\mu, n\right)$ при $\left(j,k\right) = \overline{1,K}, 0 \le \mu \le n, n = \overline{1,N}$, то для определения вероятностей перехода марковской цепи из состояния j в состояние k имеет место уравнение Маркова:

$$\pi_{jk}(\mu, n) = \sum_{i=1}^{K} \pi_{ji}(\mu, m) \cdot \pi_{ik}(m, n), j, k = \overline{1,K}, 0 \le \mu < m < n, \tag{5.7}$$

а для расчета безусловной вероятности нахождения системы в состоянии $\theta_n = \vartheta_k$ следующее соотношение:

$$p_k(n) = \sum_{i=1}^{K} p_i(\mu) \cdot \pi_{ik}(\mu, n); \; i, k = \overline{1,K}; \; 0 \le \mu < m < n. \tag{5.8}$$

Если ввести квадратные матрицы $\pi\left(\mu, n\right) = \pi_{jk}\left(\mu, n\right)$, элементами которой являются вероятности перехода и матрицу-строку $\boldsymbol{P^T}(n) = \left\{p_k(n)\right\}$, то в матричной форме уравнения (5.7) и (5.8) записываются следующим образом[57]:

$$\boldsymbol{\pi}\left(\mu, n\right) = \boldsymbol{\pi}\left(\mu, m\right) \cdot \boldsymbol{\pi}\left(m, n\right), \tag{5.9}$$

$$\boldsymbol{P^T}(n) = \boldsymbol{P^T}(\mu) \cdot \boldsymbol{\pi}(\mu, n) \;\; \textit{или} \;\; \boldsymbol{P}(n) = \boldsymbol{\pi^T}(\mu, n) \cdot \boldsymbol{P}(\mu). \tag{5.10}$$

Различают однородные и неоднородные цепи Маркова. Однородная цепь Маркова характеризуется тем, что вероятности перехода $\pi_{ik}(\mu, n)$ зависят только от разности аргументов, то есть:

$$\pi_{ik}(\mu, n) = \pi_{ik}(n - \mu), \tag{5.11}$$

при этом уравнение Маркова имеет вид:

$$\pi_{jk}(n - \mu) = \sum_{i=1}^{K} \pi_{ji}(m - \mu) \cdot \pi_{ik}(n - m), j, k = \overline{1,K}, 0 \le \mu < m < n. \tag{5.12}$$

Если условие (5.11) не выполняется, то цепь Маркова является неоднородной.

Если для однородной цепи обозначить одношаговые вероятности перехода через $\pi_{jk} \equiv \pi_{jk}(1)$, то поведение цепи описывается системой алгебраических уравнений, решением которой является вероятность того, что моделируемая система попадет в состояние k:

$$p_k = \sum_{i=1}^{K} p_o \cdot \pi_{ik}; \; k = \overline{1,K}. \tag{5.13}$$

Если обозначить матрицу вероятностей единичных переходов

$$\boldsymbol{\pi} = \begin{bmatrix} \pi_{11}\,\pi_{12}\ldots\pi_{1K} \\ \pi_{21}\,\pi_{22}\ldots\pi_{2K} \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ \pi_{K1}\,\pi_{K2}\ldots\pi_{KK} \end{bmatrix}, \tag{5.14}$$

то уравнение (5.12) можно записать в матричной форме:

$$\boldsymbol{P^T}(m+\mu) = \boldsymbol{P^T}(\mu)\cdot\boldsymbol{\pi}^m. \tag{5.15}$$

Для цепи Маркова с конечным числом состояний при выполнении условия $\pi_{jk}(n) > 0$, начиная с некоторого числа n, существуют предельные (финальные) вероятности:

$$p_k = \lim_{m\to\infty} p_k(m); k = \overline{1,K}, \tag{5.16}$$

которые не зависят от начального распределения $\left\{p_k^{(0)}\right\}$.

Такие марковские цепи называют эргодическими[20], при этом финальные вероятности являются решением системы линейных уравнений (5.14), удовлетворяющих дополнительному требованию:

$$\sum_{k=1}^{K} p_k = 1,\ p_k \geq 0. \tag{5.17}$$

Финальные вероятности образуют стационарное распределение $\{p_k\}$.

Цепь Маркова называют стационарной, начиная с t_0, если вероятности начального состояния $\left\{p_k^{(0)}\right\}$ совпадают с соответствующими финальными вероятностями $\{p_k\}$.

Цепь Маркова для наглядности часто представляют в виде ориентированного графа, вершины которого определяются состояниями цепи, а каждой дуге, идущей из состояния ϑ_j в состояние ϑ_k, ставится в соответствие число π_{jk}, если $\pi_{jk} > 0$. Пример такого графа приведен на рисунке 5.1.

Рассмотрим пример использования аппарата марковских цепей для моделирования процессов реализации угроз безопасности информации в ИС.

Пример 5.1: модель процесса инфицирования компьютера вредоносной программой, заражающей исполняемые файлы.

---

[20]Стационарный случайный процесс называется *эргодическим,* если любая его вероятностная характеристика, полученная усреднением по множеству возможных реализаций, с вероятностью, сколь угодно близкой к единице, равна временному среднему, полученному усреднением за достаточно большой промежуток времени одной единственной реализации случайного процесса, то есть выполняется равенство:

$m_x = M\left[x(t)\right] = \int_{-\infty}^{\infty} x\cdot p(x)dx = \lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T} x(t)dt$, где $x(t)$ – случайный процесс, $m_x = M\left[x(t)\right]$ – его математическое ожидание, а $\left[-T,T\right]$ – временной отрезок, на котором осуществляется усреднение по одной из реализация случайного процесса.

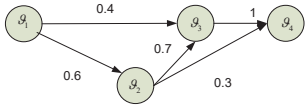


Рисунок 5.1 – Пример ориентированного графа
для однородной марковской цепи

Пусть вредоносная программа внедрена в компьютер (записана в одном из каталогов файловой системы), которая должна осуществить поиск исполняемых файлов, например, с расширением *.exe, а затем инфицировать их. Программа осуществляет последовательно поиск каждого такого файла и инфицирует его, после чего переходит к поиску и инфицированию другого файла и т.д. Пусть вероятность внедрения вредоносной программы составляет величину p_0, вероятность поиска – π_{find} и вероятность инфицирования – π_{inf}. Ориентированный граф такой марковской цепи приведен на рисунке 5.2, при этом для упрощения графа процедуры поиска и инфицирования файлов объединены в единую процедуру.

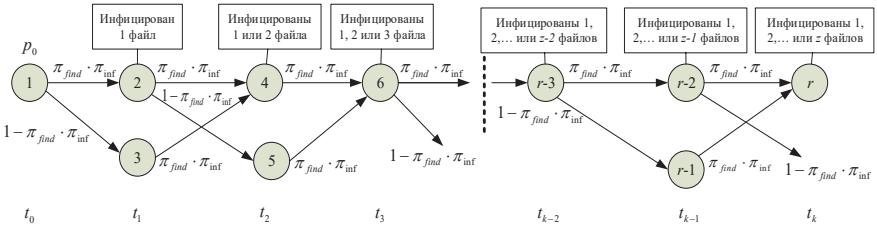


Рисунок 5.2 – Граф марковской цепи, моделирующей инфицирование компьютера вредоносной программой

Рассмотрим простой случай, когда число состояний равно 4 ($r = 4$).

Тогда переходные вероятности определяются следующим образом $\pi_{11} = 0, \pi_{12} = \pi_{find} \cdot \pi_{inf}, \pi_{13} = 1 - \pi_{find} \cdot \pi_{inf}, \pi_{24} = \pi_{34} = \pi_{find} \cdot \pi_{inf}$, а вероятности того, что в системе через два шага будет инфицирован только один файл, то есть система перейдет или в состояние 5, или по маршруту $1 \to 3 \to 4$ в состояние 4, определяется из соотношения:

$$p_2 = p_0(\pi_{find} \cdot \pi_{\inf})(1 - \pi_{find} \cdot \pi_{\inf}) \quad . \tag{5.18}$$

Тогда вероятность того, что за k шагов вредоносная программа сумеет инфицировать ровно $z, z \le \dfrac{r+1}{2}$, исполняемых файлов, определяется из соотношения:

$$p_k = p_0 C_k^z (\pi_{find} \cdot \pi_{\inf})^z (1 - \pi_{find} \cdot \pi_{\inf})^{k-z} \quad , \tag{5.19}$$

то есть распределение вероятности представляет собой распределение Бернулли, при этом среднее количество инфицированных файлов за k шагов составит величину $k \cdot \pi_{find} \cdot \pi_{\inf}$.

Несмотря на глубокую проработанность, аппарат марковских цепей нашел лишь весьма ограниченное применение при моделировании процессов в предметной области ТЗИ (если не учитывать их использование в виде процессов Д.Кендалла в аппарате полумарковских процессов, см. раздел 5.7). В основном он применяется в случаях, когда оказывается достаточным представление процесса в виде мгновенных вероятностных переходов из состояния в состояние или с жестко фиксированной длительностью каждого перехода, одинаковой для всех переходов, при этом состав возможных состояний определяется заранее. Значительно более широкое применение нашел аппарат дискретных марковских процессов с непрерывным временем.

5.2. Введение в теорию дискретных марковских процессов с непрерывным временем

Пусть моделируемый процесс $\theta(t)$ может принимать значения $\vartheta_k, k = \overline{1,K}$, причем смена состояний происходит в случайные моменты времени (рисунок 5.3).

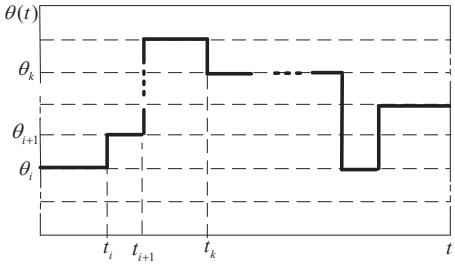


Рисунок 5.3 – Дискретный марковский процесс с непрерывным временем

Если ввести вероятности перехода процесса из состояние i в состояние j как условные вероятности того, что в момент времени t процесс будет в состоянии ϑ_j, если в момент времени t_0 он был в состоянии ϑ_i, то есть:

$$\pi_{ij}(t_0,t) = P\left\{\theta(t) = \vartheta_j \mid \theta(t_0) = \vartheta_i\right\}, \tag{5.20}$$

то, во-первых, справедливы соотношения

$$\sum_{j=1}^{K} \pi_{ij}(t_0,t) = 1,\ \pi_{ij}(t_0,t) \geq 0, i, j = \overline{1,K}, \tag{5.21}$$

$$\pi_{ij}\left(t_0,t\right) = \delta_{ij} = \begin{cases} 1 \text{ при } i = j; \\ 0 \text{ при } i \neq j, \end{cases} \tag{5.22}$$

и, во-вторых, имеет место следующее уравнение Колмогорова-Чепмена:

$$\pi_{ij}\left(t_0, t+\Delta t\right)=\sum_{k=1}^{K}\pi_{ik}\left(t_0, t\right)\cdot\pi_{kj}\left(t, t+\Delta t\right),\ t>t_0,\ \Delta t>0\,. \tag{5.23}$$

Для малых временных интервалов вероятности переходов в первом приближении имеют вид:

$$\begin{aligned}&\pi_{kk}(t, t+\Delta t)\approx 1+\lambda_{kk}(t)\cdot\Delta t;\\&\pi_{kj}(t, t+\Delta t)\approx\lambda_{kj}(t)\cdot\Delta t,\end{aligned} \tag{5.24}$$

то есть прямо пропорциональны размеру этого временного интервала, где параметр $\lambda_{kj}\left(t\right)$ имеет смысл интенсивности переходов.

Так как выполняется условие (5.24), то из (5.23) следует, что

$$\lambda_{kk}(t)=-\sum_{j(j\neq k)}\lambda_{kj}(t),\ \lambda_{kj}(t)\geq 0\,, \tag{5.25}$$

и уравнение Колмогорова-Чепмена путем предельного перехода при $\Delta t\to 0$ преобразуется к виду:

$$\frac{\partial}{\partial t}\pi_{ij}\left(t_0, t\right)=\sum_{k=1}^{K}\lambda_{kj}\cdot\pi_{ik}\left(t_0, t\right). \tag{5.26}$$

Аналогичному уравнению удовлетворяют не только вероятности переходов, но и абсолютные вероятности состояний:

$$\frac{d}{dt}p_j\left(t\right)=\sum_{k=1}^{K}\lambda_{kj}\cdot p_k\left(t\right). \tag{5.27}$$

Дискретный марковский процесс с непрерывным временем считается однородным, если вероятности перехода

$$\pi_{i,j}(t_0, t)=\pi_{i,j}(\tau),\ \tau=t-t_0\,, \tag{5.28}$$

зависят только от разности времен между началом и завершением переходов из состояния i в состояние j, а не от абсолютных значений времени.

Для однородных марковских процессов интенсивность $\lambda_{kj}\left(t\right)=\lambda_{kj}$ не зависит от времени и уравнение Колмогорова-Чепмена (5.26) преобразуется к виду:

$$\frac{d}{d\tau}\pi_{i,j}(\tau)=\sum_{k=1}^{K}\lambda_{kj}\cdot\pi_{ik}(\tau)\,. \tag{5.29}$$

Если для однородного марковского процесса существует предельное значение вероятностей переходов

$$p=\lim_{\tau\to\infty}\pi_{ij}(\tau)\,, \tag{5.30}$$

которые не зависят от начального состояния, то такие процессы относят к эргодическим, для которых существует однозначно определенное стационарное состояние, при этом вероятности стационарных состояний находятся путем решения системы алгебраических уравнений:

$$\sum_{k=1}^{K} \lambda_{kj} \cdot p_k = 0, \ \sum_{k=1}^{K} p_k = 1 . \qquad (5.31)$$

При моделировании того или иного процесса с использованием рассматриваемого аппарата этот процесс представляют в виде ориентированного графа, вершины которого соответствуют дискретным состояниям процесса, и дуги указывают на пути перехода процесса между состояниями во времени. Пример такого графа приведен на рисунке 5.4.

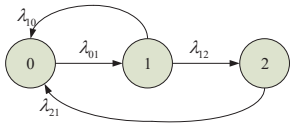


Рисунок 5.4 – Пример ориентированного графа для марковского процесса с дискретным множеством состояний и непрерывным временем

После построения такого графа достаточно просто написать систему уравнений Колмогорова-Чепмена для вероятностей нахождения моделируемой системы в каждом из состояний, соблюдая следующие мнемонические правила:

1) в левой части каждого уравнения указывается производная вероятности нахождения системы в данном состоянии;

2) в правой части каждого уравнения указываются произведения вероятностей нахождения системы в состояниях, смежных с данным состоянием, переход в которые или из которых в данное состояние указан на графе дугами с ненулевыми интенсивностями, при этом интенсивность берется со знаком «минус», если дуга является исходящей из рассматриваемого состояния и со знаком «плюс», если дуга является входящей в рассматриваемое состояние.

Так, для указанного на рисунке 5.4 графа система дифференциальных уравнений, описывающих динамику изменения состояний рассматриваемой системы, с начальными условиями имеет вид:

$$\begin{cases} p_0^{'}(t) = -\lambda_{01} \cdot p_0(t) + \lambda_{10} \cdot p_1(t) + \lambda_{21} \cdot p_2(t); \\ p_1^{'}(t) = \lambda_{01} \cdot p_0(t) - \lambda_{10} \cdot p_1(t) - \lambda_{12} \cdot p_1(t); \\ p_2^{'}(t) = \qquad\qquad \lambda_{12} \cdot p_1(t) - \lambda_{21} \cdot p_2(t), \end{cases} \qquad (5.32)$$

при начальных условиях $p_0(0) = 1, p_1(0) = p_2(0) = 0, \ p_0 + p_1 + p_2 = 1$.

Наиболее широко на практике для решения таких систем дифференциальных уравнений применяется метод преобразования Лапласа, что позволяет свести всё к решению системы алгебраических уравнений с последующим обратным его преобразованием Лапласа. Например, для системы уравнений (5.32) его преобразование Лапласа записывается следующим образом:

$$
\begin{aligned}
s \cdot x_0(s) &= -\lambda_{01} \cdot x_0(s) + \lambda_{10} \cdot x_1(s) + \lambda_{21} \cdot x_2(s); \\
s \cdot x_1(s) &= \lambda_{01} \cdot x_0(s) - \lambda_{10} \cdot x_1(s) - \lambda_{12} \cdot x_1(s); \\
s \cdot x_2(s) &= \lambda_{12} \cdot x_1(s) - \lambda_{21} \cdot x_2(s),
\end{aligned} \tag{5.33}
$$

где $x_i(s) = \int_0^\infty e^{-s \cdot t} \cdot f_i(t) \cdot dt,\ i = \overline{0,2}$ преобразование Лапласа от вероятности или ее производной для i-го состояния.

Решение данной системы уравнений относительно вероятности перехода системы в конечное состояние 2 имеет вид

$$
P_2(t) = \frac{\lambda_{12} \cdot \lambda_{01}}{\lambda_{21} \cdot \left(\lambda_{01} + \lambda_{10} + \lambda_{12}\right) + \lambda_{01} \cdot \lambda_{12}} + \frac{e^{-A_1 \cdot t}}{A_1 \cdot \left(2 \cdot A_1 + \lambda_{01} + \lambda_{10} + \lambda_{12} + \lambda_{21}\right)} + \tag{5.34}
$$

$$
+ \frac{e^{-A_2 \cdot t}}{A_2 \cdot \left(2 \cdot A_2 + \lambda_{01} + \lambda_{10} + \lambda_{12} + \lambda_{21}\right)},
$$

где $A_{1,2} = -\frac{\lambda_{01} + \lambda_{10} + \lambda_{12} + \lambda_{21}}{2} \pm \sqrt{\frac{(\lambda_{01} + \lambda_{10} + \lambda_{12} + \lambda_{21})^2}{4} - \lambda_{21} \cdot \lambda_{10}}$.

Из приведенного видно, что уже при трех состояниях решение системы уравнений имеет весьма громоздкий вид, а при количестве состояние более 4 в общем случае не имеет аналитического решения и, как правило, решается численными методами. Это приводит к существенным ограничениям в применении теории марковских процессов в практике моделирования. Кроме того, имеются и другие существенные недостатки, к которым относятся следующие:

1) для определения интенсивностей переходов, зачастую, необходимо дополнительно разрабатывать соответствующие модели;

2) в связи с необходимостью сокращения количества учитываемых состояний процесса (чаще всего, за счет их агрегирования) возникают трудности в интерпретации состояний процесса;

3) невозможно моделировать параллельные взаимосвязанные процессы;

4) невозможно вводить в систему дифференциальных уравнений логические условия, которые часто обусловливают протекание моделируемого процесса.

Все это привело к применению и других подходов к моделированию динамики реализации процессов, рассмотренных далее в разделе 6.

5.3. Элементы теории случайных потоков, существенные для моделирования процессов в предметной области защиты информации

Теория случайных потоков событий начала развиваться в рамках теории марковских процессов и формировавшейся затем на ее основе теории массового обслуживания, при этом в первую очередь рассматривался случайный поток заявок на обслуживания [57, 83–88]. Позже эта теория

приобрела и некоторое самостоятельное значение, поскольку стала применяться значительно более широко в самых различных областях (например, при проектировании телекоммуникационных систем, при анализе характеристик случайного трафика, в моделях размножающихся популяций и др.). Элементы этой теории находят применении и при моделировании различных процессов в предметной области защиты информации. Понятие случайного потока событий вводится следующим образом [89].

Пусть на отрезке (τ_0, τ) действительной оси, где $\tau \geq \tau_0$, появляется некоторое случайное число точек k в случайных местах оси, при этом $\tau_0 \leq \tau_1 \leq \tau_2 \leq \ldots \leq \tau_k \leq \ldots \leq \tau$ и, если не считать различия в координатах, появившиеся точки являются неразличимыми, что соответствует однородным событиям. При этом, если рассматривается временная ось, то момент времени τ_i называют моментом наступления i-го события. Тогда случайным потоком называют случайный процесс $N(\tau_0, \tau)$, реализация которого характеризует число точек, появившихся на интервале (τ_0, τ), как функцию правого конца этого интервала. Согласно этому определению реализации случайного потока являются неубывающими ступенчатыми целочисленными функциями, область определения которых Ω лежит выше τ_0 и в общем случае может быть бесконечной справа. Если обозначить вероятность появления в области Ω ровно n точек как $p_n(\Omega)$, то для нее справедливо условие:

$$\sum_{n=0}^{\infty} p_n(\Omega) = 1 . \tag{5.35}$$

Часто весьма удобным является описание числа появившихся точек с помощью производящей функции:

$$\zeta(z;\Omega) = \sum_{n=0}^{\infty} p_n(\Omega) \cdot z^n , \tag{5.36}$$

при этом

$$p_n(\Omega) = \frac{1}{m!} \cdot \left. \frac{d^n \zeta(z,\Omega)}{dz^n} \right|_{z=0} , n = 1,2\ldots . \tag{5.37}$$

Чаще всего при моделировании различных процессов областью их определения является некоторый промежуток времени (возможно бесконечный). Если рассматривать поток однородных событий, возникающих в моменты времени $\tau_0 \leq \tau_1 \leq \tau_2 \leq \ldots \leq \tau_k \leq \ldots \leq \tau$, где положить $\tau_0 = 0$ и обозначить через $d_k = \tau_k - \tau_{k-1}$ расстояние между соседними моментами времени, то такой поток считается заданным, если задано распределение случайного вектора $(d_1, d_2, \ldots, d_k)$.

В работах А.Я. Хинчина [86, 88] показано, что если однородный случайный поток обладает свойствами:

– стационарности, то есть для любой группы непересекающихся отрезков времени вероятность появления в них соответственно $n_1, n_2, ..., n_k$ событий (точек) зависит только от этих чисел и от длин указанных отрезков времени, но не зависит от их расположения на оси времени, при этом для промежутка времени (T, t) вероятность появления n событий не зависит от T и является функцией только переменных t и n;

– отсутствия последействия, то есть вероятность появления n событий в течение времени (T, t) не зависит от того, сколько событий произошло до момента времени T, а это означает взаимную независимость появления того или иного числа событий в непересекающихся отрезках времени, то есть независимость всех случайных величин d_k.

Для задания такого потока достаточно задать набор функций $F_k(t) = p(d_k \le t), k \ge 1$, где $F_k(t)$ – функция распределения k-го интервала времени. Поток с ограниченным последействием, для которого $F_2(t) = F_3(t) = ... = F(t)$ называют рекуррентным потоком с запаздыванием [87];

– ординарности потока событий – условия практической невозможности появления двух или больше событий в один и тот же момент времени, которое формально записывается в виде $\lim\limits_{\Delta t \to 0} \frac{p_{n>1}(\Delta t)}{\Delta t} = 0$, то для такого потока функция распределения вероятности для всех промежутков времени между событиями определяется по формуле $F(t) = 1 - e^{-\lambda \cdot t}$, а вероятность возникновения за время t ровно n событий определяется по формуле Пуассона:

$$p_n(t) = \frac{(\lambda \cdot t)^n}{n!} \cdot e^{-\lambda \cdot t}, \tag{5.38}$$

где λ – параметр, получивший название интенсивность потока.

Такой поток называют простейшим или стационарным пуассоновским. Если обозначить через $p_{\ge 1}(t)$ вероятность появления хотя бы одного события за время t, то для ее расчета справедлива формула

$$p_{\ge 1}(t) = \sum_{n=1}^{\infty} p_n(t) = 1 - p_0(t), \tag{5.39}$$

а для интенсивности потока – следующее предельное равенство:

$$\lim_{t \to 0} \frac{p_{\ge 1}(t)}{t} = \lambda, \tag{5.40}$$

при этом среднее количество событий, произошедших на интервале времени t, будет равно $\overline{n}(t) = \lambda \cdot t$.

Если бы параметр λ зависел от времени, то тогда процесс был бы неоднородным и для него справедлива была бы вместо соотношения (5.38) следующая формула:

$$p_n(t_0,t)=\frac{1}{n!}\cdot(\int_{t_0}^{t}\lambda(x)dx)^n\exp(-\int_{t_0}^{t}\lambda(x)dx)\,. \qquad (5.41)$$

Далее будут рассматриваться только нашедшие широкое применение в практике моделирования однородные процессы.

Следует отметить, что при использовании теории случайных потоков часто приходится учитывать не только случайные времена возникновения событий, но и другие условия их возникновения. Например, к таким условиям можно отнести:

включения или выключения компьютеров сети, при этом в периоды времени, когда компьютеры выключены, реализация угроз безопасности информации в них невозможна;

эпизодическое использование некоторых прикладных программ, с которыми ассоциируются ряд вредоносных программ, при этом без запуска таких приложений вредоносные программы не могут быть активированы и др.

Для учета динамики возникновения таких условий можно при моделировании использовать квазирекуррентные потоки, в которых для моделирования таких ситуаций вводится понятие «вызывающего промежутка времени» [87], вне которого вообще невозможно возникновение угрозы, то есть динамика возникновения угроз определяется временными характеристиками потока «вызывающих промежутков времени» и собственно попытками атак. В этом случае поток угроз может быть представлен в виде потока пачек импульсов, в котором поток огибающих равнозначен рекуррентному потоку «вызывающих промежутков времени» (поток «вызовов»), а поток внутри этих промежутков соответствует собственно рекуррентному потоку самих угроз (рисунок 5.5).

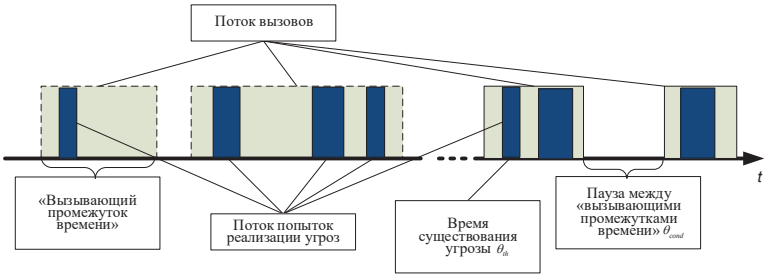


Рисунок 5.5 – Квазирекуррентный поток попыток угроз реализации сетевых атак

Такой поток импульсов получил название квазирекуррентного [87]. Для него время до появления первой попытки реализации угрозы представляет собой сумму случайных независимых величин θ_{cond} и θ_{th}, то есть времени до появления условий существования угрозы и времени возникновения

собственно самой попытки: $\tau = \theta_{th} + \theta_{cond}$. Тогда вероятность возникновения угрозы определяется из соотношения:

$$P_{beg}(t) = 1 + \frac{1}{\overline{\mu}_{cond} - \overline{\mu}_{th}}[\overline{\mu}_{th}\exp(-\overline{\mu}_{cond}t) - \overline{\mu}_{cond}\exp(-\overline{\mu}_{th}t)], \quad (5.42)$$

где $\overline{\mu}_{cond}$ и $\overline{\mu}_{th}$ – средние интенсивности возникновения условия для реализации угроз и попыток такой реализации соответственно,

$$\overline{\mu}_{cond} = \frac{1}{\overline{\tau}_{cond} + \overline{\theta}_{cond}} \; ; \; \overline{\mu}_{th} = \frac{1}{\overline{\tau}_{th} + \overline{\theta}_{th}}. \quad (5.43)$$

В определенной мере альтернативой использования теории рекуррентных потоков является теория прореживания потоков, поскольку выполнение подобных условий приводит к тому, что не все события в исходном потоке выполняются, при этом происходит прореживание исходного потока. Использование операции прореживания позволяет значительно расширить моделирующие возможности теории потоков.

Для определения прореженного потока в соответствии с [87], кроме задания исходного потока, необходимо каждому моменту времени t_k поставить в соответствие функцию $\delta_k = \delta(t_k)$, принимающую значение 0 для потерянных событий и 1 – для состоявшихся событий, а также задать вероятности:

$$P(\delta_1, \delta_2, ..., \delta_k) = P\{\delta(t_1) = \delta_1, \delta(t_2) = \delta_2, ..., \delta(t_k) = \delta_k\}. \quad (5.44)$$

Операция прореживания потока относится к рекуррентной, если она задана следующими вероятностями [87]:

$$\begin{cases} P(1) = a_0; \\ P(0,1) = a_1; \\ P(0,0,1) = a_2; \\ \dots\dots\dots\dots\dots\dots \\ P(\underbrace{0,0,...,0}_{k},1) = a_k; \\ \dots\dots\dots\dots\dots\dots \\ \sum_{k>0} a_k = 1. \end{cases} \quad (5.45)$$

Если обозначить через $B(t)$ функцию распределения вероятностей для промежутка времени между соседними событиями прореженного потока, то имеет место соотношение [87]:

$$B(t) = a_0 F(t) + a_1 F_2(t) + ... + a_k F_{k+1}(t), \quad (5.46)$$

так как

$$F_1(t) = F(t), F_2(t) = \int_0^t F_1(t-x)dF(x), ..., F_k = \int_0^t F_{k-1}(t-x)dF(x)... \tag{5.47}$$

$$\text{или } F_k(t) = \left[F(t)\right]_*^{(k)},$$

где правая часть – это стильтьесовская свертка k-го порядка [92], при этом характеристическая функция для свертки функций распределения равна произведению характеристических функций для функций распределения, входящих в свертку, то есть:

$$\chi_k(s) = \prod_{m=1}^{k} \chi_m(s), m = \overline{1,k}, \tag{5.48}$$

а функция распределения длительности паузы между соседними событиями в прореженном потоке определяется следующим образом:

$$B(t) = \sum_{k\geq 0} a_k \left[F(t)\right]_*^{k+1} = \int_0^{\lambda t} \left(\sum_{k\geq 0} a_k \frac{x^k}{k!} e^{-x} \right) \cdot dx. \tag{5.49}$$

Средняя интенсивность прореженного потока определяется из соотношения:

$$\lambda_b = \lambda \Big/ \left(1 + \sum_{k\geq 1} k \cdot a_k \right). \tag{5.50}$$

Рассмотрим два примера.

Пример 5.2. Пусть исходный поток событий является пуассоновским, любое событие которого теряется с вероятностью $p = 1 - q$ и остается с вероятностью p. Тогда для прореженного потока

$$B(t) = 1 - \exp\left(-p \cdot \lambda \cdot t\right), \tag{5.51}$$

а его средняя интенсивность равна $\lambda_b = p \cdot \lambda$ и дисперсия $\sigma_b = \lambda_b$ [92].

Пример 5.3. Исходный пуассоновский поток с интенсивностью λ просеивается следующим образом: первые k событий теряются, а $(k+1)$-й остается, затем снова следующие k событий теряются, а следующий остается и т.д. В этом случае в соответствии с (5.47)

$$B(t) = \int_0^{\lambda \cdot t} \frac{x^k}{k!} \cdot e^{-x} \cdot dx. \tag{5.52}$$

Рекуррентный поток, определяемый такой функцией распределения, называется потоком Эрланга [86]. Для этого потока математическое ожидание длительности паузы между соседними успешными событиями и среднеквадратическое ее отклонение определяются из соотношений:

$$M_\theta(k) = \frac{k+1}{\lambda}; \ \sigma_\theta(k) = \frac{\sqrt{k+1}}{\lambda}. \tag{5.53}$$

При больших значениях k распределение паузы между событиями в потоке становится близким к нормальному, а при $k \to \infty$ результирующий поток становится детерминированном периодическим потоком (с периодом также стремящимся к бесконечности).

Применительно к моделированию динамики реализации угроз безопасности информации, особенно связанных с проникновением в операционную среду атакуемого компьютера, необходимо отметить, что процесс реализации, по сути, представляет собой процесс вероятностного прореживания исходного потока угроз. Действительно, если исходный поток угроз проникновения имеет среднюю интенсивность возникновения угроз $\lambda_{beg} = \frac{1}{\tau_{beg}}$, то результирующий поток реализаций угрозы в среднем будет иметь интенсивность:

$$\overline{\lambda_u} = \lambda_{beg} \cdot p_{pr} \cdot p_g, \tag{5.54}$$

где p_{pr}, p_g – вероятности прореживания исходного потока на этапе проникновения нарушителя в операционную среду компьютера (вероятность проникновения) и на этапе выполнения деструктивного действия (вероятность выполнения деструктивного действия),

$$p_{pr} = \frac{1}{\lambda_{beg} \cdot \left| \frac{d}{ds} \chi\left(s, \tau_{pr}\right) \right|_{s=0}}; \; p_g = \frac{1}{\lambda_{pr} \cdot \left| \frac{d}{ds} \chi\left(s, \tau_g\right) \right|_{s=0}}; \tag{5.55}$$

$\chi(s, \tau_{pr})$, $\chi(s, \tau_g)$ – характеристические функции случайных времен проникновения в операционную среду τ_{pr} и выполнения несанкционированного действия τ_g.

Пусть исходный поток угроз является произвольным рекуррентным потоком, при этом каждая попытка реализации завершается успешно лишь с вероятностью:

$$p_{реал} = p_{pr} \cdot p_g. \tag{5.56}$$

Тогда с учетом [93] функцию распределения вероятности для времени между успешными попытками реализации угрозы можно записать следующим образом:

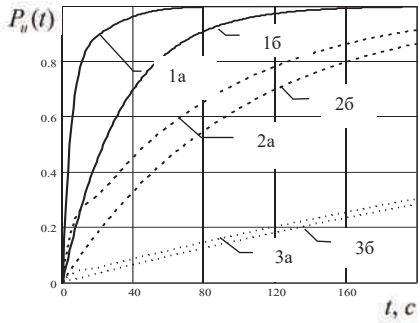
$$P_u(t) = p_{real} \cdot \lambda_{beg} \cdot \int_0^t e^{-\lambda_{beg} \cdot x} \cdot \sum_{m=1}^{\infty} \frac{(\lambda_{beg} \cdot x)^{k(m-1)}}{[k \cdot (m-1)!]} (1 - p_{real})^{m-1} \cdot dx, \tag{5.57}$$

где k – параметр исходного потока.

Нетрудно показать, что в этом случае:

$$P_u(t) = 1 - \sum_{m=1}^{\infty} \sum_{j=0}^{k \cdot (m-1)} \frac{(\lambda_{beg} \cdot t)^j \cdot p_{real} (1 - p_{real})^{m-1} \cdot e^{-\lambda_{beg} \cdot t}}{j!}. \tag{5.58}$$

С учетом изложенного на рисунке 5.6 приведена зависимость (5.57).



а – для потока с произвольным исходным распределением, аппроксимируемым потоком Эрланга с параметром $k = 6$;

б – для пуассоновского потока;

1 – для $\lambda_{beg} = 0.2, p_{real} = 0.9$; 2 – для $\lambda_{beg} = 0.2, p_{real} = 0.3$; 3 – для $\lambda_{beg} = 0.2, p_{real} = 0.05$

Рисунок 5.6 – Зависимость вероятности реализации угроз от времени при различных законах распределения времени между попытками реализации угроз в исходном потоке

На этом же рисунке показаны зависимости для пуассоновского исходного потока. По графику видно, что с уменьшением вероятности $p_{реал}$ поток приближается к пуассоновскому.

Эта тенденция может быть доказана более строго. В [90, 93] показано, что для получаемого путем рекуррентного прореживания с вероятностью p потока характеристическая функция $\chi_*(s)$ для интервала времени между соседними событиями в прореженном потоке имеет вид:

$$\chi_*(s) = 1 - \frac{p \cdot \chi(s)}{1 - (1-p) \cdot \chi(s)}, \tag{5.59}$$

где $\chi(s)$ – характеристическая функция для интервала времени между соседними событиями в исходном потоке.

В [94] предложено в качестве меры близости потока к пуассоновскому использовать коэффициент вариации, определяемый как отношение среднеквадратического отклонения промежутка времени между событиями в потоке к его математическому ожиданию:

$$K_{var} = \frac{\sigma_\theta}{M_\theta} = \left. \frac{\sqrt{\chi''(s) - \left(\chi'(s)\right)^2}}{\chi'(s)} \right|_{s=0}. \tag{5.60}$$

При этом крайними случаями являются детерминированный периодический поток, для которого $K_{var} = 0$, и пуассоновский поток, для которого $K_{var} = 1$.

Нетрудно показать, что коэффициент вариации для прореженного с вероятностью p потока связывается с коэффициентом вариации исходного потока соотношением:

$$K_{var} = \left.\frac{\sqrt{\chi''(s) - \left(\chi'(s)\right)^2}}{\chi'(s)}\right|_{s=0} = \sqrt{1 + p \cdot \left(K_{var}^2 - 1\right)}, \qquad (5.61)$$

представленным в виде графической зависимости на рисунке 5.7.

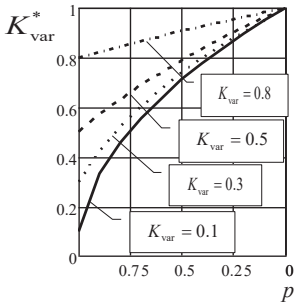


Рисунок 5.7 – Зависимость коэффициента вариации прореженного потока от вероятности прореживания и коэффициента вариации исходного потока

Это важное обстоятельство обусловливает возможность при моделировании процессов, в которых имеет место то или иное вероятностное прореживание исходного потока, заменять исходный поток на пуассоновский без заметных отклонений в оценках характеристик прореженных потоков.

Существенным для моделирования процессов в предметной области защиты информации является *использование элементов теории совпадения случайных импульсных потоков* [89].

В приложениях наиболее часто приходится иметь дело с потоком взаимно неперекрывающихся во времени импульсов, когда пауза между моментами появления соседних импульсов $T_i = t_i - t_{i-1} > \tau_{i-1}$, где t_i и t_{i-1} – моменты появления i-го и $i-1$ импульсов, а τ_{i-1} и τ_i – длительности $(i-1)$-го и i-го импульсов (рисунок 5.8).

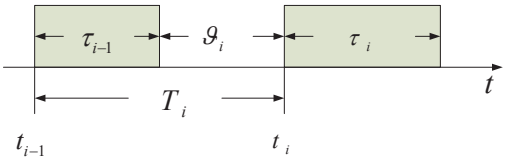


Рисунок 5.8 – Реализация потока неперекрывающихся потоков прямоугольных импульсов

В случае стационарного потока взаимно неперекрывающихся импульсов произвольно взятый момент времени в пределах основания импульса при $t \to \infty$ окажется с вероятностью $p = \overline{\mu} \cdot \overline{\tau}$, где $\overline{\mu}$ – средняя частота следования импульсов, а $\overline{\tau}$ – средняя их длительность.

В различных приложениях нередко возникает задача, связанная с определением средней частоты совпадения заданного числа N потоков. Решение такой задачи было приведено в [89], при этом полагалось, что для всех совпадающих потоков заданы плотности распределения вероятностей для длительностей импульсов $\alpha(\tau)$ и пауз между ними $\beta_n(\tau), n = \overline{1,N}$. В этом случае средняя частота $\overline{\mu_{N,k}(\delta)}$ совпадения k из N импульсов при условии, что продолжительность их перекрытия будет не менее δ, определяется из соотношения:

$$\overline{\mu_{N,k}(\delta)} = -\frac{1}{k!}\frac{\partial^{k+1}}{\partial s^k \cdot \partial \delta} \cdot \prod_{n=1}^{N} \left[Q_n(\delta) + s \cdot P_n(\delta) \right] \Big|_{s=0}, \tag{5.62}$$

где $P_n(\delta) = \overline{\mu_n} \cdot \int_{\delta}^{\infty} dx \int_{x}^{\infty} \alpha_n(y)\,dy;\ Q_n(\delta) = \overline{\mu_n} \cdot \int_{\delta}^{\infty} dx \int_{x}^{\infty} \beta_n(y)\,dy$.

В частности, если имеется N независимых импульсных потоков, то средняя частота их совпадения (когда в совпадении участвуют импульсы всех N потоков) определяется из следующим образом:

$$\overline{\mu_{N,N}(\delta)} = \sum_{n=1}^{N} \frac{1}{\overline{\tau_n} - \delta} \cdot \prod_{r=1}^{N} \overline{\mu_r} \cdot (\overline{\tau_r} - \delta), \tag{5.63}$$

а при $\delta = 0$ формула (5.63) еще более упрощается

$$\overline{\mu_{N,N}} = \sum_{n=1}^{N} \frac{1}{\overline{\tau_n}} \cdot \prod_{r=1}^{N} \overline{\mu_r} \cdot \overline{\tau_r}\,. \tag{5.64}$$

Важной, как правило, является также задача оценки средней длительности совпадения импульсов N потоков, то есть средней длительности импульса потока совпадений, образованных в результате перекрытия по времени k из N импульсов. В [89] показано, что средняя длительность рассчитывается по формуле:

$$\overline{\tau_{N,k}} = \frac{p_{N,k}}{\overline{\mu_{N,k}}}, \tag{5.65}$$

где $p_{N,k} = \frac{1}{k!}\frac{d^k}{ds^k}\prod_{n=1}^{N}(1 - p_n + s \cdot p_n)$ и $p_n = \overline{\mu_n} \cdot \overline{\tau_n}$. (5.66)

Если для всех n справедливо $\overline{\mu_n} = \overline{\mu}$ и $\overline{\tau_n} = \overline{\tau}$, то имеют место соотношения:

$$\overline{\tau_{N,k}} = \frac{1}{\overline{\mu}} \cdot \frac{p \cdot (1-p)}{(N-k) \cdot p + k \cdot (1-p)}, p = \overline{\mu} \cdot \overline{\tau},\ \ при\ \ n \neq k, \tag{5.67}$$

$$\overline{\mu_{N,k}} = \frac{N!}{(N-k)!\cdot k!}\cdot \overline{\mu}\cdot\left[(N-2k)\cdot p + k\right]\cdot p^{k-1}\cdot(1-p)^{N-(k-1)}. \qquad (5.68)$$

Если $k = N$, то соотношения (5.67) преобразуются к виду:

$$\overline{\tau_{N,N}} = \left(\sum_{n=1}^{N}\frac{1}{\overline{\tau_n}}\right)^{-1}. \qquad (5.69)$$

Пример 5.4. Пусть пользователь подключается к сети общего пользования лишь эпизодически – один раз в сутки в среднем на двадцать минут и достаточно редко пользуется прикладной программой (например, программой Wordpad – раз в месяц в среднем в течение 3 часов), с которой ассоциирована вредоносная программа, инфицирующая компьютер при условии, что запущена прикладная программа и компьютер находится в сети. Спрашивается, через какое время с вероятностью 0.95 компьютер будет инфицирован. Поток, моделирующий работу пользователя в сети определяется в этом случае параметрами средней интенсивности подключения к сети общего пользования, равной $\overline{\mu_{net}} = 1\frac{1}{сутки}$ и средней длительности работы $\overline{\tau_{net}} = \frac{20}{60\cdot 24}$ суток ≈ 0.014 суток в ней, а параметры потока, моделирующего работу с прикладной программой, соответствуют интенсивности $\overline{\mu_{soft}} = \frac{1}{30}\frac{1}{\text{сутки}}$ и длительности $\overline{\tau_{soft}} = 0.125$ суток. Тогда параметры потока совпадения будут равны: интенсивность $\overline{\mu_{22}} = \overline{\mu_{net}}\cdot\overline{\mu_{soft}}\cdot\left(\overline{\tau_{net}} + \overline{\tau_{soft}}\right) \approx 0.0046\frac{1}{\text{сутки}}$, при этом вероятность инфицирования компьютера определяется из соотношения $P_{inf} = 1 - e^{-\overline{\mu_{22}}\cdot t}$, откуда время инфицирования с вероятностью $P_{inf} = 0.95$ определяется следующим образом: $t_{inf} = \frac{1}{\overline{\mu_{22}}}\cdot\ln\frac{1}{1-P_{inf}} \approx 650$ суток.

5.4. Некоторые модели динамики возникновения угроз, основанные на теории потоков

Необходимость моделирования динамики возникновения угроз безопасности информации, как правило, связана с угрозами проникновения в операционную среду ИС, то есть с угрозами, реализуемыми на системном и прикладном уровнях. Это определяется тем, что для оценки возможности их реализации за заданное время необходимо знать характеристики исходного потока угроз безопасности информации в ИС.

Динамика возникновения угроз безопасности информации в ИС обусловлена временными характеристиками процессов появления источников

угроз – злоумышленников (нарушителей), интенсивностью возникновения условий для физического доступа в помещение, к средствам вычислительной техники, к коммуникационным элементам компьютерной системы, интенсивностью попыток реализации угроз нарушителем, действиями пользователей, такими как включение, выключение компьютера, подсоединение его к компьютерной сети или отключение от сети и т.п. Указанные факторы характеризуются или статистической неопределенностью, или отсутствием точных сведений об их наличии и времени возникновения, или неопределенностью и неполнотой данных об отдельных характеристиках угроз безопасности информации, что предопределяет вероятностный характер процесса возникновения угроз безопасности информации. Большая часть таких факторов может быть учтена при применении для описания динамики возникновения угроз методов теории случайных потоков, что обусловлено случайным характером процесса возникновения угрозы, отсутствием, как правило, в таком процессе параллельных событий и возможностью представления его в виде последовательности однородных событий.

Возможность возникновения угрозы может быть оценена с применением показателя – вероятности возникновения угрозы за заданное время или характеризоваться среднестатистическими характеристиками – средней интенсивностью потока угроз и среднем временем существования угрозы.

Далее рассматриваются модели динамики возникновения двух групп угроз:

а) физического доступа к элементу ИС (непосредственного доступа к компьютеру);

б) удаленного несанкционированного доступа в операционную среду компьютера (угроз сетевых атак).

Математическая модель динамики возникновения угроз физического доступа к ИС и к ее элементам основана на том, что в общем случае каждая угроза существует эпизодически и может быть представлен в виде потока событий, показанного на рисунке 5.9.

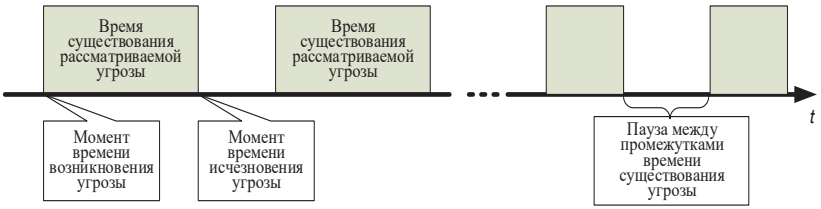


Рисунок 5.9 – Иллюстрация процесса возникновения угрозы безопасности информации во времени

Момент возникновения угрозы в общем случае определяется:

– интенсивностью появления источников данной угрозы $\left(\mu_{par}\right)$;

– долей (вероятностью) использования каждого момента появления источника для попытки реализации угрозы (P_{par});

– выбором и возможностью реализации способа проникновения, к которым могут относиться вывод из строя системы сигнализации и охраны ($s=1$), подбор необходимых параметров аутентификации ($s=2$), отвлечение персонала ($s=3$) или выбор момента его отсутствия ($s=4$) и др.;

– интенсивностью возникновения условий функционирования ИС (μ_{cond}), которые существенны для реализации данной угрозы (например, проникновение только в нерабочее время или только в рабочее время, когда функционирует ИС).

Продолжительность существования угрозы определяется:

– временем наличия источников угрозы (τ_{par});

– временем существования условий, которые существенны для реализации данной угрозы (τ_{cond}).

Если имеется несколько способов проникновения к i-му элементу ИС и тот или иной способ (s) выбирается нарушителем с некоторой вероятностью $P_{sel.i}\left(s\right), s=\overline{1,S_i}$ и не зависит от источника угрозы, то вероятность того, что нарушитель воспользуется одним из известных способов проникновения, находится из простого соотношения:

$$P_{sel.i}=1-\prod_{s=1}^{S_i}\left[1-P_{sel.i}\left(s\right)\right]. \qquad (5.70)$$

С учетом изложенного основные соотношения для определения характеристик потока угроз и вероятностей возникновения угрозы приведена в таблице 5.1.

Для использования данной модели необходимо прогнозировать средние интенсивности потоков инцидентов существования условий для реализации угроз, продолжительности таких инцидентов, а также временные характеристики потока угроз применительно к каждой ИС.

Математическая модель динамики возникновения угроз безопасности информации, связанных с удаленным доступом, определяется такими факторами, как интенсивность попыток проведения сетевых атак и моментом возникновения угрозы, что в общем случае определяется:

– интенсивностью появления источников данной угрозы (μ_{par});

– долей (вероятностью) использования каждого момента появления источника для попытки реализации угрозы (P_{par}).

Таблица 5.1 – Основные соотношения для расчета характеристик динамики возникновения угроз безопасности информации в ИС

Наименование модели	Условия возникновения угрозы	Основные соотношения для расчета	Примечания
Модель динамики возникновения угроз физического доступа к ИС и к ее элементам	1. Эпизодически появляется один источник угрозы. 2. Эпизодически возникают условия в КС, необходимые для реализации угрозы	$P_{beg,i}(t) = \{1 - \exp[-P_{set,i}\bar{\mu}_{cond,i}P_{par,i}\bar{\mu}_{par,i}(\bar{\tau}_{cond,i} + \bar{\tau}_{par,i})\,t]\}$; $\bar{\mu}_{beg,i} = P_{set,i} \cdot P_{par,i} \cdot \bar{\mu}_{cond,i} \cdot \bar{\mu}_{par,i} \cdot (\bar{\tau}_{cond,i} + \bar{\tau}_{par,i})$; $\bar{\tau}_{begi} = \dfrac{\bar{\tau}_{cond,i} \cdot \bar{\tau}_{par,i}}{\bar{\tau}_{cond,i} + \bar{\tau}_{par,i}}$	$P_{beg,i}(t)$ - вероятность возникновения i-й угрозы за время t; $P_{set,i}$ - вероятность выбора способа проникновения к элементам ИС; $P_{par,i}$ - вероятность использования источником в каждый момент своего появления попытки реализации угрозы; $\bar{\mu}_{beg,i}$, $\bar{\mu}_{par,i}$ - средние интенсивности возникновения i-й угрозы и появления ее источника соответственно; $\bar{\mu}_{cond,i}$ - интенсивность возникновения условий функционирования ИС, которые существенны для реализации i-й угрозы; $\bar{\tau}_{begi}$, $\bar{\tau}_{pari}$ - среднее время существования i-й угрозы и существования источника i-й угрозы соответственно; $\bar{\tau}_{cond,i}$ - среднее время существования условий в КС, необходимых для реализации i-й угрозы;
	1. Эпизодически появляется один источник угрозы. 2. Условия в ИС, необходимые для реализации угрозы, существуют всегда	$P_{beg,i}(t) = 1 - \exp[-\bar{\mu}_{par,i} \cdot P_{set,i} \cdot P_{par,i} \cdot t]$; $\bar{\mu}_{begi} = P_{set,i} \cdot P_{par,i}\bar{\mu}_{par,i}$	
	Условия для реализации угрозы случайны во времени, но при каждом таком условии осуществляется попытка реализации угрозы одним источником	$P_{beg,i}(t) = 1 - \exp[-\bar{\mu}_{cond,i}P_{set,i}t]$; $\bar{\mu}_{begi} = \bar{\mu}_{cond,i} \cdot P_{set,i}$	
	1. Эпизодически появляются несколько источников угрозы. 2. Эпизодически возникают условия в ИС, необходимые для реализации угрозы	$P_{beg,i}(t) = \{1 - \exp[-(\sum_{z \in Z_i} P_{set,iz}P_{par,z}\bar{\mu}_{par,z})\bar{\mu}_{cond,i}(\bar{\tau}_{cond,i} + \bar{\tau}_{par,i})t]\}$; $\bar{\mu}_{beg,i} = \sum_{z \in Z_i}[1 - \prod_{s \in S_i}[1 - P_{set,iz}(s)]]P_{par,i}\bar{\mu}_{par,i}\bar{\mu}_{cond,i}(\bar{\tau}_{cond,i} + \bar{\tau}_{par,i})$; $\bar{\tau}_{begi} = \dfrac{\bar{\tau}_{cond,i} \cdot \bar{\tau}_{par,i}}{\bar{\tau}_{cond,i} + \bar{\tau}_{par,i}}$	Те же, а также Z_i - множество источников i-й угрозы

Наименование модели	Условия возникновения угрозы	Основные соотношения для расчета	Примечания
Математическая модель динамики возникновения угроз безопасности информации, связанных с удаленным доступом при рекуррентном потоке угроз	1. Эпизодически возникают попытки удаленного НСД. 2. Эпизодически возникают условия в ИС для реализации НСД. Остальные условия не учитываются	$P_{beg}(t) = 1 - \exp[-\bar{\mu}_{cond}\bar{\mu}_{th}(\bar{\tau}_{cond} + \bar{\tau}_{th})P_{can}\, t]$; $P_{can} = 1 - \prod_{k=1}^{K_{can}}[1 - P_{can}(k)]$; $\bar{\mu}_{beg} = \bar{\mu}_{cond} \cdot \bar{\mu}_{th} \cdot (\bar{\tau}_{cond} + \bar{\tau}_{th})$; $\bar{\tau}_{beg} = \frac{\bar{\tau}_{cond} \cdot \bar{\tau}_{th}}{\bar{\tau}_{cond} + \bar{\tau}_{th}}$	$P_{beg}(t)$ - вероятность возникновения угрозы за время t; P_{can} - вероятность того, что хотя бы один из каналов НСД может быть использован нарушителем для реализации данной угрозы; $P_{can}(k)$ – вероятность использования k-го канала НСД для реализации рассматриваемой угрозы; K_{can} - количество каналов НСД; $\bar{\mu}_{th}$ - интенсивность попыток НСД; k_{can}^{att} - количество каналов НСД, которые можно задействовать для реализации данной угрозы в условиях применения мер и средств защиты
	1. Эпизодически возникают попытки удаленного НСД. 2. Поток попыток НСД является рекуррентным. 3. Условия в ИС для реализации НСД существуют всегда. 4. Время, необходимое для реализации угроз не ограничено	$P_{beg}(t) = 1 - \exp[-\bar{\mu}_{th}\, P_{can}\, t]$; $P_{can} = 1 - \prod_{k=1}^{K_{can}}[1 - P_{can}(k)]$; $P_{can} = \frac{k_{can}^{att}}{K_{can}}$	
	1. Эпизодически возникают попытки удаленного НСД. 2. Поток попыток НСД является рекуррентным. 3. Условия в ИС для реализации НСД существуют всегда. 4. Время, необходимое для реализации угроз ограничено	$P_{beg}(t) = 1 - \exp[-\bar{\mu}_{beg} \cdot P_{lim}(\tau_{th} \geq \tau_{u}) \cdot t]$; $P_{lim}(\tau_{th} \geq \tau_{u}) = \exp(-\tau_{u} / \bar{\tau}_{th})$; $\bar{\mu}_{beg} = \bar{\mu}_{cond} \cdot \bar{\mu}_{th} \cdot (\bar{\tau}_{cond} + \bar{\tau}_{th})$; $\bar{\tau}_{beg} = \frac{\bar{\tau}_{cond} \cdot \bar{\tau}_{th}}{\bar{\tau}_{cond} + \bar{\tau}_{th}}$	Те же, а также $P_{lim}(\tau_{th} \geq \tau_{u})$ – вероятность того, что продолжительность существования угроз τ_{u} превышает допустимое значение, при этом при меньшей продолжительности угрозы можно не рассматривать
Математическая модель динамики возникновения угроз безопасности информации, связанных с удаленным доступом при квазирекуррентном потоке угроз	1. Эпизодически возникают попытки удаленного НСД, но в период возможного существования условий для реализации НСД. 2. Поток попыток НСД является квазирекуррентным 3. Условия в ИС для реализации НСД эпизодически	$P_{beg}(t) = 1 + \frac{1}{\bar{\mu}_{cond} - \bar{\mu}_{th}}[\bar{\mu}_{th}\exp(-\bar{\mu}_{cond}t) - \bar{\mu}_{cond}\exp(-\bar{\mu}_{th}t)]$; $\bar{\mu}_{cond} = \frac{1}{\bar{\tau}_{cond} + \bar{\theta}_{cond}}$; $\bar{\mu}_{th} = \frac{1}{\bar{\tau}_{th} + \bar{\theta}_{th}}$	Те же, а также $\bar{\mu}_{cond}$ и $\bar{\mu}_{th}$ – средние интенсивности соответственно потоков угроз и промежутков времени, в которые они могут существовать; $\bar{\theta}_{cond}$ - средняя длительность паузы между периодами существования условий для реализации угроз; θ_{th} - средняя длительность паузы между попытками реализации угроз

Более того, даже при малом количестве каналов НСД, но при отсутствии мер и средств защиты эта вероятность также близка к единице, и лишь при применении необходимых мер и средств защиты – снижается.

В ряде случаев поток попыток реализации угроз не может быть отнесен к рекуррентным, поскольку при одном и том же сеансе работы компьютера в сети может состояться несколько попыток сетевых атак.

Существенным ограничением изложенного подхода сегодня является то, что рассматриваются только известные каналы НСД, то есть известные уязвимости программного и аппаратного обеспечения ИС. Вместе с тем на практике часто нарушители используют вновь выявляемые уязвимости, что обусловливает необходимость прогноза возникновения новых уязвимостей. В этой связи может быть использован предложенный в [95] подход к прогнозу количества вновь возникающих уязвимостей, где теоретически и экспериментально показано, что безусловная плотность распределения вероятности возникновения новых уязвимостей может быть представлена в виде:

$$f_N\left[n(t)\right] = \frac{(n+a-1)!}{n!\cdot(a-1)!}\cdot\left(\frac{t}{t+b}\right)^n\cdot\left(1-\frac{t}{t+b}\right)^a, \qquad (5.71)$$

где n – количество выявляемых новых уязвимостей за период времени t;

a,b – параметры распределения, подбираемые для конкретной операционной системы.

При этом количество возникающих уязвимостей характеризуется математическим ожиданием и дисперсией:

$$M\left[n(t)\right] = \frac{a\cdot t}{b} \text{ и } D\left[n(t)\right] = \frac{a\cdot t}{b\cdot(b+t)}. \qquad (5.72)$$

Такой подход позволяет строить аналитические прогнозные модели возникновения новых уязвимостей и с их учетом оценивать возможность возникновения новых угроз безопасности информации в ИС.

Таким образом, для аналитического моделирования процесса возникновения угроз безопасности информации необходимо знать среднестатистические характеристики возникновения угроз. На сегодняшний день статистика возникновения угроз безопасности информации в ИС практически отсутствует. В этих условиях или сама вероятность может быть оценена на основе применения аппарата нечетких множеств или оцениваются с помощью этого аппарата в нечетком виде среднестатистические характеристики потоков возникновения угроз безопасности информации. При этом разработанные модели не претерпевают существенных изменений за исключением того, что вместо обычных операций с числами применяются операции с нечеткими числами.

5.5. Общий подход к построению марковских моделей процессов возникновения и реализации угроз безопасности информации с дискретными состояниями и непрерывным временем

Угроза безопасности информации в ИС может быть реализована на сетевом, системном и прикладном уровнях[21] [26, 35].

На сетевом уровне реализация угроз направлена:

на перехват сообщений (перехват трафика), передаваемых по сетям общего пользования;

на нарушение функционирования узлов и коммуникационных элементов как внешних сетей, через которые осуществляется передача сообщений из защищаемой ИС, что приводит или к невозможности использования ею информационно-телекоммуникационной сети общего пользования, или к нарушению функционирования самой ИС.

Такие угрозы реализуются путем:

1) создания ложных узлов с последующем перенаправлением на них перехватываемого трафика;

2) создания на узле, доверенном для атакуемой ИС (хоста ИС), режима удаленного терминала;

3) внедрения в коммуникационные элементы по пути трафика атакуемого хоста или в доверенные для него узлы программных средств перехвата пакетов сообщений («снифферов»);

4) подмены доверенного хоста в сети общего пользования.

Если угроза реализуется в сети общего пользования (Internet), то динамика ее реализации определяется динамикой передачи перехватываемых сообщений, а меры защиты, как правило, связаны с криптографической защитой трафика и в данной работе далее не рассматриваются.

Если угроза реализуется в самой ИС, то процесс ее реализации будет определяться возможностью создания нарушителем ложного объекта или внедрения «сниффера», включения режима удаленного терминала и подмены физического адреса хоста ИС и др. Все эти действия осуществляются, как правило, заблаговременно и не связаны с динамикой реализации угрозы перехвата трафика в ИС, которая будет определяться, в основном, динамикой передачи сообщений в ИС. В этих условиях, как правило, вряд ли целесообразно моделировать процессы передачи подлежащих перехвату сообщений в ИС.

На *системном и прикладном уровнях* угрозы реализуются путем проникновения в операционную среду ИС (хоста ИС), то есть доступа к командам его ОС, к прикладным программам, к учетным записям, к системному реестру и т.д., а также с получением отклика на специально посылаемые пакеты сообщений, обрабатываемые ОС хостов. При этом на системном уровне для реализации угроз используются уязвимости ОС (в том числе драйверов, системных утилит и т.п.), а на прикладном уровне –

[21] Угрозы, реализуемые на микропрограммном уровне, в данной работе не рассматриваются.

уязвимости прикладных программ (программ пользователей, систем управления базами данных и т.п.).

Как правило, такие угрозы направлены:

на хищение (копирование или просмотр) или модификацию пользовательской информации (текстовых, графических, видеофайлов, в том числе записей в базах данных);

на выявление технологической информации, в том числе сетевого и физического адреса, типа ОС, состава установленных прикладных программ, состава и типажа подключенного периферийного оборудования и т.п.;

на выявление паролей доступа;

на изменение учетных записей;

на выявление директорий и каталогов с представляющей интерес информацией с целью последующего ее копирования;

на выявление состава принятых мер защиты информации и настроек применяемых средств защиты с целью последующего использования для реализации угроз хищения информации в данном компьютере;

на уничтожение, перемещение или нарушение доступности пользовательских файлов (текстовых, графических, видеофайлов, прикладных программ);

на внедрение вредоносных программ;

на несанкционированный запуск приложений;

на изменение настроек ОС, в том числе программных и программно-аппаратных средств защиты с целью обеспечения возможности скрытного доступа с другого компьютера этой же ИС или из внешней сети, если компьютер подключен к ней и т.д.

Кроме того, угрозы могут быть направлены на нарушение функционирования компьютера или отдельных прикладных программ.

Процесс реализации угрозы сетевой атаки в общем случае состоит из четырех этапов [26]:

– сбора информации;

– вторжения (проникновения в операционную среду);

– реализации деструктивных действий;

– ликвидации следов атаки.

Моделирование процесса реализации угроз, как правило, осуществляется в основном для этапов вторжения и реализации деструктивных действий. В этом случае процесс реализации угрозы за время t в отношении m-го блока защищаемой информации (файла, папки, директории, записи в базе данных и др.) может быть формально записан следующим образом:

$$R_u(t) = \left\{ Y_{uj}(\tau_{prj}) \right\} \oplus g_m(\tau_{gm}),\ j = \overline{1, J_{ug}},\ m = \overline{1, M_g^{(\inf)}}, \qquad (5.73)$$

где Y_{uj} – j-е действие случайной продолжительности τ_{prj}, направленное на проникновение в операционную среду хоста, входящее во множество действий мощностью $\aleph_u = J_{ug}$, необходимых для выполнения g_m-го

несанкционированного действия случайной продолжительности τ_{gm} в отношении m-го блока защищаемой информации при реализации u-й угрозы;

$\oplus$ – знак совокупности рассматриваемых действий;

$M_{\inf}^{(g)}$ – общее количество блоков защищаемой информации в хосте, относительно которого может быть выполнено g-е несанкционированное действие, соответствующее u-й угрозе.

Общая продолжительность процесса реализации угрозы является суммой времен выполнения парциальных действий, в том числе g-го несанкционированного действия с информацией:

$$\tau_u = \begin{cases} \sum_{j=1}^{J_{ug}} \tau_j^{(pr)} + \tau_m^{(g)} - \text{в случае одного } m - \text{го блока информации;} \\ \sum_{j=1}^{J_{ug}} \tau_j^{(pr)} + \sum_{b=1}^{M_{inf}^{(g)}} \tau_m^{(g)} - \text{в случае } M_{inf}^{(g)} \text{ блоков информации.} \end{cases} \quad (5.74)$$

При этом каждое действие начинается после окончания предыдущего действия и не зависит от времени выполнения всех ранее выполненных действий, то есть процесс реализации угрозы обладает свойством марковского процесса, когда будущее процесса не зависит от его прошлого, а определяется только его настоящим.

Если плотности распределения вероятностей указанных величин равны соответственно $f\left(\tau_j^{(pr)}\right)$ и $f\left(\tau_m^{(g)}\right)$, то плотность распределения вероятности для их суммы находится из соотношения:

$$f(\tau_u) = L^{-1}\left\{\prod_{j=1}^{J_{ud}} \chi(s, \tau_j^{(pr)}) \cdot \chi(s, \tau_m^{(g)})\right\}, \quad (5.75)$$

где $\chi\left(s, \tau_j^{(pr)}\right)$ и $\chi\left(s, \tau_m^{(g)}\right)$ – характеристические функции случайных величин $\tau_j^{(pr)}$ и $\tau_m^{(g)}$ соответственно;

L^{-1} – обратное преобразование Лапласа с параметром s.

Плотность распределения вероятностей указанных случайных величин, как правило, неизвестны, но поскольку эти величины являются положительно определенными, то плотности распределения могут быть аппроксимированы с любой точностью гамма-распределением [96, 92]:

$$f(x) = \frac{\lambda^{\alpha}}{\Gamma(\alpha)} \cdot x^{\alpha-1} \cdot e^{-\lambda \cdot x}, \quad (5.76)$$

где λ – средняя интенсивность, определяемая как обратная величина математического ожидания рассматриваемой случайной величины (например, $\lambda = \frac{1}{\overline{\tau_m^{(g)}}}$), а α – параметр γ-распределения, подбираемый при аппроксимации.

Для плотностей распределения в виде гамма-функции характеристические функции имеют вид:

$$\chi(s,\tau_j^{(pr)})=\frac{1}{(1+s\cdot\overline{\tau_j^{(pr)}})^{k_j^{(pr)}}};\quad \chi(s,\tau_m^{(g)})=\frac{1}{(1+s\cdot\overline{\tau_m^{(g)}})^{k_g}},\tag{5.77}$$

где $k_j^{(pr)}, k_g$ – параметры соответствующих γ-распределений.

В общем виде вероятность реализации угрозы за время t может быть рассчитана из соотношения [97]:

$$\begin{aligned}P_u(t)=1-\frac{1}{\left(k_1^{(pr)}-1\right)!}\cdot\sum_{k=1}^{k_1^{(pr)}}H_{1k}^{(pr)}\cdot t^{k_1^{(pr)}-k}\cdot e^{-\frac{t}{\overline{\tau_1^{(pr)}}}}-\\-\frac{1}{\left(k_2^{(pr)}-1\right)!}\cdot\sum_{k=1}^{k_2^{(pr)}}H_{2k}^{(pr)}\cdot t^{k_2^{(pr)}-k}\cdot e^{-\frac{t}{\overline{\tau_2^{(pr)}}}}-\frac{1}{\left(k_g-1\right)!}\cdot\sum_{k=1}^{k_g}H_k^{(g)}\cdot t^{k_g-k}\cdot e^{-\frac{t}{\overline{\tau_m^{(g)}}}},\end{aligned}\tag{5.78}$$

где $H_{1k}^{(pr)}, H_{2k}^{(pr)}, H_k^{(g)}$ – вспомогательные функции, определяемые по сходной для всех формуле

$$H_k^{(x)}=\frac{1}{(k-1)!\cdot(k_x-k)!}\cdot\frac{d^{k-1}}{ds^{k-1}}\cdot\left[\frac{\left(1+s\cdot\overline{\tau_x}\right)^{k_x}}{\left(1+s\cdot\overline{\tau_1^{(pr)}}\right)\cdot\left(1+s\cdot\overline{\tau_2^{(pr)}}\right)\cdot\left(1+s\cdot\overline{\tau_m^{(g)}}\right)\cdot}\right],\tag{5.79}$$

где вместо индекса x подставляется один из индексов для случайных величин $\tau_j^{(pr)}$ и $\tau_m^{(g)}$.

В простейшем случае экспоненциальных распределений и двух действиях в ходе проникновения в операционную среду (например, подбор пароля и изменение учетных записей) параметры $k_1^{(pr)}, k_2^{(pr)}$ и k_g равны единице, а вероятность того, что за время t угроза будет реализована относительно m-го блока информации, имеет следующий вид:

$$\begin{aligned}P_{mu}(t)=1-\frac{\overline{\tau_1^{(pr)}}^2\cdot e^{-\frac{t}{\overline{\tau_1^{(pr)}}}}}{(\overline{\tau_1^{(pr)}}-\overline{\tau_2^{(pr)}})(\overline{\tau_1^{(pr)}}-\overline{\tau_m^{(g)}})}-\frac{\overline{\tau_2^{(pr)}}^2\cdot e^{-\frac{t}{\overline{\tau_2^{(pr)}}}}}{(\overline{\tau_2^{(pr)}}-\overline{\tau_1^{(pr)}})(\overline{\tau_2^{(pr)}}-\overline{\tau_m^{(g)}})}\\-\frac{\overline{\tau_m^{(g)}}^2 e^{-\frac{t}{\overline{\tau_m^{(g)}}}}}{(\overline{\tau_m^{(g)}}-\overline{\tau_1^{(pr)}})(\overline{\tau_m^{(g)}}-\overline{\tau_2^{(pr)}})}\quad\text{при } \overline{\tau_1^{(pr)}}\neq\overline{\tau_2^{(pr)}},\ \overline{\tau_1^{(pr)}}\neq\overline{\tau_m^{(g)}} \text{ и } \overline{\tau_2^{(pr)}}\neq\overline{\tau_m^{(g)}}.\end{aligned}\tag{5.80}$$

Если $\overline{\tau_1^{(pr)}}=\overline{\tau_2^{(pr)}}=\overline{\tau^{(pr)}}\neq\overline{\tau_m^{(g)}}$, то формула (5.80) преобразуется к виду:

$$P_{mu}(t)=1+\frac{\overline{\tau^{(pr)}}\left(2\overline{\tau_m^{(g)}}-\overline{\tau^{(pr)}}\right)+t\cdot\left(\overline{\tau_m^{(g)}}-\overline{\tau^{(pr)}}\right)}{\left(\overline{\tau_m^{(g)}}-\overline{\tau^{(pr)}}\right)^2}\cdot e^{-\frac{t}{\overline{\tau^{(pr)}}}}-\frac{\overline{\tau_m^{(g)}}^2}{\left(\overline{\tau_m^{(g)}}-\overline{\tau^{(pr)}}\right)^2}\cdot e^{-\frac{t}{\overline{\tau_m^{(g)}}}}.\tag{5.81}$$

Тем не менее, рассчитывать вероятность реализации угрозы по громоздким формулам затруднительно, поэтому целесообразно использовать приближенную формулу, основанную на экспоненциальной зависимости, которая, как правило, с достаточной для практики точностью описывает поведение функции $P_{mu}(t)$:

$$P_{mu}(t) = 1 - e^{-\frac{t}{\overline{\tau_{mu}}}}, \tag{5.82}$$

где $\overline{\tau_{mu}}$ – математическое ожидание времени реализации угрозы относительно m-го блока информации, определяемой ф общем случае по фрмуле (5.75); Так как все суммируемые величины $\tau_j^{(pr)}$ и $\tau_m^{(g)}$ независимы, то среднее суммарное время реализации угрозы определяется по формуле (5.74) для одного m-го блока информации.При этом следует иметь в виду, что другое несанкционированное действие относительно рассматриваемого блока защищаемой информации соответствует другой угрозе.

Если рассматривать множество угроз, отличающихся только объектом воздействия (с одинаковым источником угроз, эксплуатируемой уязвимостью, способом реализации, содержанием несанкционированного действия), то математическое ожидание времени их реализации относительно множества блоков защищаемой информации $M_{inf}^{(g)}$ с учетом формулы (5.75) определяется из соотношения:

$$\overline{\tau_u\left(M_{inf}^{(g)}\right)} = \sum_{j=1}^{J_{ug}} \overline{\tau_j^{(pr)}} + \sum_{m=1}^{M_{inf}^{(g)}} \overline{\tau_m^{(g)}}. \tag{5.83}$$

При этом вероятность реализации угрозы оценивается по формуле (5.82), где вместо $\overline{\tau_{mu}}$ подставляется величина $\overline{\tau_u\left(M_{inf}^{(g)}\right)}$.

Если угроз безопасности информации, связанных с проникновением в операционную среду хоста, много, то при анализе возможности их реализации сначала необходимо провести их группирование по сходным элементам описания следующим образом.

1. Все угрозы разделяются на две группы, в одну из которых включаются угрозы с внешним источником, в другую – с внутренним источником.

2. Далее угрозы, входящие в каждую из двух групп, разделяют по подгруппам, соответствующим применяемым сходным (одинаковым) способам проникновения в операционную среду хоста.

3. Затем каждую подгруппу делят на подмножества, каждое из которых соответствует своему несанкционированному действию (независимо от того, какой блок информации будет подвергнут воздействию).

В результате будут образованы подмножества угроз, каждое из которых характеризуется одинаковыми источником, способом проникновения и деструктивным действием. Это позволяет существенно сократить объем работ по анализу возможностей реализации угроз относительно любого или всей совокупности блоков защищаемой информации в хосте ИС.

Пусть для r-й ($r=\overline{1,2}$) группы угроз может иметь место S_r способов проникновения в операционную среду, то есть имеется S_r подгрупп угроз. Каждая s_r-я подгруппа ($s_r=\overline{1,S_r}$) делится на $G(s_r)$ несанкционированных действий, которые могут быть выполнены относительно блоков защищаемой информации в хосте. Тогда применительно к одному m-му блоку защищаемой информации, s_r-му способу проникновения и $g(s_r)$ несанкционированному действию ($g(s_r)=\overline{1,G(s_r)}$) формула (5.83) преобразуется к виду:

$$\overline{\tau_{mu}\left[g(s_r)\right]}=\sum_{j=1}^{J_{ud}}\overline{\tau_j^{(pr)}(s_r)}+\overline{\tau_m\left[g(s_r)\right]}. \qquad (5.84)$$

А для всего множества блоков информации $M_{inf}\left[g(s_r)\right]$, относительного которых выполняется одинаковое несанкционированное действие $g(s_r)$, математическое ожидание времени реализации угроз составит величину:

$$\overline{\tau_u\left\{M_{inf}\left[g(s_r)\right]\right\}}=\sum_{j=1}^{J_{ug}}\overline{\tau_j^{(pr)}}+\sum_{m=1}^{M_{inf}\left[g(s_r)\right]}\overline{\tau_m\left[g(s_r)\right]}. \qquad (5.85)$$

При этом вероятность реализации угрозы рассчитывается по формуле (5.82) с подстановкой вместо $\overline{\tau_{mu}}$ в случае воздействия только на один m-й блок информации величины $\overline{\tau_u\left[g(s_r)\right]}$ и в случае воздействия на множество $M_{inf}\left[g(s_r)\right]$ блоков информации – величины $\overline{\tau_u\left\{M_{inf}\left[g(s_r)\right]\right\}}$.

Далее рассматриваются некоторые модели динамики реализации угроз в отдельном компьютере и модели сетевых атак.

5.6. Некоторые марковские модели процессов реализации угроз безопасности информации

5.6.1. Марковские модели динамики выполнения несанкционированных действий относительно блоков защищаемой информации в отдельном компьютере

Для реализации несанкционированных действий относительно блоков защищаемой информации (системной или пользовательской) нарушителю или инициируемому им процессу необходимо получить доступ в операционную среду компьютера с некоторыми, возможно ограниченными, правами доверенного пользователя (процесса). Далее несанкционированное действие выполняется или штатными (имеющимися в составе операционной системы) программами (командами), или с применением специально разработанных программ. При этом действия могут выполняться:

– относительно любого одного или нескольких произвольно выбранных блоков информации;

– относительно заранее известного (как правило, по имени) блока информации, местонахождение которого может быть как известным, так и неизвестным нарушителю. В последнем случае нарушитель должен провести поиск именованного блока информации;

– относительно одного или нескольких блоков информации, когда необходимость выполнения несанкционированного действия нарушитель должен определить после предварительного анализа их содержания;

– относительно одного или нескольких блоков информации, когда нарушитель выполняет несколько взаимосвязанных несанкционированных действий.

Кроме того, относительно одного или совокупности блоков информации может выполняться только одно несанкционированное действие или совокупность взаимосвязанных действий.

Для случая, когда выполняется только одно из несанкционированных действий относительно одного или нескольких блоков информации, соотношения для расчета вероятностно-временных характеристик динамики его выполнения, полученные на основе решения систем дифференциальных уравнений Колмогорова (см. раздел 5.2), приведены в таблице 5.2.

Значительно сложнее оказываются марковские модели выполнения некоторой совокупности взаимосвязанных несанкционированных действий относительно одного блока информации с его поиском или тем более относительно нескольких блоков информации.

Однако такие модели особенно важны в тех случаях, когда неизвестно, какое несанкционированное действие может быть выполнено в результате НСД. При выполнении совокупности несанкционированных действий имеется определенная семантическая и статистическая взаимосвязь между возможными сочетаниями деструктивных действий. В таблице 5.3 приведена матрица взаимосвязей различных совокупностей деструктивных действий, которые могут иметь место в компьютерной системе. При ограниченном времени имеется лишь определенная вероятность выполнения того или иного сочетания деструктивных действий, уменьшающаяся с уменьшением отведенного для этого времени.

Первое действие в кортеже выполняется по мере получения доступа к блоку информации за время τ_{g1}, второе после первого за время τ_{g2} и т.д.

Суммарное время выполнения взаимосвязанных действий, входящих в подмножество Ω_n, составляет величину:

$$\tau_{\Omega_n} = \overline{\tau_{pr}} + \sum_{\omega \in \Omega_n} \tau_{g\omega}, \qquad (5.86)$$

где $\overline{\tau_{pr}}$ – среднее время, необходимое для получения несанкционированного доступа;

Таблица 5.2 – Вероятностно временные характеристики динамики выполнения деструктивных действий в отдельном компьютере при моделировании с использованием аппарата марковских процессов

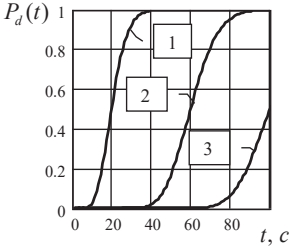
Наименование модели	Условия моделирования	Соотношения для расчета вероятностно – временных характеристик	Принятые обозначения
Модель динамики выполнения деструктивного действия относительно любого одного блока информации	Нарушитель случайно выбирает любой из блоков информации	$P_g(t) = 1 - e^{-\frac{t}{\overline{\tau_g}}}$	$\overline{\tau_g}$ - среднее время, необходимое для выполнения деструктивного действия
Модель динамики выполнения деструктивного действия относительно многих блоков информации	Блоки информации (файлы), относительно которых выполняется это действие, выбираются случайно	$P_g(V_g) = 1 - \sum_{r=0}^{n \cdot J - 1} \frac{e^{-\lambda_g V_g} (\lambda_g \cdot V_g)^r}{r!}$, $V_g = \sum_{j=1}^{J} V_{g,j}$; $\lambda_g = \frac{1}{\overline{\tau_g}} \approx \frac{1}{k \cdot \overline{V_g}}$; $\overline{\tau_g} = \frac{d}{d\tau_g} \frac{1}{(1 + s \cdot \tau_g)^{n \cdot J}} \Big\|_{\tau_s = 0}$; $\tau_g = \sum_{j=1}^{J} [\tau_{sj} + \tau_{g,j}(V_{g,j})]$ – при последовательном перемещения курсора от файла (блока) к файлу (блоку); $\tau_g = \tau_s + \sum_{j=1}^{J} \tau_{g,j}$ – при выделении группы блоков (каталогов, файлов) Пример расчета для процедуры уничтожения файлов приведен на рисунке 5.10	τ_g и $\tau_{g,j}$ - время, в течение которого происходит деструктивное действие относительно всех блоков информации и относительно j-го блока информации соответственно; V_g и $V_{g,j}$ – объем всех и j-го блока информации, относительно которого выполняется деструктивное действие; J - общее количество блоков информации; λ_g и n – параметры аппроксимирующего γ-распределения; $\tau_{s,j}, \tau_s$ – среднее время перемещения курсора на j-й блок информации и выделения группы блоков
Модель динамики выполнения деструктивного действия относительно заранее известного именованного блока информации	Нарушитель осуществляет поиск блока информации по его имени "вручную" и выполняет деструктивное действие без анализа содержания. Граф состояний процесса выполнения деструктивного действия с предварительным поиском необходимой нарушителю информации и система соответствующих дифференциальных уравнений приведена на рисунке 5.11	$P_g(t) = 1 - Ae^{-\left(\frac{a - \sqrt{a^2 - 4b}}{2}\right)t} + Be^{-\left(\frac{a + \sqrt{a^2 - 4b}}{2}\right)t}$, $A = \frac{2b}{(a - \sqrt{a^2 - 4b})\sqrt{a^2 - 4b}}$, $B = \frac{2b}{(a + \sqrt{a^2 - 4b})\sqrt{a^2 - 4b}}$; $a = \lambda_s + \lambda_w + \lambda_g$, $b = \lambda_s \lambda_g$. В первом приближении $P_g(t) \approx 1 - \exp\left[- \frac{t}{\overline{\tau_g} + \overline{\tau_s} \cdot \left(1 + \frac{\overline{\tau_g}}{\overline{\tau_w}}\right)} \right]$. Результаты расчета вероятности при точном и приближенном вычислении приведены на рисунке 5.12	$\lambda_s = \frac{1}{\overline{\tau_s}}$ – интенсивность поиска информации ($\overline{\tau_s}$ – среднее время поиска файла); $\lambda_w = \frac{1}{\overline{\tau_w}}$ - интенсивность перемещения курсора на очередной файл ($\overline{\tau_w}$ – среднее время перемещения курсора на очередной файл); $\lambda_g = \frac{1}{\overline{\tau_g}}$ - средняя интенсивность выполнения деструктивного действия ($\overline{\tau_g}$ - среднее время выполнения деструктивного действия).

Наименование модели	Условия моделирования	Соотношения для расчета вероятностно – временных характеристик	Принятые обозначения
	Нарушитель осуществляет поиск блока информации по его имени (расширению, дате создания и т.п.) с использованием штатной программы поиска и выполняет деструктивное действие без анализа содержания	$P_{\text{д}}(t)=1-\frac{\bar{\tau}_s\cdot e^{-\frac{t}{\bar{\tau}_s}}}{(\bar{\tau}_s-\bar{\tau}_{\text{д}})}-\frac{\bar{\tau}_{\text{д}}\cdot e^{-\frac{t}{\bar{\tau}_{\text{д}}}}}{(\bar{\tau}_{\text{д}}-\bar{\tau}_s)}$	$\bar{\tau}_s$ - среднее время поиска файла; $\bar{\tau}_{\text{д}}$ - среднее время выполнения деструктивного действия
Модель динамики выполнения деструктивного действия относительно блока информации, когда нарушитель должен определить необходимость выполнения деструктивных действий после предварительного анализа его содержания.	Нарушитель осуществляет поиск блока информации по его имени "вручную" и выполняет деструктивное действие после анализа содержания. Граф состояний процесса и система соответствующих дифференциальных уравнений приведены на рисунке 5.13	$P_{\text{д}}(t)\approx 1-\exp\{-\frac{t}{\chi'(s)\big\|_{s=0}}\}$, $\chi'(s)\Big\|_{s=0}=\bar{\tau}_{dec}+\bar{\tau}_{\text{д}}+\frac{1}{P_{dec}}(\bar{\tau}_s+\bar{\tau}_{an}+\bar{\tau}_{ir})$; График зависимости приведен на рисунке 5.14.	$\bar{\tau}_s$ - среднее время поиска блока информации; $\bar{\tau}_{ir}$ - среднее время возврата к поиску нового блока информации, $\bar{\tau}_{an}$ - среднее время анализа блока информации; $\bar{\tau}_{dec}$ - среднее время, необходимое для принятия решения на выполнение деструктивного действия; P_{dec} - вероятность принятия решения на деструктивное действие относительно проанализированного блока информации; $\bar{\tau}_{\text{д}}$ – среднее время выполнения деструктивного действия; $\chi'(s)$– производная от характеристической функции случайного суммарного времени поиска, анализа содержания и выполнения деструктивного действия; S – параметр преобразования Лапласа

Таблица 5.3 – Взаимосвязь возможных сочетаний несанкционированных действий* в ИС

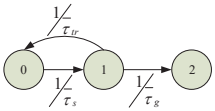
Номер совокуп-ности действий	Наименование действия (совокупности действий)	Попарная взаимосвязь сочетаний деструктивных действий (совокупности действий)					
		Уничтожение	Копирование	Модификация файла	Перемещение	Запуск исполняемого файла	Блокирование
1	Уничтожение	1	0	0	0	0	0
2	Копирование	1	1	1	1	1	1
3	Модификация файла	0	0	1	1	1	1
4	Перемещение файла	0	1	1	1	1	1
5	Запуск исполняемого файла	1	1	1	1	1	1
6	Блокирование	0	0	0	0	0	1
7	Копирование и модификация	0	0	0	1	1	0
8	Копирование и перемещение	0	0	1	0	1	1
9	Копирование и блокирование	0	0	0	0	0	0
10	Модификация и перемещение	0	0	0	0	1	1
11	Модификация и запуск	1	0	0	1	0	1
12	Копирование, модификация и перемещение	0	0	0	0	1	1
13	Копирование, модификация и блокирование	0	0	0	0	0	0
14	Копирование, модификация и запуск	0	0	0	0	0	1
16	Перемещение, модификация, запуск	1	0	0	1	0	1

*Во втором столбце указаны статистически независимые действия или сочетания действий. Сначала читается действие или совокупность деструктивных действий во втором столбце и в соответствующей строке при наличии в ней единицы в одном из столбцов, определяется действие, указанное в столбце, которое может быть дополнительно выполнено после выполнения выбранной совокупности действий. Если во втором столбце указывается только одно действие, то ищется второе действие, которое может быть выполнено после него за рассматриваемый период времени, если два, – то третье действие, если три, – то четвертое (маловероятно выполнение большего количества действий относительно одного и того же файла).



1) $n \cdot J = 10$; 2) $n \cdot J = 30$; 3) $n \cdot J = 50$; при $\lambda_{del} = 0.5\frac{1}{c}$

Рисунок 5.10 – Зависимость вероятности уничтожения не менее J файлов с общим объемом V_{del} от времени



0 – состояние, когда нарушитель начинает поиск информации;
1 – нарушитель просмотрел очередной файл на предмет отнесения его к искомому и либо продолжает поиск, либо предпринимает несанкционированное действие по отношению к нему;
2 – состояние, когда несанкционированное действие в отношении найденного файла завершено;

$\lambda_s = \frac{1}{\tau_s}$ – интенсивность поиска информации;

$\lambda_{tr} = \frac{1}{\tau_{tr}}$ – интенсивность перемещения курсора на очередной файл;

$\lambda_g = \frac{1}{\tau_g}$ средняя интенсивность выполнения деструктивного (несанкционированного) действия;

$$\begin{cases} P_0'(t) = -\lambda_s \cdot P_0(t) + \lambda_{tr} \cdot P_1(t); \\ P_1'(t) = \lambda_s \cdot P_0(t) - (\lambda_{tr} + \lambda_g) \cdot P_1(t); \\ P_2'(t) = \lambda_g \cdot P_1(t), \end{cases}$$

при начальный условиях: $P_0(0) = 1, P_1(0) = P_2(0) = 0; P_0'(0) = P_1'(0) = P_2'(0) = 0$

Рисунок 5.11 – Граф состояний процесса выполнения несанкционированного действия в случае поиска необходимой нарушителю информации и соответствующая ему система дифференциальных уравнений Колмогорова

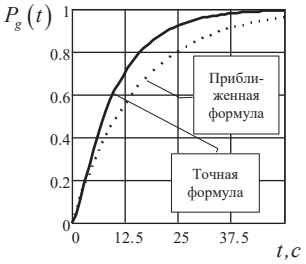
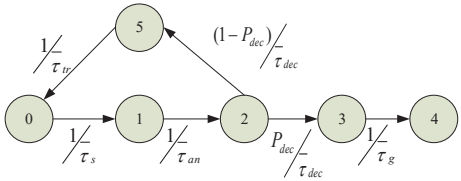


Рисунок 5.12 – Графики зависимости от времени вероятности реализации угрозы, представленной в графическом виде на рисунке 5.10, для точной и приближенной формул расчета (см. таблицу 5.2)



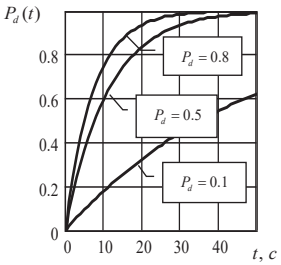
0 – состояние, когда нарушитель начинает поиск информации;
1 – нарушитель завершил поиск очередного файла и начинает анализировать его содержание;
2 – состояние, когда нарушитель завершил анализ содержания файла;
3 – нарушитель принял решение на проведение несанкционированного действия;
4 – нарушитель завершил выполнение несанкционированного действия;
5 – нарушитель не нашел нужного файла и принял решение начать поиск другого файла;

$\bar{\tau}_s$ – среднее время поиска блока информации; $\bar{\tau}_{tr}$ – среднее время возврата к поиску нового блока информации; $\bar{\tau}_{an}$ – среднее время анализа блока информации; $\bar{\tau}_{dec}$ – среднее время, необходимое для принятия решения на выполнение деструктивного действия; P_{dec} – вероятность принятия решения на деструктивное действие относительно проанализированного блока информации; $\bar{\tau}_g$ – средняя интенсивность выполнения деструктивного действия;

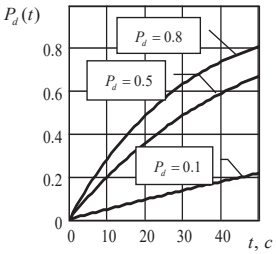
$$\begin{cases} P_0'(t) = -\frac{1}{\bar{\tau}_s}\cdot P_0(t) + \frac{1}{\bar{\tau}_{tr}}\cdot P_5(t); P_1'(t) = \frac{1}{\bar{\tau}_s}\cdot P_0(t) - \frac{1}{\bar{\tau}_{an}}\cdot P_1(t); P_2'(t) = \frac{1}{\bar{\tau}_{an}}\cdot P_1(t) - \frac{1}{\bar{\tau}_{dec}}\cdot P_2(t); \\ P_3'(t) = \frac{P_{dec}}{\bar{\tau}_{dec}}\cdot P_2(t) - \frac{1}{\bar{\tau}_g}\cdot P_3(t); P_4'(t) = \frac{1}{\bar{\tau}_g}\cdot P_3(t); P_5'(t) = \frac{1-P_{dec}}{\bar{\tau}_{dec}}\cdot P_2(t) - \frac{1}{\bar{\tau}_{tr}}\cdot P_5(t), \end{cases}$$

при начальных условиях: $P_0(0)=1, P_1(0)=P_2(0)=P_3(0)=P_4(0)=0; P_0'(0)=P_1'(0)=P_2'(0)=P_3'(0)=P_4'(0)=0;$ и $P_0(t)+P_1(t)+P_2(t)+P_3(t)=1$

Рисунок 5.13 – Граф состояний процесса выполнения несанкционированного действия относительно блока информации и система соответствующих процессу дифференциальных уравнений

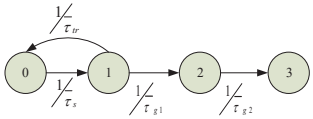


а) $\bar{\tau}_{dec}+\bar{\tau}_d = 1c$, $\bar{\tau}_s+\bar{\tau}_{an}+\bar{\tau}_{tr} = 5c$



б) $\bar{\tau}_{dec}+\bar{\tau}_d = 5c$, $\bar{\tau}_s+\bar{\tau}_{an}+\bar{\tau}_{tr} = 20c$

Рисунок 5.14 – Зависимости вероятности выполнения деструктивного действия от времени и параметров динамики процесса подготовки и выполнения деструктивного действия



0 – состояние, когда нарушитель начинает поиск информации (компьютер включен, операционная среда загружена);

1 – нарушитель просмотрел очередной файл на предмет отнесения его к искомому и либо продолжает поиск, либо принял решение на выполнение несанкционированного действия;

2 – нарушитель выполнил первое несанкционированное действие из предусмотренного кортежа;

3 – нарушитель выполнил первое несанкционированное действие из предусмотренного кортежа;

$$\begin{cases} P_0'(t) = -\dfrac{1}{\tau_s}\cdot P_0(t) + \dfrac{1}{\tau_{tr}}\cdot P_1(t); \\ P_1'(t) = \dfrac{1}{\tau_s}\cdot P_0(t) - \dfrac{1}{\tau_{tr}}\cdot P_1(t) - \dfrac{1}{\tau_{g1}}\cdot P_1(t); \\ P_2'(t) = \dfrac{1}{\tau_{g1}}\cdot P_1(t) - \dfrac{1}{\tau_{g2}}\cdot P_2(t); \\ P_3'(t) = \dfrac{1}{\tau_{g2}}\cdot P_2(t), \end{cases}$$

при начальных условиях: $P_0(0)=1, P_1(0)=P_2(0)=P_3(0)=0; P_0'(0)=P_1'(0)=P_2'(0)=P_3'(0)=0$ и $P_0(t)+P_1(t)+P_2(t)+P_3(t)=1$

Рисунок 5.15 – Граф состояний процесса выполнения совокупности двух деструктивных действий в случае поиска необходимой нарушителю информации

Если последовательность выбора блоков информации нарушителем произвольная и равновероятная, относительно каждого блока могут быть проведены деструктивные действия из любого сочетания, входящего в множество Ω, то общее количество вариантов выполнения деструктивных действий относительно J блоков информации составит величину $N^J \cdot J!$. Каждый вариант представляет собой в этом случае кортеж действий относительно определенной последовательности (перестановки) блоков информации. Если обозначить номер каждой перестановки через $r \in R$, где $R = J!$, а номер каждого кортежа действий (в том числе и повторяющихся от блока к блоку) через $z \in Z$, то плотность вероятности выполнения кортежа деструктивных действий для r-й перестановке определяется по формуле:

$$w_{gr}(\Omega_{nr}, t) = L^{-1}\left\{\frac{1}{(1+s\cdot\bar{\tau}_{pr})}\cdot\prod_{j\in J_r}\prod_{n=1}^{N_j}\frac{1}{(1+s\cdot\bar{\tau}_{\Omega_{nj}})}\right\}, \qquad (5.92)$$

где J_r – множество номеров, соответствующее r-й перестановке блоков информации, а вероятность выполнения деструктивных действий и среднее время – по формулам (5.91) и (5.92) соответственно.

5.6.2. Марковские модели динамики проникновения в операционную среду компьютера и выполнения несанкционированного действия

Осуществив физический доступ к компьютеру, нарушитель имеет возможность проникнуть в операционную среду компьютера и провести деструктивные действия. При проникновении в операционную среду компьютера нарушитель получает права доверенного пользователя (возможно не в полном объеме) для выполнения штатных программ и команд,

предусмотренных в операционной системе. В данной работе рассмотрены некоторые наиболее характерные случаи проникновения в операционную среду компьютера, когда или меры защиты отсутствуют, или из мер защиты могут применяться лишь первоочередные меры, например, идентификации и аутентификации[22]:

а) при включенном, но оставленном без присмотра пользователя компьютере;

б) при выключенном компьютере, когда нарушитель вынужден включить компьютер и загрузить операционную среду при отсутствии паролей для входа как в базовую систему ввода/вывода (BIOS), так и в операционную среду;

в) при выключенном компьютере, когда нарушитель вынужден включить компьютер и загрузить операционную среду при наличии пароля в базовой системе ввода/вывода (BIOS) и пароля для загрузки операционной системы.

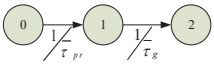
Процесс проникновения нарушителю в операционную среду компьютера представляется в виде процесса обслуживания заявок (попыток проникновения) стохастической системой массового обслуживания с одним прибором без очереди с ограниченным временем обслуживания каждой заявки [83]. При этом графически такой процесс представляется в виде ориентированного графа состояний и переходов, которому соответствует система дифференциальных стохастических уравнений Колмогорова, решение которой позволяет найти вероятности проникновения нарушителя за заданное время в операционную среду в зависимости от существенных параметров, определяющих возможности такого проникновения. Возможность применения такого подхода обусловлена тем, что процесс проникновения, как правило, представляет собой простую последовательность действий со случайными продолжительностями выполнения во времени, что позволяет использовать указанный выше аппарат.

Рассмотрим модель динамики проникновения в операционную среду включенного, но оставленного без присмотра, компьютера и выполнения деструктивного действия. Процесс проникновения в операционную среду компьютера и выполнения деструктивного действия может быть представлен в виде марковского процесса, граф состояний которого соответствует приведенному на рисунке 5.16.

Для такого процесса вероятность того, что нарушитель сумеет проникнуть в операционную среду компьютера и выполнить деструктивное действие за время t, рассчитывается по формуле:

$$P_u(t) = 1 + \frac{\overline{\tau_{pr}}}{(\overline{\tau_g} - \overline{\tau_{pr}})} \cdot e^{-\frac{t}{\overline{\tau_{pr}}}} + \frac{\overline{\tau_g}}{(\overline{\tau_{pr}} - \overline{\tau_{gr}})} \cdot e^{-\frac{t}{\overline{\tau_{gr}}}}. \qquad (5.93)$$

[22] Не рассматривается, например, случай загрузки с отчуждаемого носителя альтернативной операционной системы.



0 – состояние, когда нарушитель начинает попытки проникновения в операционную среду компьютера;
1 – нарушитель получил доступ в операционную среду;
2 – нарушитель выполнил деструктивное действие;

$$P_0'(t) = -\frac{1}{\overline{\tau_{pr}}} \cdot P_0(t);$$

$$P_1'(t) = \frac{1}{\overline{\tau_{pr}}} \cdot P_0(t) - \frac{1}{\overline{\tau_g}} \cdot P_1(t);$$

$$P_2'(t) = \frac{1}{\overline{\tau_g}} \cdot P_1(t),$$

при начальных условиях $P_0(0) = 1, P_1(0) = P_2(0) = 0; P_0'(0) = P_1'(0) = P_2'(0) = 0$ и $P_0(t) + P_1(t) + P_2(t) = 1$

Рисунок 5.16 – Граф состояний марковского процесса проникновения в операционную среду компьютера и выполнения несанкционированного действия

При этом среднее время проникновения и выполнения несанкционированного действия составляет величину: $\overline{\tau_u} = \overline{\tau_{pr}} + \overline{\tau_g}$.

Рассмотрим модель проникновения в операционную среду компьютера и выполнения несанкционированного действия при необходимости его включения.

Если не установлены ограничения на доступ посторонних лиц, то процесс проникновения в операционную среду обусловлен временем, необходимым на загрузку операционной среды и временем, необходимым на выполнение несанкционированного действия.

В качестве несанкционированного действия, не снижая общности, рассмотрим уничтожение файлов.

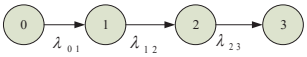
При случайном выборе файла, относительно которого выполняется несанкционированное действие (при этом поиск файла отсутствует) процесс проникновения в операционную среду, перемещения по файловой системе и уничтожения файла может быть представлен в виде графа состояний, показанного на рисунке 5.17.

Пусть средние интенсивности переходов из состояния в состояние равны соответственно $\lambda_{01} = \frac{1}{\overline{\tau_{acc}}}$, $\lambda_{12} = \frac{1}{\overline{\tau_{com}}}$, $\lambda_{23} = \frac{1}{\overline{\tau_g}}$,

где $\overline{\tau_{acc}}$ – среднее время между попытками физического доступа к компьютеру;

$\overline{\tau_{com}}$ – среднее время, необходимое для загрузки компьютера;

$\overline{\tau_g}$ – среднее время, необходимое для выполнения деструктивного действия.



0 – состояние, когда нарушитель начинает попытки физического доступа к компьютеру;
1 – нарушитель получил доступ к компьютеру и включил его, началась загрузка операционной среды;
2 – операционная среда загрузилась и нарушитель начал выполнения несанкционированное действие;
3 – несанкционированное действие выполнено;

$$P_0'(t) = -\lambda_{01} \cdot P_0(t);$$
$$P_1'(t) = \lambda_{01} \cdot P_0(t) - \lambda_{12} \cdot P_1(t);$$
$$P_2'(t) = \lambda_{12} \cdot P_1(t) - \lambda_{23} \cdot P_2(t);$$
$$P_3'(t) = \lambda_{23} \cdot P_2(t),$$

при начальных условиях $P_0(0) = 1, P_1(0) = P_2(0) = P_3(0) = 0; P_0'(0) = P_1'(0) = P_2'(0) = P_3'(0) = 0$ и $P_0(t) + P_1(t) + P_2(t) + P_3(t) = 1$

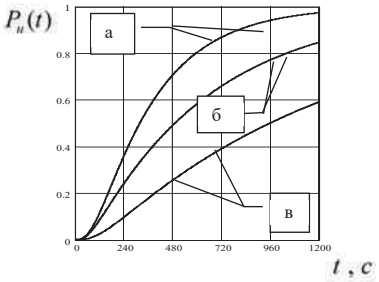
Рисунок 5.17 – Граф состояний процесса уничтожения одного файла или каталога нарушителем

Решение системы дифференциальных уравнений, соответствующей приведенному графу, для вероятности проникновения в операционную среду компьютера и выполнения деструктивного действия имеет вид:

$$P_u(t) = P_3(t) = 1 - \frac{Ae^{-\lambda_{01} \cdot t}}{\lambda_{01}(\lambda_{12} - \lambda_{01})(\lambda_{23} - \lambda_{01})} - \frac{Ae^{-\lambda_{12} \cdot t}}{\lambda_{12}(\lambda_{01} - \lambda_{12})(\lambda_{23} - \lambda_{12})} - \frac{Ae^{-\lambda_{23} \cdot t}}{\lambda_{23}(\lambda_{01} - \lambda_{23})(\lambda_{12} - \lambda_{23})}, \qquad (5.94)$$

где $A = \lambda_{01} \cdot \lambda_{12} \cdot \lambda_{23}$ – вспомогательный параметр.

Полученная зависимость в виде графиков для типовых параметров приведена на рисунке 5.18.



а – $\bar{\tau}_{acc} = 300c$, $\bar{\tau}_{com} = 40\ c$, $\bar{\tau}_g = 60\ c$; б – $\bar{\tau}_{acc} = 600c$, $\bar{\tau}_{com} = 40\ c$, $\bar{\tau}_g = 30\ c$;
в – $\bar{\tau}_{acc} = 1200c$, $\bar{\tau}_{com} = 50\ c$, $\bar{\tau}_g = 70\ c$ (жирные линии соответствуют расчетам по формуле (6.88), а тонкая и пунктирные линии – экспоненциальному приближению)

Рисунок 5.18 – Зависимости от времени вероятности проникновения в операционную среду компьютера и выполнения несанкционированного действия

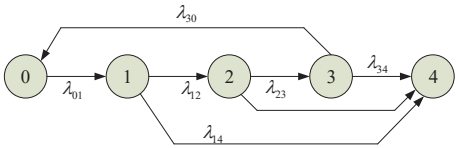
Среднее время, необходимое для реализации угрозы, определяется из соотношения:

$$\bar{\tau}_u = \frac{1}{\lambda_{01}} + \frac{1}{\lambda_{12}} + \frac{1}{\lambda_{23}} = \bar{\tau}_{acc} + \bar{\tau}_{com} + \bar{\tau}_{dd}\,. \qquad (5.95)$$

Для сравнения на рисунке 5.18 показаны графики экспоненциальной зависимости: $P_u(t) = 1 - e^{-\frac{t}{\bar{\tau}_u}}$, из которых видно, что ошибка в оценках вероятности реализации угрозы не превышает 10% в начальный период времени и становится весьма малой с увеличением времени реализации угрозы.

Это подтверждает целесообразность применения более простых соотношений для оценки указанных вероятностей при ориентировочных оценках возможностей реализации угрозы.

Рассмотрим *модель динамики проникновения в операционную среду компьютера, когда нарушитель вынужден включить компьютер и загрузить операционную среду при нелобходимости ввода пароля для базовой системы ввода/ вывода (BIOS, UEFI) и пароля для загрузки операционной системы.* Тогда граф состояний процесса проникновения в операционную среду компьютера и соответствующая система дифференциальных уравнений для этого случая представлен на рисунке 5.19. λ_{34}



0 – нарушитель получил физический доступ к компьютеру и включил его, пытается подобрать пароль BIOS;
1 – нарушителю удалось подобрать пароль BIOS, однако операционная система запросила пароль для получения доступа в операционную среду, нарушитель начал подбор пароля для получения доступа в операционную среду первый раз;
2 – пароль доступа в операционную среду первый раз подобрать не удалось и нарушитель делает вторую попытку;
3 – пароль доступа при второй попытке подобрать не удалось и нарушитель предпринимает последнюю попытку;
4 – пароль подобран и нарушитель получил доступ в операционную среду компьютера и готов к выполнению несанкционированного действия;

$$\begin{cases} P_0'(t) = -\lambda_{01} \cdot P_0(t) + \lambda_{34} \cdot P_3(t); \\ P_1'(t) = \lambda_{01} \cdot P_0(t) - \lambda_{12} \cdot P_1(t) - \lambda_{14} \cdot P_1(t); \\ P_2'(t) = \lambda_{12} \cdot P_1(t) - \lambda_{23} \cdot P_2(t) - \lambda_{24} \cdot P_2(t); \\ P_3'(t) = \lambda_{23} \cdot P_2(t) - \lambda_{34} \cdot P_3(t) - \lambda_{30} \cdot P_3(t); \\ P_4'(t) = \lambda_{14} \cdot P_1(t) + \lambda_{24} \cdot P_2(t) + \lambda_{34} \cdot P_3(t), \end{cases}$$

при начальных условиях $P_0(0) = 1; P_1(0) = P_2(0) = P_3(0) = P_4(0), P_0'(0) = P_1'(0) = P_2'(0) = P_3'(0) = P_4'(0) = 0$

Рисунок 5.19 – Граф состояний марковского процесса получения доступа в операционную среду компьютера и выполнения несанкционированного действия при необходимости преодоления парольной защиты

При этом интенсивности переходов из состояния в состояние определяются следующим образом:

$$\lambda_{01} = \frac{P_{BIOS}}{\bar{\tau}_{BIOS}}; \lambda_{12} = \lambda_{23} = \lambda_{30} = \frac{1 - P_{os}}{\bar{\tau}_{os}} = \lambda_1; \lambda_{14} = \lambda_{24} = \lambda_{34} = \frac{P_{os}}{\bar{\tau}_{os}} = \lambda_2, \qquad (5.96)$$

где P_{BIOS} и $\bar{\tau}_{BIOS}$ – вероятность подбора и среднее время набора пароля BIOS;

P_{os} и $\bar{\tau}_{os}$ – вероятность подбора и средняя продолжительность набора пароля доступа в операционную систему;

λ_1, λ_2 – вспомогательные параметры для обозначения равных по величине интенсивностей.

Вероятность (P_{BIOS} или P_{os}) и среднее время ($\bar{\tau}_{BIOS}$ или $\bar{\tau}_{os}$) подбора пароля путем прямого перебора рассчитывается по известным формулам [98]:

$$P_{pass} \approx \frac{n}{A^s}; \overline{\tau_{pass}} = \frac{A^s \cdot T_{pass}}{2}, \tag{5.97}$$

где n – количество попыток подбора пароля;

A – число символов в алфавите, на основе которого сформирован пароль;

s – длина пароля (количество символов в пароле);

T_{pass} – время выполнения одной попытки ввода пароля.

Решение соответствующей приведенному графу системы дифференциальных уравнений относительно вероятности нахождения в последнем состоянии имеет следующий вид:

$$P_{acc}(t) = L^{-1}\left\{\frac{Ch(s)}{Zn(s)}\right\} \approx 1 - e^{-\frac{t}{\overline{\tau_{acc}}}}, \tag{5.98}$$

где $Ch(s) = \lambda_{01}[\lambda_1^2\lambda_2 + \lambda_2(s+\lambda_1+\lambda_2)^2 + \lambda_1\lambda_2(s+\lambda_1+\lambda_2)$;

$$\begin{aligned} Zn(s) = s[&s(s+\lambda_{01}+\lambda_1+\lambda_2)(s+\lambda_1+\lambda_2) + s\lambda_{01}\lambda_1(s+\lambda_1+\lambda_2) + \\ &+ \lambda_{01}\lambda_1\lambda_2(s+\lambda_1+\lambda_2) + s\lambda_1^2\lambda_{01} + \lambda_{01}\lambda_2(s+\lambda_1+\lambda_2)^2]; \end{aligned} \tag{5.99}$$

$\overline{\tau_{acc}}$ – среднее время реализации угрозы доступа,

$$\overline{\tau_{acc}} = \frac{\overline{\tau_{BIOS}}}{P_{BIOS}} + \frac{3(1-P_{os})\overline{\tau_{os}} + \overline{\tau_{os}} + (1-P_{os})^3 \cdot \frac{\overline{\tau_{BIOS}}}{P_{BIOS}}}{P_{os}[1+(1-P_{os})+(1-P_{os})^2]}. \tag{5.100}$$

Графики зависимости среднего времени доступа и вероятности получения доступа для различных значений временных параметров набора паролей и вероятностей их подбора представлены на рисунках 5.20 – 5.22.

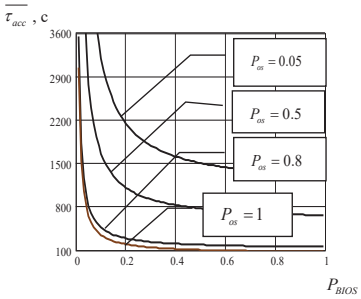


Рисунок 5.20 – Зависимость среднего времени реализации угрозы доступа от вероятности подбора пароля доступа в BIOS

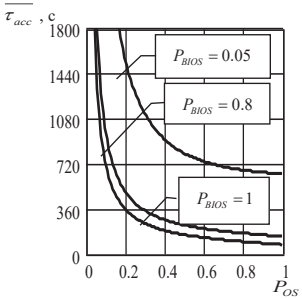
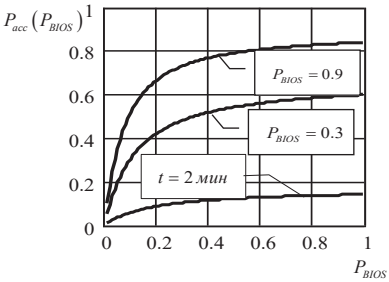
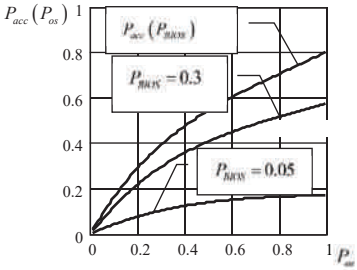


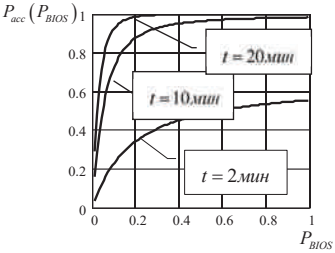
Рисунок 5.21 – Зависимость среднего времени реализации угрозы доступа от вероятности подбора пароля доступа в операционную систему



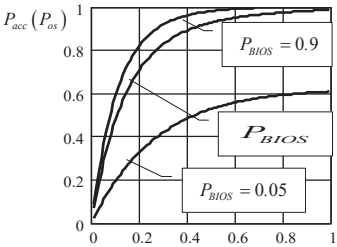
а) $P_{os} = 0.1$, $\tau_{BIOS} = 40c$, $\tau_{os} = 30c$



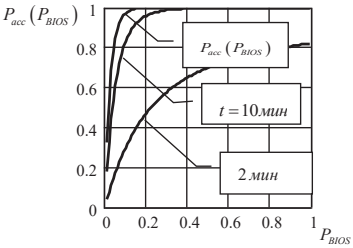
г) $t = 2$ мин, $\tau_{BIOS} = 40c$, $\tau_{os} = 30c$



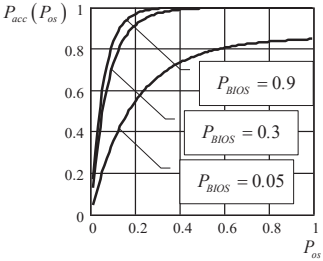
б) $P_{os} = 0.5$, $\tau_{BIOS} = 40c$, $\tau_{os} = 30c$



д) $t = 10$ мин, $\tau_{BIOS} = 40c$, $\tau_{os} = 30c$



в) $P_{os} = 0.5$, $\tau_{BIOS} = 40c$, $\tau_{os} = 30c$



е) $t = 20$ мин, $\tau_{BIOS} = 40c$, $\tau_{os} = 30c$

Рисунок 5.22 – Зависимости вероятностей реализации угрозы доступа от вероятности подбора пароля доступа в BIOS и в операционную систему

5.6.3. Марковские модели динамики реализации некоторых сетевых атак

Применение марковских моделей для анализа динамики реализации сетевых атак в значительной мере ограничено невозможностью моделирования параллельных процессов, которые существенны для некоторых атак. В этом отношении более адекватно отражают динамику атак модели на основе аппарата составных сетей Петри-Маркова, в основе которых

лежит теория полумарковских процессов. Эти модели рассматриваются в следующем разделе.

Вместе с тем некоторые из атак вполне адекватно могут быть описаны во времени и с помощью марковских процессов. В данной работе рассмотрены марковские модели, во-первых, наиболее распространенных сетевых атак, а, во-вторых, тех, которые позволяют отразить суть подходов к их моделированию:

– атака типа «Удаленный доступ» в операционную среду компьютера по протоколу межсетевого взаимодействия;

– типовой сетевой атаки «Отказ в обслуживании»;

– атаки, направленной на внедрении программных закладок;

– атаки типа «Удаленный запуск приложений».

Изложенные при этом подходы могут быть распространены и на другие сетевые атаки, которые не включают в себя параллельно выполняемые во времени процедуры.

Марковская модель динамики реализации сетевой атаки типа «Удаленный доступ в операционную среду компьютера». Удаленный доступ в операционную среду компьютера может осуществляться по многим протоколам межсетевого взаимодействия прикладного уровня, таким как Telnet, FTP, TFTP и др.

Как правило, при этом нарушителю необходимо пройти процедуру аутентификации по паролю. Подбор пароля осуществляется преимущественно с помощью специально созданных для этого программ. Модель динамики такой атаки сходна с моделью физического доступа нарушителя к аппаратуре (в помещение). При этом вероятность реализации такой атаки за заданное время при экспоненциальном приближении и отсутствии ограничений на количество попыток доступа оценивается по простой формуле:

$$P_u(t) = 1 - \exp\left[-t \Big/ \left(\frac{\overline{\tau_{pass}}}{P_{pass}} + \overline{\tau_{acc}} \right) \right], \qquad (5.101)$$

где $\overline{\tau_{pass}}$ – среднее время подбора пароля;

P_{pass} – вероятность успешной попытки подбора пароля.

Если установлено ограничение на количество попыток доступа, то модель значительно усложняется. В этом случае время доступа оценивается только для ограниченного числа попыток I и представляет собой сумму:

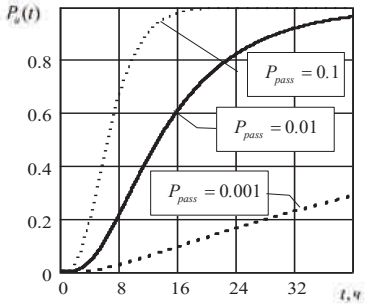
$$\tau_I = \sum_{i=1}^{I} (\tau_{acc.i} + \frac{\tau_{pass}}{P_{pass}}) . \qquad (5.102)$$

Вероятность того, что за время t нарушитель получит доступ, находится из соотношения:

$$P_u(t) = \int_0^t \gamma(\lambda_{acc} \cdot (t-x), I) \cdot \lambda_{pass} \cdot P_{pass} \cdot e^{-\lambda_{pass} \cdot P_{pass} \cdot x} \cdot dx , \qquad (5.103)$$

где $\gamma(y,i)$ табулированная неполная гамма-функция, рассчитываемая для целочисленных значений параметра i (см. раздел 5.5).

График, построенный по полученной формуле, приведен на рисунке 5.23.



$\bar{\tau}_{acc} = 2\frac{1}{ч},\ \bar{\tau}_{pass} = 5мин,\ I = 3$

Рисунок 5.23 – Зависимость вероятности удаленного доступа от времени и вероятности подбора пароля для выбранных временных характеристик динамики попыток доступа

Марковские модели динамики реализации сетевых атак типа «Отказ в обслуживании». В настоящее время может быть реализовано достаточно много различных атак, относящихся к атакам «Отказ в обслуживании». В частности, такие атаки могут быть основаны на IP–дефрагментации, на формировании некорректных ICMP–запросов, на формировании некорректных TCP–запросов, на создании «шторма» пакетов с запросами на соединение и др. В научной литературе отсутствуют математические модели динамики реализации таких атак. В основу таких моделей могут быть положены теория марковских процессов и теория массового обслуживания. При общей методологии построения таких моделей все-таки каждая из них имеет определенную специфику.

К наиболее характерным из них относятся модели атак «шторм TCP–запросов» (SYN–flooding), «шторм ICMP-эхо-ответов» (атака Smurf), атаки типа «Land». Краткая характеристика таких моделей приведена ниже.

Рассмотрим сначала модель динамики реализации самой известной из таких атак – «шторм» TCP–запросов (SYN–flooding). Суть атаки заключается в том, что на атакуемый хост передается большое количество запросов на установление «полуоткрытого» виртуального соединения по выбранному порту. В результате операционная система, ожидая подтверждения (квитанции) о готовности абонента к связи, держит в буфере информацию о предполагаемом соединении, повторяя посылку сообщений – квитанций о

получении запроса на соединение. Если таких запросов поступило много, то буфер оказывается полностью заполненным и система перестает принимать другие запросы по данному порту (некоторые операционные системы старых версий «зависают» и нуждаются в перезагрузке [99]). Динамика реализации такой атаки моделируется в виде системы массового обслуживания с очередью фиксированной длины при следующих условиях: обслуживающий прибор при поступлении рассматриваемых заявок отказывает и может быть восстановлен сразу по мере прохождении фиксированного промежутка времени после переполнения очереди.

Вероятность того, что операционная система к моменту времени t перестанет принимать новые пакеты информации, определится из соотношения:

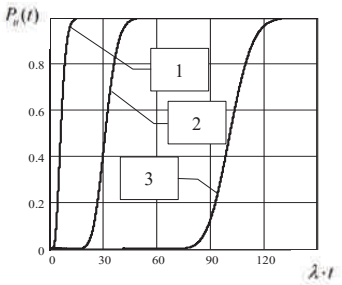
$$P_u(t) = 1 - \sum_{j=0}^{J_{\lim}-1} \frac{(\lambda_{syn} \cdot t)^j}{j!} \cdot e^{-\lambda_{syn} \cdot t} \ \ при \ t \le \tau_{\lim}, \qquad (5.104)$$

где λ_{syn} – интенсивность поступления пакетов с запросами на соединение (пакетов с установленным флагом SYN);

$J_{\lim}$ – предельное количество возможных соединений по одному порту, при котором буфер оказывается заполненным;

$\tau_{\lim}$ – время, отводимое операционной системой на повторную передачу квитанций с флагами SYN и ACK и время ожидания ответа.

График зависимости (5.105) приведен на рисунках 5.24 и 5.25.

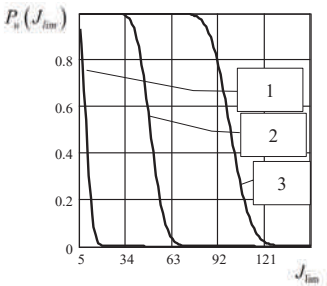


при условии $\tau_{\lim} > t$

$1 - J_{\lim} = 5,\ 2 - J_{\lim} = 30,$

$3 - J_{\lim} = 100$

Рисунок 5.24 – Зависимость вероятности «зависания» операционной системы от параметров атаки SYN–flooding



при условии $\tau_{\lim} > t$

$1 - \lambda_{syn} \cdot t = 10,\ \ 2 - \lambda_{syn} \cdot t = 50,$

$2 - \lambda_{syn} \cdot t = 100$

Рисунок 5.25 – Зависимость вероятности «зависания» операционной системы от предельного количества соединений по порту при атаке SYN-flooding

Математическое ожидание времени, необходимого для реализации атаки, составляет величину:

$$\overline{\tau_u} = J_{lim} / \lambda_{syn} . \qquad (5.105)$$

Если приходит следующий $(J_{lim}+1)$-й пакет с запросом на соединение, то этот пакет отбрасывается. Если атака обнаруживается до этого момента времени с некоторой вероятностью P_{det} и ее развитие блокируется, то вероятность реализации атаки может быть рассчитана по формуле:

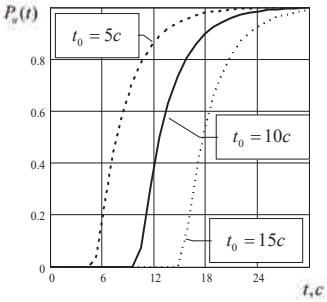
$$P_u(t) = 1 - \frac{\lambda_{syn} \cdot \overline{\tau_u} \cdot e^{-\frac{(t-t_0)\cdot(1-P_{det})}{\overline{\tau_u}}}}{\lambda_{syn} \cdot \overline{\tau_u} - (1-P_{det})} + \frac{(1-P_{det}) \cdot e^{-\lambda_{syn}\cdot(t-t_0)}}{1-\lambda_{syn} \cdot \overline{\tau_u} \cdot (1-P_{det})}, \qquad (5.106)$$

где t_0 – время ожидания подтверждения сеанса связи.

Графики, построенные по полученной формуле, приведены на рисунке 5.26.

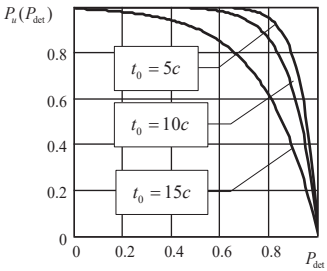
Рассмотрим модель динамики реализации атаки – шторм ICMP-«эхо-ответов» (Smurf) [26, 100, 101]. Суть атаки заключается в посылке «эхо-запроса» по протоколу ICMP по широковещательному адресу с указанием в качестве адреса отправителя IP-адреса компьютера – цели атаки, ответить на которые может множество компьютеров.

В результате на вход компьютера с указанным IP-адресом поступает поток пакетов (ICMP-«эхо-ответов»), который операционная система не в состоянии быстро обработать, что приводит к тому, что данный компьютер фактически блокируется.



$\lambda_{syn} = 1\frac{1}{c}, \bar{\tau}_{serv} = 0.01c, J_{\lim} = 30$

а)



$t = 20c$, $\lambda_{syn} = 1\frac{1}{c}, \bar{\tau}_{serv} = 0.01c, J_{\lim} = 30$

б)

Рисунок 5.26 – Зависимость вероятности реализации атаки от возможностей ее обнаружения и времени ожидания квитанции на подтверждение сеанса связи

Динамика реализации такой атаки моделируется с помощью аппарата теории массового обслуживания следующим образом. Пусть отсутствует

эффект перегрузки канала приема «лавины пакетов» по сети (из-за ограниченной пропускной способности сети), тогда «лавина пакетов» приведет к перегрузке работы операционной системы: если операционная система занята обработкой очередного пакета, то вновь приходящий пакет сохраняется в буфере после обработки сетевым адаптером. При переполнении буфера операционная система блокирует прием очередных пакетов. С учетом изложенного операционную систему компьютера можно рассматривать как систему массового обслуживания с ограниченной очередью и потерями в случае превышения длиной очереди определенного порога.

Пусть N_{buf} – максимальное число пакетов, которые может быть размещено в буфере, и $\overline{\tau_{serv}}$ – среднее время обработки каждого пакета, при этом выполняются условия марковости рассматриваемого процесса [83].

Если в систему поступает большое количество пакетов, то она не успевает их обрабатывать и блокируется для очистки буфера на время, составляющее несколько десятков секунд. С учетом [87] вероятность того, что в системе к моменту времени поступит ровно N_{buf} пакетов, может быть рассчитана по формуле:

$$P_{proh}(t) = \rho^{N_{buf}} \cdot \frac{(\rho - 1)}{\rho^{N_{buf}+1} - 1} - \frac{2 \cdot \rho^{\frac{N_{buf}+1}{2}}}{N_{buf}+1} \cdot \sum_{k=1}^{N_{buf}} \frac{a_{0k} \cdot a_{N_{buf}k}}{a_k} \cdot e^{-\frac{a_k \cdot t}{\overline{\tau_{serv}}}}, \qquad (5.107)$$

где ρ – параметр загрузки системы потоком пакетов,

$\rho = \lambda_H \cdot \overline{\tau}_{serv}$;

λ_H – интенсивность потоа пакетов;

a_k, a_{0k}, a_{jk} – вспомогательные параметры,

$$a_k = 1 + \rho - 2 \cdot \sqrt{\rho} \cdot cos\frac{k \cdot \pi}{N_{buf}+1};\ a_{0k} = sin\frac{k \cdot \pi}{N_{buf}+1};\ a_{N_{buf}k} = sin\frac{N_{buf} \cdot k \cdot \pi}{N_{buf}+1}.$$

Из соотношения (5.107) видно, что при $\rho > 1$ и при больших значениях N_{buf} вероятность блокирования операционной системы близка к единице. Графики, построенные по полученной зависимости, приведены на рисунках 5.27 и 5.28.

Пусть атакуемый хост до атаки не имеет в буфере ни одного пакета, при этом среднее время, необходимое для блокирования операционной системы, определяется из соотношения:

$$\overline{\tau_{proh}} = \frac{2 \cdot \overline{\tau_{serv}} \cdot \rho^{\frac{N_{buf}+1}{2}}}{N_{buf}+1} \cdot \sum_{k=1}^{N_{buf}} \frac{a_{0k} \cdot a_{kN}}{a_k^{\ 2}}. \qquad (5.108)$$

Графики зависимости $\overline{\tau_{proh}}(N_{buf})$ при различных значениях загрузки операционной системы и $\overline{\tau_{proh}}(\rho)$ при разных предельных значениях количества принимаемых пакетов приведены на рисунках 5.29 и 5.30.

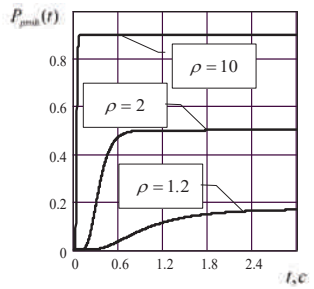


Рисунок 5.27 – Зависимость вероятности заполнения буфера максимальным количеством пакетов при атаке Smurf от времени

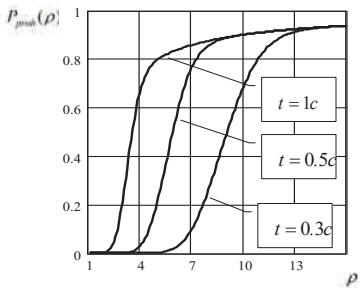


Рисунок 5.28 – Зависимость вероятности заполнения буфера максимальным количеством пакетов при атаке Smurf от загрузки системы

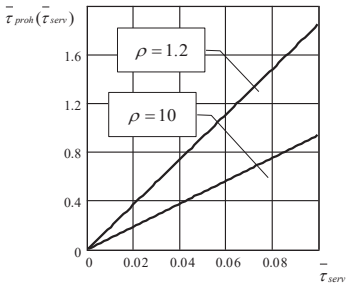


Рисунок 5.29 – Зависимость среднего времени заполнения буфера от среднего времени обработки каждого пакета

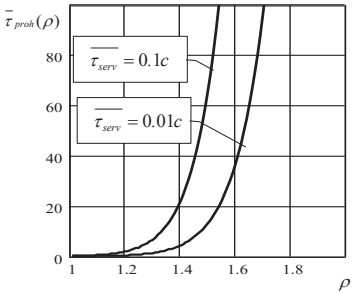


Рисунок 5.30 – Зависимость среднего времени заполнения буфера от загрузки системы

При поступлении $\left(N_{buf}+1\right)$-го пакета система блокируется. При этом вероятность реализации атаки за заданное время может быть рассчитана по формуле:

$$P_u\left(t\right)=1-e^{-\frac{t}{\bar{\tau}_{proh}}}. \qquad (5.109)$$

Если на компьютере установлена система обнаружения атак, которая обнаруживает атаку типа Smurf с вероятностью $P_{\det}$, то

$$P_u\left(t\right)=1-e^{-\frac{t\cdot\left(1-P_{det}\right)}{\bar{\tau}_{proh}}}. \qquad (5.110)$$

Графики, полученные по формулам (5.109) и (5.110), приведены на рисунках 5.31 и 5.32 соответственно.

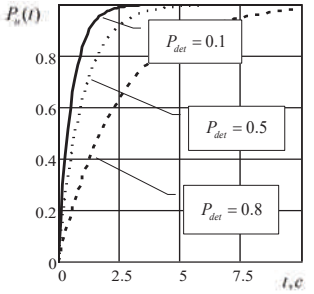


Рисунок 5.31. Зависимость вероятности реализации атаки Smurf от времени

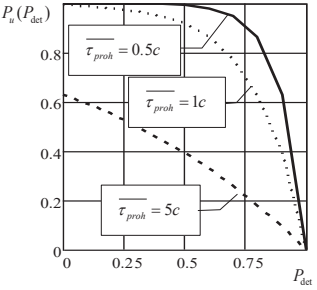


Рисунок 5.32. Зависимость вероятности реализации атаки Smurf от вероятности ее обнаружения при $t = 1c$

Примерно так же развиваются атаки:

– основанные на IP-дефрагментации, когда общий объем фрагментов превышает максимально допустимый размер пакета, при этом система «зависает» или даже выходит из строя;

– «направленный шторм эхо-запросов» по протоколу ICMP (Ping flooding);

– шторм запросов к FTP-серверу;

– шторм сообщений почтовому серверу (Spam).

Несколько отличается от приведенного динамика развития атак, направленных на нарушение логической связности между объектами компьютерной сети при передаче атакующим управляющих сообщений от имени сетевых устройств, приводящих к изменению маршрутно-адресных данных.

К таким атакам относятся, например, ICMP Redirect Host, DNS-flooding или атаки, связанные с передачей нарушителем пакетов с нестандартными атрибутами (атаки типа «Land», «TearDrop», «Bonk», «Nuke», «UDP- bomb» [33, 35, 99–101]), а также с длиной пакета, превышающей максимально допустимый его размер (атака типа «Ping Death», атака демона ftpd UNIX-хоста).

Рассмотрим модели развития подобных атак на примере атаки «Land!», суть которой состоит в том, что на атакуемый хост посылается TCP-пакет, у которого IP-адрес и порт отправителя совпадают с IP-адресом и портом получателя.

В результате обработки такого пакета операционная система «виснет», а динамика атаки будет определяться моментом появления такого пакета.

При экспоненциальном характере процесса возникновения угроз такой атаки вероятность того, что за время t в системе появится нестандартный пакет и этот пакет не будет выявлен и отброшен операционной системой (то есть атака будет реализована), определяется из соотношения, аналогичного соотношению (5.109).

Необходимо отметить, что многие их указанных атак сегодня успешно парируются мерами и средствами операционных систем, в которые постоянно вводятся дополнения и усовершенствования. Так, весьма сложно реализовать атаку типа Smurf, поскольку блокируется обработка сверх установленного количества пакетов в единицу времени и тем самым снижается эффект от атаки типа «Land» и ей подобных. Однако, появляются новые способы воздействия, но при этом для формализованного описания динамики реализации новых атак могут быть, как правило, использованы предложенные математические модели.

Марковская модель динамики реализации сетевых атак, основанных на внедрении программных закладок. Особенность несанкционированного доступа, основанного *на внедрении программных закладок,* заключается в том, что первоначально проводится внедрение программной закладки – вредоносной программы, например, типа «троянский конь». Внедрение может осуществляться разными методами. Например, часто используется метод подмены IP-адреса на IP-адрес обычного web-прокси-сервера, который выступает в роли промежуточного звена между web-пользователем и web-сервером (все операции обмена данными между клиентом и web-сервером направляются через прокси-сервер) [100–103]. Возможно заражение через почтовый сервер. С помощью программной закладки осуществляется сканирование сети с целью выявления открытых портов, составления карты сети, осуществления попытки связи с зараженного компьютера с выдачей информации о сети на тот или иной IP-адрес. Кроме того, возможен подбор или перехват пароля, передача файла с паролями и др. Рассмотрим вариант атаки, когда вредоносная программа осуществляет поиск открытых портов, затем перехватывает пароль и устанавливает соединение с абонентом по IP-адресу, в ходе которого нарушитель захватывает сеанс TCP- соединения и получает доступ в операционную среду хоста-цели атаки. Граф состояний описанного процесса развития сетевой атаки приведен на рисунке 5.33.

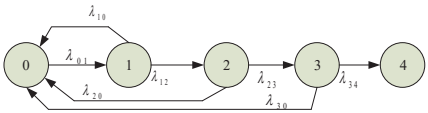
Интенсивности переходов из состояния в состояние для данного марковского процесса определяются следующим образом:

$\lambda_{01} = P_{inc} / \overline{\tau_{inc}}$ – интенсивность внедрения программной закладки, где $P_{inc}, \overline{\tau_{inc}}$ – вероятность и среднее время внедрения программной закладки;

$\lambda_{12} = P_{port} / \overline{\tau_{port}}$ – интенсивность выявления открытого порта, где $P_{port}, \overline{\tau_{port}}$ – вероятность и среднее время выявления открытого порта;

$\lambda_{23} = P_{pass} / \overline{\tau_{pass}}$ – интенсивность выявления пароля, где $P_{pass}, \overline{\tau_{pass}}$ – вероятность и среднее время выявления пароля;

$\lambda_{34} = P_{con} / \overline{\tau_{con}}$ – интенсивность установления сеанса связи между программной закладкой и нарушителем, где $P_{con}, \overline{\tau_{con}}$ – вероятность и среднее время установления сеанса связи;



0 – состояние, когда нарушитель пытается внедрить программную закладку;

1 – состояние, когда программная закладка внедрена и осуществляется анализ сети (выявление открытых портов);

2 – состояние, когда выявлен открытый порт и программная закладка осуществляет перехват пароля доступа;

3 – состояние, когда пароль доступа перехвачен и программная закладка запрашивает сеанс связи от имени хоста-цели атаки с передачей выявленной информации нарушителю;

4 – состояние, когда сеанс связи с нарушителем установлен, то есть нарушитель получил доступ в операционную среду компьютера по сети с использованием выявленного порта

$$\begin{cases} P_0'(t) = -\lambda_{01} \cdot P_0(t) + \lambda_{34} \cdot P_3(t) + \lambda_{10} \cdot P_1(t) + \lambda_{20} \cdot P_2(t) + \lambda_{30} \cdot P_3(t); \\ P_1'(t) = \lambda_{01} \cdot P_0(t) - \lambda_{12} \cdot P_1(t) - \lambda_{10} \cdot P_1(t); \\ P_2'(t) = \lambda_{12} \cdot P_1(t) - \lambda_{20} \cdot P_2(t) - \lambda_{23} \cdot P_2(t); \\ P_3'(t) = \lambda_{23} \cdot P_2(t) - \lambda_{30} \cdot P_3(t) - \lambda_{34} \cdot P_3(t); \\ P_4'(t) = \lambda_{34} \cdot P_3(t), \end{cases}$$

при начальных условиях $P_0(0) = 1; P_1(0) = P_2(0) = P_3(0) = P_4(0), P_0'(0) = P_1'(0) = P_2'(0) = P_3'(0) = P_4'(0) = 0$

Рисунок 5.33 – Граф состояний, описывающий динамику развития атаки, основанной на внедрении программной закладки

$\lambda_{01} = \dfrac{1 - P_{inc}}{\overline{\tau_{inc}}}, \lambda_{12} = \dfrac{1 - P_{port}}{\overline{\tau_{port}}}, \lambda_{23} = \dfrac{1 - P_{pass}}{\overline{\tau_{pass}}}, \lambda_{34} = \dfrac{1 - P_{con}}{\overline{\tau_{con}}}$ – интенсивности перехода процесса в начальное состояние в случае невыявления свободного порта, пароля и неустановления сеанса связи соответственно.

При этом вероятность подбора пароля за n попыток при прямом переборе, по-прежнему, рассчитывается из соотношения (5.99). Если учитывать динамику подбора пароля во времени при условии, что количество попыток подбора пароля не ограничено и поток попыток соответствует пуассоновскому, то вероятность подбора за время τ определяется по формуле:

$$P_{pass}(\tau) = 1 - e^{-\frac{\lambda_{tr} \cdot \tau}{A^s}}. \qquad (5.111)$$

где λ_{tr} – интенсивность попыток удаленного подбора пароля;

A – число символов в алфавите, на основе которого сформирован пароль;

s – длина пароля.

Среднее время, необходимое для подбора пароля, находится следующим образом [98]:

$$\overline{\tau_{pass}} = \frac{A}{\lambda_{tr}}. \qquad (5.112)$$

При ограничении на количество попыток необходимо ввести условие $\lambda_{tr} \cdot \tau < n_{\lim}$, где $n_{\lim}$ – предельное количество попыток.

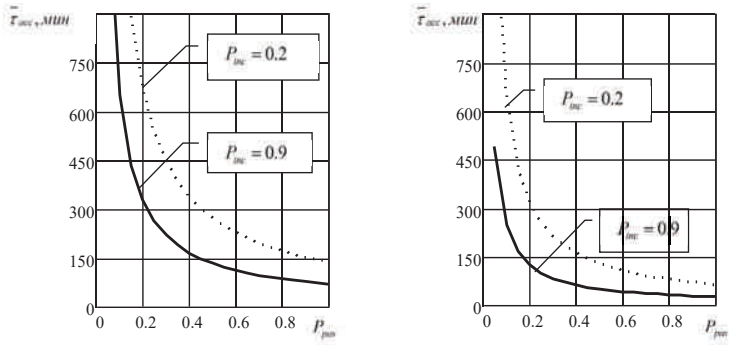
Для приведенного графа состояний процесса среднее время перехода в последнее состояние, то есть среднее время несанкционированного доступа определяется из соотношения:

$$\overline{\tau_{acc}} = \frac{P_{inc} \cdot \overline{\tau_{port}} + \overline{\tau_{inc}}}{P_{inc} \cdot P_{port} \cdot P_{pass} \cdot P_{con}} + \frac{P_{pass} \cdot \overline{\tau_{con}} + \overline{\tau_{pass}}}{P_{pass} \cdot P_{con}}, \qquad (5.113)$$

а вероятность несанкционированного доступа за время t определяется из соотношения:

$$P_g(t) = 1 - \exp\left(-t/\overline{\tau_{acc}}\right). \qquad (5.114)$$

Графики зависимости $\overline{\tau_{acc}}\left(P_{pass}\right)$ представлены на рисунке 5.34.



$\overline{\tau}_{port} = 1мин$, $\overline{\tau}_{inc} = 10мин, \overline{\tau}_{con} = 3мин,$
$\overline{\tau}_{pass} = 30с, P_{port} = 0,8,\ P_{con} = 0.7$

$\overline{\tau}_{port} = 10мин$, $\overline{\tau}_{inc} = 30мин, \overline{\tau}_{con} = 5мин,$
$\overline{\tau}_{pass} = 30с, P_{port} = 1,\ P_{con} = 1$

Рисунок 5.34 – Зависимости времени несанкционированного удаленного доступа к информации от вероятности подбора пароля доступа абонента

Необходимо отметить, в приведенной модели фактически не учитывается время подбора пароля, то есть считается, что нарушитель имеет достаточно времени для удаленного подбора пароля.

5.6.4. Марковские модели динамики инфицирования вредоносными программами информационных систем

Особенностью реализации угроз применения вредоносных программ в ИС является существенная зависимость опасности угрозы от времени. Возрастание опасности угрозы может представлять собой лавинообразный процесс. В этих условиях особенно важным становится оценка возможной динамики развития указанного процесса, то есть динамики «заражения» («инфицирования») вредоносной программой файлов в файловой системе каждого компьютера, совокупности компьютеров в компьютерной сети, множества не объединенных в сети компьютеров компьютерным вирусом.

Динамика реализации таких угроз определяется целым рядом факторов, таких как класс вредоносной программы, способ внедрения и распространения, содержание несанкционированного действия, ассоциация с другими программами, время инфицирования, интенсивность запуска инфицированных файлов и др.

Разнообразие этих факторов обусловливает то, что пока не удается разработать какую-либо одну (единую) математическую модель динамики распространения всех вредоносных программ.

До настоящего времени наиболее глубоко рассмотрены марковские модели распространения вредоносных программ в [104, 35]. Так, в [104] предложены марковские модели распространения резидентных и нерезидентных компьютерных вирусов в компьютере и компьютерных сетях. Однако эти модели, во-первых, не позволяют рассчитать вероятность того, что за заданное время может быть поражен конкретный защищаемый блок информации, то есть оценивать опасность угроз выполнения вредоносных программ применительно к блокам защищаемой информации. Во-вторых, в них не учитываются меры и средства защиты от вредоносных программ. Вместе с тем заложенные в эти модели идеи весьма плодотворны и были использованы в данной работе. При этом были рассмотрены пять основных моделей динамики распространения вредоносных программ:

– загрузочного вируса без поиска и с поиском поражаемого файла в пределах файловой системы одного компьютера с выполнением разового деструктивного действия;

– файлового вируса или загрузочного вируса с многократным последовательным заражением файлов с определенными расширениями имен и отсутствии ограничения на количество поражаемых файлов[23] в пределах файловой системы одного компьютера

– файлового вируса или загрузочного вируса с многократным последовательным заражением файлов с определенными расширениями имен и наличии ограничения на количество восприимчивых к заражению файлов в пределах файловой системы одного компьютера;

– файлового или загрузочного при распространении среди множества компьютеров, не объединенных в ИС;

– файлового, загрузочного вируса или сетевого «червя» при распространении по ИС.

Основные полученные соотношения для оценки вероятностно-временных характеристик реализации угроз применения вредоносных программ применительно к указанным моделям приведены в таблице 5.4. Приведенные соотношения позволяют рассчитать вероятностно-временные характеристики динамики такого распространения.

[23] Ограничение в распространении связано только с количеством файлов в файловой системе, которые могут быть поражены данным вирусом, но не характеристиками самой вредоносной программы.

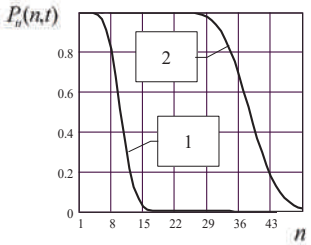
Таблица 5.4 – Вероятностно временные характеристики динамики реализации угроз применения вредоносных программ

Наименование модели	Условия реализации угрозы	Основные соотношения	Принятые обозначения
Модель динамики заражения компьютера загрузочным вирусом, выполняющим какое-либо разовое деструктивное действие	Заражение происходит при использовании зараженного отчуждаемого носителя – дискеты или CD-диска. Деструктивное действие осуществляется без поиска поражаемой информации (например, форматирование жесткого диска)	$P_{ti}(t)=1-\frac{\overline{\tau}_{th}^{2}\cdot e^{-\frac{t}{\overline{\tau}_{th}}}\cdot\left(1-P_{det}^{(g)}\right)\cdot\left(1-P_{det}^{(inf)}\right)}{\left[\overline{\tau}_{th}\cdot\left(1-P_{det}^{(inf)}\right)-\overline{\tau}_{inf}\right]\cdot\left[\overline{\tau}_{th}\cdot\left(1-P_{det}^{(g)}\right)-\overline{\tau}_{g}\right]}-$ $-\frac{\overline{\tau}_{inf}^{2}\cdot e^{-\frac{t\cdot\left(1-P_{det}^{(inf)}\right)}{\overline{\tau}_{inf}}}\cdot\left(1-P_{det}^{(g)}\right)}{\left[\overline{\tau}_{inf}-\overline{\tau}_{th}\cdot\left(1-P_{det}^{(inf)}\right)\right]\cdot\left[\overline{\tau}_{inf}\cdot\left(1-P_{det}^{(g)}\right)-\overline{\tau}_{g}\cdot\left(1-P_{det}^{(inf)}\right)\right]}-$ $-\frac{\overline{\tau}_{g}^{2}\cdot e^{-\frac{t\cdot\left(1-P_{det}^{(inf)}\right)}{\overline{\tau}_{g}}}\cdot\left(1-P_{det}^{(inf)}\right)}{\left[\overline{\tau}_{g}-\overline{\tau}_{th}\cdot\left(1-P_{det}^{(g)}\right)\right]\cdot\left[\overline{\tau}_{g}\cdot\left(1-P_{det}^{(inf)}\right)-\overline{\tau}_{inf}\cdot\left(1-P_{det}^{(g)}\right)\right]};$ $\overline{\tau}_{th}=\frac{1}{P_{inf}\left(\overline{\mu}_{out}+\overline{\mu}_{bis}\right)}$	$\overline{\tau}_{th}$ – среднее время между инцидентами возникновения угроз заражения вирусом; $\overline{\tau}_{inf}$ – среднее время, необходимое для заражения компьютера данным вирусом (вредоносной программой) с момента возникновения инцидента; $\overline{\tau}_{g}$ – среднее время выполнения деструктивного действия; $P_{det}^{(inf)}$ – вероятность обнаружения вредоносной программы и своевременного пресечения ее распространения на этапе инфицирования; $P_{det}^{(g)}$ – вероятность обнаружения вредоносной программы при попытке выполнения деструктивного действия и своевременного его пресечения; P_{inf} – вероятность инфицирования отчуждаемого носителя при применении в зараженном компьютере; $\overline{\mu}_{out}$ - средняя интенсивность применения пользователем чужих дискет (или CD-дисков); $\overline{\mu}_{bis}$ – средняя интенсивность использования своих дискет пользователем для записи информации на чужом компьютере
	Заражение происходит при использовании зараженного отчуждаемого носителя – дискеты или CD-диска. Деструктивное действие осуществляется с поиском поражаемой информации (например, по расширению имени файла)	$P_{ti}(t)=1-\frac{\overline{\tau}_{th}^{3}\cdot e^{-\frac{t}{\overline{\tau}_{th}}}\cdot(1-P_{det}^{(inf)})(1-P_{det}^{(g)})(1-P_{det}^{(s)})}{\{\overline{\tau}_{th}\cdot(1-P_{det}^{(inf)})-\overline{\tau}_{inf}\}\{\overline{\tau}_{th}\cdot(1-P_{det}^{(g)})-\overline{\tau}_{g}\}\{\overline{\tau}_{th}(1-P_{det}^{(s)})-\overline{\tau}_{s}\}}-$ $-\frac{\overline{\tau}_{inf}^{3}\cdot e^{-\frac{t}{\overline{\tau}_{inf}}}\cdot(1-P_{det}^{(s)})}{\{\overline{\tau}_{inf}-\overline{\tau}_{th}(1-P_{det}^{(inf)})^{2}\}\{\overline{\tau}_{inf}-\overline{\tau}_{g}(1-P_{det}^{(inf)})^{2}\}\{\overline{\tau}_{inf}\cdot(1-P_{det}^{(s)})-\overline{\tau}_{s}(1-P_{det}^{(inf)})^{2}\}}-$ $-\frac{\overline{\tau}_{g}^{3}\cdot e^{-\frac{t}{\overline{\tau}_{g}}}\cdot(1-P_{det}^{(s)})\cdot(1-P_{det}^{(inf)})}{\{\overline{\tau}_{g}-\overline{\tau}_{th}(1-P_{det}^{(inf)})\}\{\overline{\tau}_{g}(1-P_{det}^{(inf)})-\overline{\tau}_{inf}(1-P_{det}^{(g)})\}\{\overline{\tau}_{g}(1-P_{det}^{(s)})-\overline{\tau}_{s}(1-P_{det}^{(g)})\}}-$ $-\frac{\overline{\tau}_{s}^{3}\cdot e^{-\frac{t}{\overline{\tau}_{s}}}}{\{\overline{\tau}_{s}-\overline{\tau}_{th}(1-P_{det}^{(s)})\}\{\overline{\tau}_{s}(1-P_{det}^{(inf)})-\overline{\tau}_{inf}(1-P_{det}^{(s)})\}\{\overline{\tau}_{s}-\overline{\tau}_{g}(1-P_{det}^{(s)})\}};$ $\overline{\tau}_{th}=\frac{1}{P_{inf}\left(\overline{\mu}_{out}+\overline{\mu}_{bis}\right)}$	Те же, а также: $P_{det}^{(s)}$ – вероятность обнаружения вредоносной программы при поиске блока информации, подлежащего поражению; $\overline{\tau}_{s}$ – среднее время поиска блока информации (файла) для выполнения предусмотренного во вредоносной программе действия

Наименование модели	Условия реализации угрозы	Основные соотношения	Принятые обозначения
Модель распространения файловых вирусов в файловой системе при отсутствии ограничения на количество восприимчивых к заражению файлов и автозапуске вредоносной программы после каждого инфицирования	При попадании вредоносной программы (например, через флоппи-диск) в файловую систему дальнейшее инфицирование происходит путем последовательного автозапуска программы после каждого инфицирования очередного файла с некоторым расширением имени (например, исполняемых файлов с расширением .COM или .EXE.)	$t_{dkf} = t_{rec} + n \cdot (t_{ret} + t_{inf})$; $P_n(t,n) = L^{-1}\{\frac{1}{s\cdot(1+s\cdot\tau_1)}\cdot\frac{1}{[(1+s\cdot\tau_2)(1+s\cdot\tau_3)]^n}\}$; $P_n(t,n) = \frac{\sqrt{\pi}}{(n-1)!(\tau_2\cdot\tau_3)^n}\cdot\int_0^t dy\int_0^y \frac{e^{-\frac{y-z}{\tau_1}}}{\tau_1}\cdot(\frac{x}{2a})^{n-\frac{1}{2}}\cdot e^{-bz}\cdot I_{n-\frac{1}{2}}(ax)dx$ $a = \frac{\tau_2-\tau_3}{2\cdot\tau_2\cdot\tau_3}$, $b = \frac{\tau_2+\tau_3}{2\cdot\tau_2\cdot\tau_3}$; $\tau_1 = \frac{\bar{\tau}_{rec}}{1-P_{det}^{(rec)}}$; $\bar{\tau}_2 = \frac{\bar{\tau}_{ret}}{1-P_{det}^{(ret)}}$; $\tau_3 = \frac{\bar{\tau}_{inf}}{1-P_{det}^{(inf)}}$; $P_n(n,l,t) = \sum_{k=l}^{n}\frac{C_{m_{inf}}^{l}\cdot C_{n-m_{inf}}^{k-l}}{C_n^k}\cdot P_n(k,t)$	$t_{rec}, t_{ret}, t_{inf}$ – времена, необходимые соответственно для того, чтобы вредоносная программа записалась, осуществился успешно поиск подлежащего поражению файла и чтобы файл был поражен (выполнено деструктивное действие). a, b - вспомогательные параметры; $\bar{\tau}_{rec}, \bar{\tau}_{ret}, \bar{\tau}_{inf}$ – средние значения величин $t_{rec}, t_{ret}, t_{inf}$. соответственно; $P_n(n,l,t)$ - вероятность того, что за время t только l файлов из n пораженных относятся к защищаемым; L^{-1} - обратное преобразование Лапласа с параметров s; $P_{det}^{rec}, P_{det}^{ret}, P_{det}^{inf}$ - вероятности обнаружения вредоносной программы при ее записи, при поиска подлежащего поражению файла и при попытке выполнения деструктивного действия
Модель динамики распространения нерезидентной вредоносной программы в случае, когда для ее распространения необходимо запустить на выполнение какой-либо из зараженных файлов и количество восприимчивых к заражению файлов ограничено [104]	Распространение вредоносной программы происходит по мере заражения файлов в текущем каталоге. Интенсивность заражения зависит от интенсивности использования файлов-вирусоносителей и от количества файлов, подверженных заражению. В начале процесса поражено некоторое количество файлов	$P_n(t,n) = \begin{cases} \frac{(n-l)!}{(m+h)\times(h-l)!}\times\sum_{j=h}^{m} \frac{e^{-\frac{j}{m+h}m_{inf}(1-P_{det})t}}{\prod_{\substack{k=h\\k\neq j}}^{m}(k-j)} & \text{для } n=\overline{h+1, h+m-1}; \\ \frac{(m+h-1)!}{(m+h)^m\times(h-1)!}\times[\sum_{j=h}^{m+h-1}(-\frac{j\times\prod_{\substack{k=h\\k\neq j}}^{m}\frac{m+h}{k-j}}{m+h}\times e^{-\frac{j}{m+h}m_{inf}(1-P_{det})t})+ \\ +\prod_{j=h}^{m+h-1}\frac{m+h}{j}] & \text{для } n=m+h; \\ \frac{(n-l)!}{(m+h)^{m-h}\times(h-l)!}\times\sum_{j=h}^{n} e^{-\frac{j}{m+h}m_{inf}(1-P_{det})t} & \text{для } n=h; \end{cases}$ $\mu_{inf} = \frac{\lambda_{inf}\cdot n}{m+h}$	μ_{inf} – интенсивность инфицирования файлов; P_{det} – вероятность того, что вирус обнаруживается антивирусной программой при каждом запуске одного из пораженных файлов, и пресекается его распространение; n - количество зараженных файлов в каталоге за время t; m - количество файлов в каталоге, восприимчивых к заражению; h - количество файлов в каталоге, зараженных к началу рассматриваемого процесса распространения вредоносной программы; p_{det} - вероятность обнаружения вредоносной программы в файловой системе

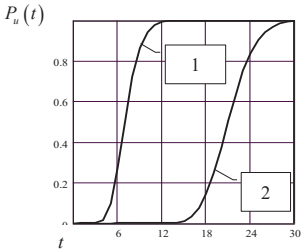
Наименование модели	Условия реализации угрозы	Основные соотношения	Принятые обозначения
Модель динамики распространения вируса на множестве компьютеров за счет переноса вредоносных программ на отчуждаемых носителях	От каждого компьютера заражается некоторое количество других компьютеров. количество компьютеров достаточно велико и не ограничивает процесс распространения вредоносной программы	$\xi(t)=\xi_1(t)+\xi_2(t)+\ldots+\xi_k(t)$; $P\{\xi_i(t)=n\}=p_n(t,n),\ n=0,1,\ldots$; $p_n(\Delta t,n)=\lambda_n\cdot\Delta t+o(\Delta t),\ n\neq 1$; $\frac{d}{dt}p_n(t,n)=\sum_k\lambda_k\cdot p_{nk}(t,n)$; $\begin{cases}F(t,z)=\sum_{n=0}^{\infty}p_n(t,n)\cdot z^n,\\ F_k(t,z)=\sum_{n=0}^{\infty}p_{nk}(t,n)\cdot z^n\end{cases}$ $\frac{d}{dt}F(t,z)=\sum_k\lambda_k\cdot F^k(t,z)$	$\xi(t)$ – количество компьютеров пораженных к моменту t; $\xi_j(t)$ количество компьютеров, пораженных вредоносной программой, перенесенной с j-го компьютера; $P_n(\Delta t,n)$ вероятность того, сто за время Δt от одного компьютера заражается n компьютеров; λ_n - интенсивность заражения n компьютеров; $p_{nk}(t,n)$ - переходные вероятности Марковского ветвящегося процесса, моделирующего распространение вредоносной программы, соответствующие вероятности того, что от k компьютеров заражаются n других компьютеров; $F(t,z)$, $F_k(t,z)$ - производящие функции [33]
Модель динамики распространения вируса по компьютерной сети [104]	Первоначально в КС заражено некоторое количество узлов. Процесс распространения является ординарным. Любой из незараженных узлов инфицируется с равной вероятностью	$P_n(t,n)=\sum_{j=0}^{m+h-1}\left(d_j+l_j\cdot t\right)\cdot exp\left[-j\cdot(m+h-j)\cdot\mu_{inf}\cdot(1-P_{det})\cdot t\right]+$ $+\sum_{j=m+1}^{n}r_{nj}\cdot exp\left[-j\cdot(m+h-j)\cdot\mu_{inf}\cdot(1-P_{det})\cdot t\right]+k\cdot exp\left[-\left(\frac{m+h}{2}\right)^2\cdot\mu_{inf}\cdot t\right]$ $d_j=\frac{(n-1)!m!}{(m-1)!(m+h-n)}\cdot\left\{\frac{[s+j(m+h-j)]}{\prod_{j=0}^{n}[s+j(m+h-j)]}\right\}_{s=-j(m+h-j)}$; $l_j=\frac{(n-1)!m!}{(m-1)!(m+h-n)}\cdot\left\{\frac{[s+j(m+h-j)]^2}{\prod_{j=0}^{n}[s+j(m+h-j)]}\right\}_{s=-j(m+h-j)}$; $k=r_{nj}$ при $j=(m+h)/2$ $r_{nj}=\frac{(n-1)!m!}{(h-1)!(M+H-n)!}\cdot\left\{\left[\prod[s+j(m+H-j)]\right]_{s=-j(m+H-j)}\right\}^{-1}$	μ_{inf} - интенсивность инфицирования компьютеров (заражения одного узла сети от другого); n – количество зараженных компьютеров за время t; m _ количество компьютеров, которые в начальный момент времени не заражены; h - количество компьютеров, зараженных к началу рассматриваемого процесса распространения вредоносной программы; p_{det} - вероятность обнаружения вредоносной программы в КС; $d_j, l_j, k, r_{n,j}$ -вспомогательные параметры; s _ параметр преобразования Лапласа

Приведенные соотношения позволяют рассчитать вероятностно-временные характеристики динамики такого распространения. В частности, для модели с многократным последовательным заражением файлов с определенными расширениями имен и отсутствии ограничения на количество поражаемых файлов зависимость вероятности $P_u(t,n)$ того, что за время t окажется заражено n файлов, от количества инфицируемых файлов в графическом виде представлена на рисунке 5.35, а от времени – на рисунке 5.36.



1) $t = 7c,\ \bar{\tau}_{rec} = 0.1c,\ \bar{\tau}_{ret} = 0.2c,\ \bar{\tau}_{\inf} = 0.5c$; 2) $t = 50c,\ \bar{\tau}_{rec} = 0.2c,\ \bar{\tau}_{ret} = 0.3,\ \tau_{\inf} = 1c$

Рисунок 5.35 – Зависимость вероятности инфицирования от количества инфицированных файлов



1) $n = 10,\ \tau_{rec} = 0.1c,\ \bar{\tau}_{ret} = 0.2c,\ \bar{\tau}_{\inf} = 0.5c$;

2) $n = 30,\ \bar{\tau}_{rec} = 0.2c,\ \bar{\tau}_{ret} = 0.3,\ \bar{\tau}_{\inf} = 1c$

Рисунок 5.36 – Зависимость вероятности инфицирования от времени

В основу модели динамики распространения вируса на множестве компьютеров была положена теория марковских случайных ветвящихся процессов.

Пример. Рассмотрим случай, когда от каждого компьютера может быть заражено не более двух других компьютеров, то есть $\lambda_0 = 0,\ \lambda_1 = -\lambda,\ \lambda_2 = \lambda,\ \lambda_k = 0\ для\ \ k > 2$.

В этом случае:

$$F(t,z) = e^{-\lambda \cdot t} \cdot \frac{z}{1 - z \cdot (1 - e^{-\lambda \cdot t})} = e^{-\lambda \cdot t} \cdot z \cdot \sum_{k=0}^{\infty} (1 - e^{-\lambda \cdot t})^k \cdot z^k, \qquad (5.115)$$

что приводит к следующим соотношениям:

$$p_0(t)=0,\ p_n(t)=e^{-\lambda\cdot t}\cdot(1-e^{-\lambda\cdot t})^{n-1},\ n=1,2,\ldots. \qquad (5.116)$$

Среднее количество пораженных компьютеров за заданное время t, если, например, в момент времени $t=0$ был поражен только один компьютер, оценивается следующим образом:

$$\bar{n}(t)=\frac{d}{dz}F(t,z)\Big|_{z=0}=e^{\lambda\cdot t}. \qquad (5.117)$$

Полученная зависимость количества поражаемых компьютеров от параметров инфицирования в виде графиков показана на рисунке 5.37.

Для учета влияния антивирусного пакета на динамику распространения вредоносной программы необходимо ввести вероятность ее обнаружения P_{det} по сигнатуре вируса и блокирования его распространения, при этом количество поражаемых компьютеров определится из соотношения:

$$\overline{n(t)}=exp\left\{P_{inf}\cdot\left(1-P_{det}\right)\cdot\left(\mu_{out}+\mu_{his}\right)\cdot t\right\} \qquad (5.118)$$

Данная зависимость графически представлена на рисунке 5.38.

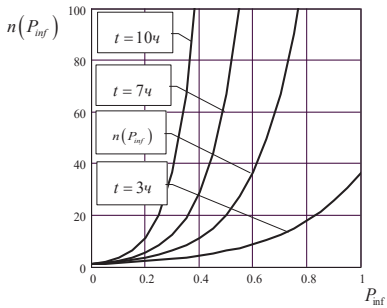


Рисунок 5.37 – Зависимость количества инфицируемых компьютеров от средней вероятности инфицирования каждого компьютера для различного времени, прошедшего с начала инфицирования

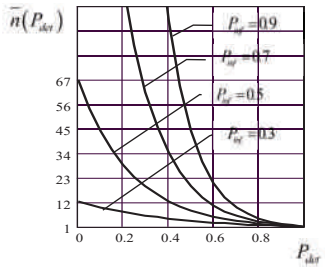


Рисунок 5.38 – Зависимость количества инфицированных компьютеров от вероятности обнаружения и блокирования ("ликвидации") вредоносной программы

Аналогичным образом строятся модели динамики распространения вредоносных программ по компьютерной сети. Отличия заключаются в необходимости введения дополнительных параметров инфицирования, таких как частота обмена информацией между хостами сети, частота передачи инфицированных файлов по сети, количества инфицированных файлов в компьютере – источнике заражения и т.д.

Следует отметить, что разработка таких моделей применительно к разным классам вредоносных программ является весьма важным и перспективным направлением развития методического обеспечения оценки эффективности защиты информации в ИС. Изложенный подход показывает возможность и пути решения данной важной для практики задачи.

5.7. Модели динамики реализации угроз с использованием аппарата полумарковских процессов

Полумарковские процессы (ПМП) представляют собой обобщение достаточно хорошо изученных в теории вероятностей марковских процессов. Они введены в 1954–1955 гг. независимо П. Леви, В. Смитом и Л. Такачем и сегодня достаточно широко и успешно используются в теории надежности и теории массового обслуживания. В соответствии с [83–85] суть ПМП сводится к следующему.

Пусть в начальный момент времени $t = 0$ моделируемый процесс находится в состоянии $i \in I$, в течении некоторого случайного времени θ_0, после чего процесс мгновенно переходит в состояние $j, (j \in I)$. При этом время θ_0 – случайная величина с *произвольной* функцией распределения $F_{ij}(t)$, а переход процесса из состояния i в состояние j происходит с вероятностью $\pi_{ij} > 0$ и для любого i выполняется условие $\sum_{i \in I} \pi_{ij} \le 1$. Если из состояния j процесс переходит в состояние $k \in I$, то в состоянии j процесс пребывает случайное время θ_1, имеющее произвольное распределение $F_{jk}(t)$ и т.д. (рисунок 5.39).

ПМП так же, как и цепь Маркова, является скачкообразным случайным процессом, при этом распределение времени пребывания ПМП в каждом состоянии не обязательно является показательным, что отличает полумарковский процесс от марковского.

Для ПМП характерно то, что процесс мгновенных вероятностных переходов представляет собой вложенную цепь Маркова[24]. Полумарковский процесс, как и марковский, как правило, представляют в виде ориентированного графа (рисунок 5.40).

[24] Метод вложенных цепей Маркова впервые предложил Д.Кендалл в [2].

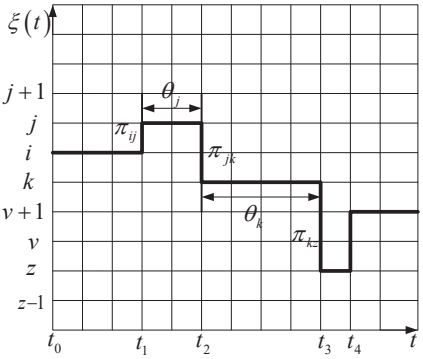


Рисунок 5.39 – Графическое представление полумарковского процесса

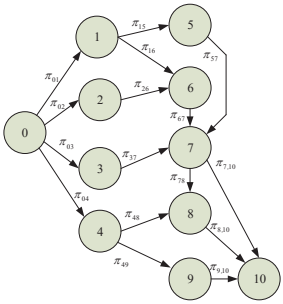


Рисунок 5.40 – Пример графа состояний и переходов полумарковского процесса

Случайные переходы из состояния в состояние в ПМП определяются матрицей $Q(x) \equiv \left\{ q_{ij}(x), i, j \in I \right\}$, получившей название *полумарковской матрицы,* которая удовлетворяет следующим условиям [84, 85]:

1) $q_{ij}(t) \equiv 0$, если $t < 0$;

2) $q_{ij}(t)$ – неубывающие, непрерывные справа функции для $t \geq 0$;

3) $\sum_{j \in I} q_{ij}(t) \equiv F_i(t) \leq 1$ – функция распределения времени пребывания процесса в состоянии i.

Примеры полумарковских матриц [85]:

$$p_0(t) = 0,\ p_n(t) = e^{-\lambda \cdot t} \cdot (1 - e^{-\lambda \cdot t})^{n-1},\ n = 1,2,...; \tag{5.119}$$

$$q_{ij}(t) = \pi_{ij},\ t \geq 0,\ \sum_{i,j \in I} \pi_{ij} = 1; \tag{5.120}$$

$$q_{ij}(t) = \pi_{ij} \cdot \left(1 - e^{-\lambda_i \cdot t}\right),\ t \geq 0,\ \sum_{i,j \in I} \pi_{ij} = 1; \tag{5.121}$$

$$q_{ij}(t)=\pi_{ij}\cdot F_i(t),\, t\geq 0,\, \sum_{i,j\in I}\pi_{ij}=1. \qquad (5.122)$$

В соответствии с [85] ПМП может быть задан четырьмя следующими способами:

1) начальным распределением $p=\{p_i, i\in I\}$, с помощью которого выбирается исходное состояние i, и полумарковской матрицей $Q(t)$;

2) начальным распределением $p=\{p_i, i\in I\}$, матрицей переходных вероятностей вложенной цепи Маркова $\{\pi_{ij}\},(i,j)\in I$ и матрицей функции распределения $F_{ij}(t),(i,j\in I)$ времени пребывания процесса в состоянии i при условии, что переход осуществляется в состояние j;

3) начальным распределением $p=\{p_i, i\in I\}$, вектором функции распределения $F_i(t)$ времени пребывания и матрицей $q_{ij}(t)$ условных вероятностей того, что ПМП $\xi(t)$ попадет в состояние j, пробыв в состоянии i время, равное t.

С помощью полумарковской матрицы определяются переходные вероятности $F_{ij}(t)$ путем решения системы интегро-дифференциальных уравнений следующего вида:

$$F_{ij}(t)=\left[1-P_i(t)\right]\cdot\delta_{ij}+\sum_{k\in I}\int_0^t q'_{ik}(x)\cdot F_{kj}(t-x)dx,(k,i,j)\in I, t>0, \qquad (5.123)$$

где $q'_{ik}(t)$ – производная от функции (элемента полумарковской матрицы);

$P_i(t)$ – вероятность того, что процесс не покинет i состояния в момент времени t;

δ_{ij} – символ Кронекера.

Если определить время пребывания в состоянии i при условии, что после его окончания процесс перейдет в состояние j через θ_{ij}, то время θ_i пребывания ПМП в состоянии i вычисляется из соотношения:

$$\theta_i=\sum_{j\in I}\theta_{ij}, \qquad (5.124)$$

а переходные вероятности π_{ij} – по формуле:

$$\pi_{ij}=\theta_{ij}/\theta_i=\theta_{ij}\sum_{j\in I}\theta_{ij}, i,j\in I. \qquad (5.125)$$

С учетом этих соотношений возможно и четвертое представление ПМП, которое включает в себя начальное распределение $p=\{p_i, i\in I\}$ и матрицу $\{\theta_{ij}\}$, при этом независимые неотрицательные величины θ_{ij} имеют распределения в виде:

$$\Theta_{i,j}(t) = 1 - \exp\left\{ -\int_0^t \frac{q_{ij}^{'}(x) \cdot dx}{1 - P_i(x)} \right\}. \qquad (5.126)$$

С помощью аппарата ПМП можно рассчитать вероятностно-временные характеристики перехода моделируемого случайного процесса с начального в любое последующее состояние.

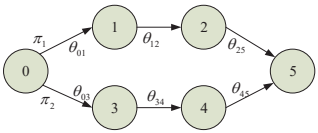
Пример 5.6. Пусть необходимо рассчитать вероятностно-временные характеристики реализации угрозы несанкционированного доступа к файлу с конфиденциальной информацией одним из следующих двух способов: 1) путем повышения привилегий и 2) путем изменения учетных записей. Оба способа основаны на внедрении соответствующих вредоносных программ. Вероятность применения первого способа составляет π_1, а второго – π_2. Граф соответствующего полумарковского процесса приведен на рисунке 5.41.

Пусть времена переходов подчиняются экспоненциальным распределениям с плотностями распределения времени пребывания в i-м состоянии:

$$f_{ij}(t) = \frac{d}{dt} F_{ij}(t) = \frac{1}{\overline{\theta}_{ij}} e^{-\frac{t}{\overline{\theta}_{ij}}}. \qquad (5.127)$$

При этом $\theta_{01} = \theta_{12} = \theta_{25} = \theta_1$ и $\theta_{03} = \theta_{34} = \theta_{45} = \theta_2$.

Из формулы (5.127) следует, что плотность распределения вероятности перехода процесса из начального в конечное состояние (состояние 8) является сверткой плотностей распределения вероятностей переходов из состояния в состояние по траекториям перемещения, соответствующим первому и второму способам доступа.



Состояния процесса:

0 – запущен процесс внедрения вредоносных программ или для изменения учетных записей, или для повышения привилегий с несанкционированным использованием системных функций;

1 – внедрена вредоносная программа для изменения учетных записей и запущена на выполнение, изменена учетная запись, созданы условия для проникновения к файлу с конфиденциальной информацией, начата попытка проникновения;

2 – получен доступ к файлу с конфиденциальной информацией, начато копирование файла в выбранную область памяти или на флэш-носитель;

3 – внедрена и запущена на выполнение вредоносная программа для повышения привилегий атакующего;

4 – повышены привилегии атакующего, начата попытка проникновения;

5 – получен доступ к файлу с конфиденциальной информацией, файл скопирован в выбранную область памяти или на флэш-носитель, угроза реализована;

Рисунок 5.41 – Граф полумарковского процесса реализации угрозы несанкционированного доступа к файлам защищаемой информации двумя способами

Тогда:

$f_{05}^{(1)}(t) = f_{01}(t) * f_{12}(t) * f_{25}(t)$ – для первого способа несанкционированного доступа;

$f_{05}^{(2)}(t) = f_{03}(t) * f_{34}(t) * f_{45}(t)$ – для второго способа несанкционированного доступа, где знак * означает операцию свертки [92, 97].

При экспоненциальном приближении это приводит к соотношениям:

$$f_{05}^{(1)}(t) = \frac{1}{2} \cdot t^2 \cdot e^{-\frac{t}{\overline{\theta_1}}}, f_{05}^{(2)}(t) = \frac{1}{2} \cdot t^2 \cdot e^{-\frac{t}{\overline{\theta_2}}}. \quad (5.128)$$

Вероятность того, что за время t процесс достигнет конечного состояния, определяется из соотношения:

$$F_{05}(t) = \pi_1 \cdot \int_0^t f_{05}^{(1)}(x) \cdot dx + \pi_2 \cdot \int_0^t f_{05}^{(2)}(x) \cdot dx \quad (5.129)$$

или

$$F_{05}(t) = 1 - \pi_1 \cdot e^{-\frac{t}{\overline{\theta_1}}} \cdot \left\{ \frac{t^2}{2 \cdot \overline{\theta_1^2}} + \frac{t}{\overline{\theta_1}} + 1 \right\} - \pi_2 \cdot e^{-\frac{t}{\overline{\theta_2}}} \cdot \left\{ \frac{t^2}{2 \cdot \overline{\theta_2^2}} + \frac{t}{\overline{\theta_2}} + 1 \right\}. \quad (5.130)$$

График полученной зависимости для выбранных значений средних времен выполнения парциальных процессов приведен на рисунке 5.42.

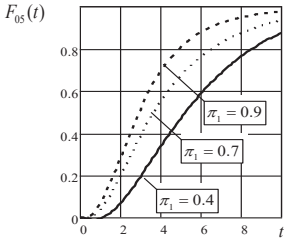


Рисунок 5.42 – График зависимости вероятности достижения моделируемым процессом конечного состояния за время t при $\pi_1 = 0.4, \pi_2 = 0.6, \overline{\theta_1} = 1, \overline{\theta_2} = 2$

Аппарат ПМП сегодня применяеnся, в основном, в теории надежности, при анализе и построении систем массового обслуживания. Находит он применение также и при решении некоторых задач в предметной области защиты информации в информационных системах.

Однако в традиционном представлении этот аппарат имеет определенные ограничения, не позволяющие создавать целый ряд важных для практики моделей динамики реализации угроз безопасности информации в ИС.

К основным из них относятся следующие:

невозможность учета логических условий, которые могут иметь место в моделируемых процессах;

сложность моделирования циклических операций с задаваемыми условиями завершения их выполнения (например, по количеству выполненных циклов);

наличие только одного начального состояния процесса, в то время как могут иметь место несколько состояний, из которых может начинаться процесс с определенными (возможно случайными) задержками во времени.

В разделе 6 рассмотрены расширения аппарата ПМП, позволяющие парировать эти и иные ограничения и значительно расширить его моделирующие возможности.

6. МАТЕМАТИЧЕСКИЕ МОДЕЛИ ДИНАМИКИ РЕАЛИЗАЦИИ УГРОЗ БЕЗОПАСНОСТИ ИНФОРМАЦИИ НА ОСНОВЕ АППАРАТА СЕТЕЙ ПЕТРИ-МАРКОВА

6.1. Традиционные сети Петри-Маркова и особенности их применения при моделировании динамики реализации угроз безопасности информации в информационных системах

Под сетью Петри-Маркова[25] (СПМ) понимается множество $\psi = \{\Xi, M\}$, при этом Ξ – сеть Петри, которая представляет собой двудольный граф вида [91]:

$$\Xi = \{A, Z, I_A(Z), O_Z(A)\}, \tag{6.1}$$

где A – множество позиций СПМ, моделирующих состояния рассматриваемого процесса (соответственно традиционно позиции сети обозначаются цифрой и буквой a);

Z – множество переходов СПМ, моделирующих условия перехода из состояния в состояние (соответственно традиционно переход сети обозначаются цифрой и буквой z);

$I_A(Z)$ – входная функция переходов (выходная функция позиций), отображающая множество A в множество Z;

$O_Z(A)$ – выходная функция переходов (входная функция позиций, отображающая множество Z в множество A,

а M представляет собой случайный процесс, определяющий логико-вероятностные свойства модели и описываемый четверкой

$$M = \left\{\vec{\mathbf{Q}}, \mathbf{\Pi}, \mathbf{f}(t), \mathbf{\Psi}\right\}, \tag{6.2}$$

при этом $\vec{\mathbf{Q}} = \{q_{1(z)}, q_{2(z)}, \ldots q_{j(z)}\}$ – вектор вероятностей запуска моделируемого процесса из каждого перехода сети;

$\mathbf{\Pi} = \left[\pi_{i(a),j(z)}\right]$ – матрица переходных вероятностей, определяющая вероятности перемещения процесса из позиции $i(a)$ в переход $j(z)$. Далее принимается, что первый индекс в переходных вероятностях соответствует позиции, а второй – переходу, и матрица записывается следующим образом $\mathbf{\Pi} = \left[\pi_{i,j}\right]$;

$\mathbf{f}(t) = \left[f_{i(a),j(z)}(t)\right] \equiv \left[f_{ij}(t)\right]$ – матрица плотностей распределения времени перемещения процесса из позиции $i(a)$ в переход $j(z)$;

$\mathbf{\Psi} = \left[\psi_{j(z),i(a)}\right] \circ \left[\psi_{ji}\right]$ – матрица логических условий, элементы которой равны 0 или 1 при выполнении задаваемых различными способами условий (например, в виде логических функций, высказываний и др.),

[25] Впервые аппарат сетей Петри-Маркова как симбиоз сетей Петри и марковских и полумарковских процессов предложен в работах ученых Тульского Государственного университета Игнатьева В.М., Ларкина Е.В, Сабо Ю.И. применительно к решению задач надежности.

$$\psi_{j(z),i(a)} = \begin{cases} \psi[I_A(z_{j(z)})], \text{ если } a_{i(a)} \in O_A(z_{j(z)}); \\ 0, \text{если } a_{i(a)} \notin O_A(z_{j(z)}), \end{cases} \tag{6.3}$$

при этом $z_{j(z)}$ – это элемент множества Z, то есть переход с номером $j(z)$, а $i(a)$ – это номер элемента множества A, то есть позиции, в которую процесс перемещается из перехода с номером $j(z)$.

Матрицы вероятностей и плотностей распределения в некоторых случаях могут заменяться полумарковской матрицей

$$\mathbf{h}(t) = \left[\pi_{i(a),j(z)} \times f_{i(a),j(z)}\right] = \left[h_{i(a),j(z)}\right]. \tag{6.4}$$

В этом случае полумарковские и логические свойства модели описываются тройкой

$$M = \{\mathbf{Q}, \mathbf{h}(t), \mathbf{\Psi}\}. \tag{6.5}$$

Последовательность перемещений по СПМ реализуется в виде полушагов.

Полушагом называют перемещение по СПМ из позиции $i(a)$ в сопряженный с нею переход $j(z)$ или из перехода $j(z)$ в сопряженную с ним позицию $k(a)$. Совокупность полушагов в СПМ таким образом представляется в виде упорядоченных множеств $\sigma_{a,z} = \{i(a), j(z)\}$ и $\sigma_{z,a} = \{j(z), k(a)\}$. Два последовательных полушага образуют шаг.

СПМ, как и сети Петри, удобно представлять в виде ориентированных взвешенных графов. Пример такого ориентированного графа приведен на рисунке 6.1.

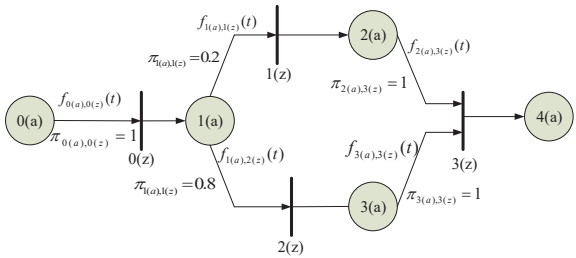


Рисунок 6.1 – Пример графа сети Петри-Маркова

На рисунке позиции обозначены кружочками, переходы – жирной чертой, а возможность выполнения полушага отмечается стрелкой, при этом над стрелкой, указывающей перемещение из позиции в переход, могут указываться плотность распределения времени и вероятность выполнения полушага из позиции в переход.

Перемещение процесса *из перехода в позицию считается мгновенным и происходит с единичной вероятностью*. С использованием понятия полушага вводятся понятия входных и выходных позиций и переходов. Выходной (входной) позицией $a_{i(a)}$ перехода $z_{j(z)}$ называется позиция, в которую (из которой) можно попасть из (в) $z_{j(z)}$ за один полушаг.

Входные и выходные позиции перехода $z_{j(z)}$ образуют множества: I_A

$$I_A(z_{j(z)}) = \left\{a_{k_1}, a_{k_2}, ...a_{k_r}\right\}, r = \overline{1, R_{j(z)}}\ ;\ O_A(z_{j(z)}) = \left\{a_{n_1}, a_{n_2}, ...a_{n_r}\right\}, r = \overline{1, R^{(O)}_{j(z)}},$$

где $R^{(I)}_{j(z)}$ – количество позиций, из которых имеются дуги в переход $z_{j(z)}$;

$R^{(O)}_{j(z)}$ – количество позиций, в которые входят дуги из перехода $z_{j(z)}$.

Входной функцией позиций $I_Z\left(A\right)$ называется функция, отображающая множество позиций A в подмножества переходов $I_Z(A) = \left\{I_Z(a_{i(a)}), ..., I_Z(a_{J(a)})\right\}$, из которых есть вход в позиции из множества A (входных переходов).

Выходной функцией позиций $O_Z\left(A\right)$ называется функция, отображающая множество A в подмножества выходных переходов $O_Z(A) = \left\{O_Z(a_{k(a)}), ..., O_Z(a_{K(a)})\right\}$, в которые есть вход из позиций, составляющих множество A (выходных переходов).

Входной функцией переходов $I_A\left(Z\right)$ называется функция, отображающая множество переходов Z в подмножества позиций $I_A(Z) = \left\{I_A(z_{j(z)}), ..., I_A(z_{J(z)})\right\}$, из которых есть вход в переходы из множества Z (входные позиции).

Выходной функцией переходов $O_A(Z)$ называется функция, отображающая множество Z в подмножества позиций $O_A(Z) = \left\{O_A(z_{j(z)}), ..., O_A(z_{J(z)})\right\}$ (выходных позиций).

СПМ считается ординарной, если в множествах $I_Z\left(a_{i(a)}\right)$ и $O_Z\left(a_{k(a)}\right)$ переход $z_{j(z)}$, а в множествах $I_A\left(z_{j(z)}\right)$ и $O_A\left(z_{j(z)}\right)$ – позиция $a_{k(a)}$ могут встретиться не более одного раза.

Переходы СПМ разделяют на примитивные и непримитивные.

Примитивным переходом СПМ называется переход $z_{j(z)}$, для входной и выходной функций которого выполняются условия $\aleph\left[I_A(z_{j(z)})\right] = \aleph\left[O_A(z_{j(z)})\right] = 1$, где $\aleph[..]$ – мощность соответствующего множества.

Все остальные типы переходов – непримитивные. Непримитивные переходы образуют подмножество $Z_{zn} \in Z$ (для них вводится специальная индексация $j(zn)$, например, $1(zn) \le 2(zn) \le ... \le j(zn) \le ... \le J(zn)$). Примеры непримитивных переходов показаны на рисунке 6.2.

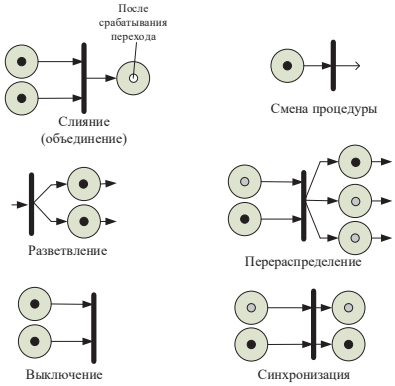


Рисунок 6.2 – Примеры непримитивных переходов СПМ

Непримитивный переход $z_{j(z)}$, для выходной функции которого выполняется условие $\aleph\left[O_A\left(z_{j(zn)}\right)\right]=0$, называется конечным переходом. Очевидно, что конечные переходы образуют непустое множество $Z_E=\left\{z_{1(E)},\ldots,z_{J(E)}\right\}\neq\varnothing$, иначе процесс перемещений по СПМ был бы бесконечным.

Стартовым переходом называется переход, из которого начинается последовательность перемещений по СПМ, при этом конечный переход не может быть стартовым, а любой непримитивный переход, не являющийся конечным, может быть стартовым.

Переходы, в зависимости от их текущего состояния, разделяются на закрытые и открытые. Переход $z_{j(z)}$ называется закрытым, если из него не разрешен полушаг ни в одну из позиций подмножества $O_A\left(z_{j(z)}\right)$.

Переход $z_{j(z)}$ называется частично открытым, если в подмножестве $O_A(z_{j(z)})$ существуют элементы, полушаг в которые разрешен. В этом случае полушаги из указанного перехода в его выходные позиции могут быть разделены на разрешенные и запрещенные.

Переход $z_{j(z)}$ называется открытым, если в подмножестве $O_A\left(z_{j(z)}\right)$ отсутствуют элементы, полушаг в которые не разрешен.

Логическим условием $\psi_{j(z),i(a)}$ разрешения полушага называется условие, при выполнении которого переход открывается для выполнения полушага $\sigma_{j(z),i(\ )}$. Полушаг из позиции $a_{i(a)}$ в переход $z_{j(z)}$ (необязательно примитивный) является случайным событием и определяется вероятностью $\pi_{i(a),j(z)}$, при этом справедливо равенство

$$\sum_{j(z)=1(z)}^{J(z)} \pi_{i(a),j(z)} = 1. \qquad (6.6)$$

Независимо от вероятностей $\pi_{i(a),j(z)}$ полушаг из позиции $a_{i(a)}$ в переход $z_{j(z)}$ производится в течение случайного времени, определяемого плотностью распределения $f_{i(a),j(z)}(t)$, для которой (поскольку указанное время является положительно определенной случайной величиной) справедливы ограничения:

$$f_{i(a),j(z)}(t) = 0 \text{ при } t \le 0 \text{ и } \int_0^{\infty} f_{i(a),j(z)}(t)dt = 1. \qquad (6.7)$$

Для расширения возможности интерпретации состояний моделируемого процесса могут использоваться так называемые псевдопозиции, введение которых в модели процессов диктуется необходимостью разделения переходов в моделях.

Перемещение процесса из псевдопозиции в переход считается мгновенным, то есть $f_{i(a),j(z)}(t) = \delta(t)$, где $\delta(t)$ – дельта-функция (функция Дирака) [92],

$$\delta(t) = \begin{cases} 0 \text{ при } t \ne 0; \\ \infty \text{ при } t = 0; \end{cases} \text{ и } \int_{-\infty}^{\infty} \delta(t)dt = 1.$$

Для обозначения перемещения процесса по СПМ используются, как и в сетях Петри, фишки (метки, маркеры), при этом позиции и непримитивные переходы в зависимости от их состояния разделяются на помеченные и непомеченные. Множество помеченных компонентов СПМ формирует ее разметку.

Разметкой СПМ $\Theta(\Pi)$ называется выделение в ее структуре позиций и непримитивных переходов путем помещения в них фишек. Как позиция, так и непримитивный переход называются помеченными, если они содержат хотя бы одну фишку. Помеченная позиция $a_{i(a)}$ обозначается $a_{i(a,\theta)}$, а помеченный переход – $z_{j(zn,\theta)}$, при этом

$$\Theta(\Pi) = \left\{ a_{1(a,\theta)}, ..., a_{i(a,\theta)}, ..., a_{I(a,\theta)}, z_{1(zn,\theta)}, ..., z_{j(zn,\theta)}, z_{J(zn,\theta)} \right\}. \qquad (6.8)$$

Разметка может быть начальной (0-разметка) и текущей (n-разметка). 0-разметка $\Theta^{(0)}(\Xi)$ формируется при старте процесса, а n-разметка $\Theta^{(n)}(\Xi)$ – после n полушагов процесса.

Для управления срабатыванием переходов могут применяться также J-переходы (введены в Е-сетях, см. раздел 4.5), которые выполняются только тогда, когда в выходной позиции отсутствует фишка и имеются фишки во входящих позициях, в том числе в позиции, инцидентной входящей управляющей дуге.

Изложенная краткая характеристика классического аппарата сетей Петри-Маркова дает лишь общее представление о нем, но позволяет сделать важные выводы.

Во-первых, классическая сеть Петри-Маркова остается автоматной сетью, то есть срабатывание переходов в ней определяется наличием в позициях, инцидентных переходу, фишек и, в случае наличия управляющих дуг (например, ингибиторных), – фишек в позициях, инцидентных таким дугам. Это обусловливает то, что в таких сетях крайне сложно, а чаще практически невозможно моделировать динамику выполнения целого ряда логических условий, таких как условия «И-НЕ», «ИЛИ-НЕ», исключающее «ИЛИ» («XOR»), «И-ИЛИ» и др., которые часто встречаются в моделях реализации угроз безопасности информации в ИС. При этом сложность состоит не в определении логических правил (что достаточно просто записывается в соответствии с математической логикой), а в определении соотношений для времен срабатывания таких переходов. В частности, в [91] показано, каким образом можно рассчитать время ожидания завершения второй процедуры в составе моделируемого процесса после завершения первой, если процесс завершается после завершения второй процедуры. Однако для других условий формулы для расчета временных характеристик отсутствуют.

Во-вторых, весьма сложно моделировать циклические процедуры с заранее установленными правилами их завершения и дальнейшего продолжения выполнения процесса.

С учетом изложенного моделирующие возможности классического аппарата сетей Петри-Маркова применительно к решению задач аналитического моделирования процессов реализации угроз безопасности информации в ИС оказываются ограниченными.

Для расширения моделирующих возможностей СПМ в [35] предложено использовать:

– окрашивание позиций, маркеров и переходов СПМ (применение цветных СПМ);

– логические запреты на выполнение переходов с использованием ингибиторых (сдерживающих) дуг[26] (по аналогии с классическими сетями Петри) или специальных ограничений на срабатывание переходов (рисунок 6.3);

– формирование так называемых иерархических сетей, в которые вложены другие, возможно, также иерархические, сети;

– приоритеты срабатывания переходов, при этом, если несколько переходов являются разрешенными, то срабатывает тот из них, который имеет наивысший приоритет;

– изменяемую структуру, в которой кратность ребер не является постоянной, при этом в самомодифицируемых сетях кратность ребра может

[26] В теории сетей Петри ингибиторные («сдерживающие») дуги применяются для обнаружения «нулевого» условия или введения запрета на выполнение перехода.

задаваться либо натуральным числом, либо определяться количеством фишек, находящихся во входных позициях некоторого перехода;

– временные сети, в которых переходам ставятся в соответствие их времена срабатывания, либо позициям ставятся в соответствие времена нахождения фишек в позициях (по сути, временные сети предшествовали сетям Петри-Маркова).

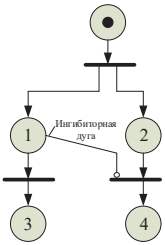


Рисунок 6.3 – Сеть Петри-Маркова с ингибиторной дугой

Следует отметить, что применение «цветных» СПМ позволяет разрабатывать модели процессов, в которых окрашенным переходам поставлены в соответствие определенные операторы, выполняемые в зависимости от цветов маркера, перехода и инцидентных переходу позиций, что заметно расширяет моделирующие возможности данного аппарата. Применение же запретов, ингибиторных дуг и специальных ограничений приводит к необходимости поиска путей определения аналитических соотношений для расчета времени срабатывания переходов сети с такими условиями, что на практике оказывается не всегда возможным.

Далее рассматриваются иные модификации аппарата СПМ, расширяющие его моделирующие возможности.

6.2. Составные сети Петри-Маркова на основе марковских процессов с пропозициональной логикой срабатывания логических переходов

Пусть СПМ состоит из нескольких подпроцессов (парциальных марковских процессов), при этом переход от одного подпроцесса к другому осуществляется только при выполнении определенных логических условий. Переход, срабатывающий при определенных логических условиях, далее называется логическим переходом. Для того чтобы отделить подпроцессы, завершающиеся логическими условиями, от других подпроцессов, вводятся непримитивные переходы, срабатывающие при определенной логике наступления событий. СПМ с логическими переходами, соединяющими парциальные процессы, в [35] названы составными сетями Петри-Маркова (ССПМ). В отличие от классических СПМ, в которых переходы срабатывают, если для каждой входящей в переход дуги имеется не менее одного маркера, в данном случае для перехода может быть определена любая логика срабатывания.

Как и классические СПМ, составные СПМ представляются в виде графов. Пример графа ССПМ, построенной с использованием марковских процессов с логическими условиями, приведен на рисунке 6.4.

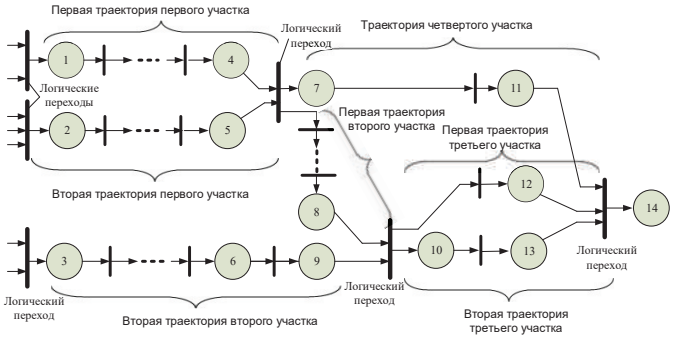


Рисунок 6.4 – Пример графа составной сети Петри-Маркова, построенной с использованием парциальных марковских процессов

На этом рисунке каждый участок графа соответствует одному из парциальных марковских процессов. При построении такого графа необходимо соблюдать следующие правила:

ССПМ должна начинаться с начальной позиции (или нескольких начальных позиций) и заканчиваться конечной позицией;

в ССПМ могут иметь место тупиковые позиции, в которых развитие процесса прекращается, для того чтобы показать, где может остановиться процесс, не достигнув конечной позиции;

должна быть установлена начальная разметка сети;

в сети не может быть две подряд позиции или два подряд перехода;

в сети могут применяться кратные дуги, но тогда, переходам, в которые входят кратные дуги, должны быть сопоставлены соответствующие правила срабатывания;

в ССПМ может иметь место разветвление процесса после переходов, то есть из перехода сети может исходить несколько дуг к разным позициям, при этом по всем таким дугам маркер (по каждому маркеру на дугу путем размножения одного маркера) перемещается в инцидентную позицию мгновенно с единичной вероятностью.

Далее для упрощения в данной работе на графе ССПМ переходы обозначаются номером с буквой z, а позиции просто нумеруются, в индексах функций и параметров первая буква означает номер состояния, из которого исходит дуга, а вторая – номер перехода, в который она входит, при двузначных цифрах применяется разделяющая запятая.

Расчет времени перемещения процесса оценивается по каждой траектории как сумма случайных времен перемещения из позиций в переходы.

При этом следует иметь в виду (см. раздел 6.1), что процесс из перехода в позицию ССПМ перемещается мгновенно, а из позиции в переход – за конечное в общем случае случайное время. Условия срабатывания логических переходов в ССПМ могут формально вводиться с использованием аппарата пропозициональной логики (таблица 6.1), с использованием операций с предикатами второго порядка или иными способами.

Таблица 6.1 – Основные логические условия и вероятности срабатывания переходов

Логическое условие	Описание условия	Вероятность срабатывания перехода
$A \wedge B$ («И»)	Логический переход срабатывает, если на его вход поступили все подпроцессы из совокупностей А и В	$P_И(t) = \prod_{i \in A} P_i(t) \cdot \prod_{j \in B} P_j(t)$
$A \vee B$ («ИЛИ»)	Логический переход срабатывает, если на его вход поступили или все подпроцессы из совокупности А, или все подпроцессы из совокупности В, или и те, и другие	$P_{ИЛИ}(t) = 1 - \prod_{i \in A}[1 - P_i(t)] \cdot \prod_{j \in B}[1 - P_j(t)] = 1 - \prod_{i \in A \wedge B}[1 - P_i(t)]$
$A \vee B \wedge C$ («ИЛИ-И»)	Логический переход срабатывает, если на его вход поступили все подпроцессы хотя бы одной из совокупностей А или В и все подпроцессы из совокупности С	$P_{ИЛИ-И}(t) = \left\{1 - \prod_{i \in A \wedge B}[1 - P_i(t)]\right\} \times \prod_{k \in C} P_k(t)$
$A \oplus B$ (XOR или эксклюзивный выбор)	Логический переход срабатывает, если на его вход поступили или только все подпроцессы из совокупности А, или только все подпроцессы из совокупности В	$P_{XOR}(t) = \prod_{i \in A} P_i(t) \cdot \prod_{j \in B}[1 - P_j(t)] + \prod_{i \in A}[1 - P_i(t)] \cdot \prod_{j \in B} P_j(t)$
$A \wedge B$ С («И-НЕ»)	Логический переход срабатывает, если на его вход поступили подпроцессы из совокупности А (для двух подпроцессов), А и В (для трех подпроцессов) и не поступил ни один подпроцесс из совокупности С	Для двух совокупностей подпроцессов $A \wedge \neg C$ С $P_{И-НЕ}(t) = \prod_{i \in A} P_i(t) \cdot \prod_{j \in C}[1 - P_j(t)]$ Для трех совокупностей подпроцессов $P_{И-НЕ}(t) = \prod_{i \in A} P_i(t) \cdot \prod_{k \in B} P_k(t) \cdot \prod_{j \in C}[1 - P_j(t)]$
$(A \vee B) \wedge \neg C$ («ИЛИ-НЕ»)	Логический переход срабатывает, если на его вход поступили подпроцессы из совокупности А или из совокупности В и не поступил ни один подпроцесс из совокупности С	$P_{ИЛИ-НЕ}(t) = \left\{1 - \prod_{i \in A \wedge B}[1 - P_i(t)]\right\} \cdot \times \prod_{k \in C}[1 - P_k(t)]$

Пусть между двумя логическими переходами по r-й траектории имеется $N^{(r)}$ позиций и дуг (рисунок 6.5), по каждой из которых процесс перемещается за конечное в общем случае случайное время $\tau_n^{(r)}, n = \overline{1, N^{(r)}}$.

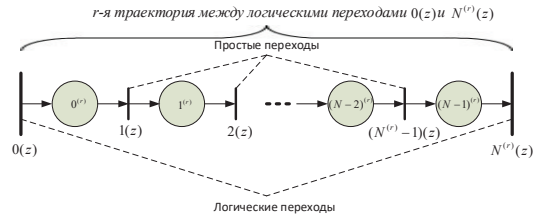


Рисунок 6.5 – Траектория перемещения процесса по ССПМ между двумя логическими переходами

Тогда математическое ожидание времени перемещения процесса из позиции $0^{(r)}(z)$ в переход $N^{(r)}(z)$ представляет собой сумму математических ожиданий независимых случайных величин и независимо от того, какие функции распределения имеют слагаемые времена, определяется из соотношения

$$\overline{\tau_{\Sigma}} = \sum_{n=1}^{N^{(r)}} \overline{\tau_{n-1,n}}\ , \tag{6.9}$$

где $\tau_{n-1,n}$ – математическое ожидание времени перемещения процесса между позицией с номером n-1 в переход с номером n.

В [35] была показана возможность использования экспоненциального приближения для выходящего из логического перехода потока, а также условия, при которых такое приближение приемлемо для ориентировочных оценок.

Так как времена между событиями являются положительно определенными случайными величинами, то плотность распределения вероятности для них всегда может быть аппроксимирована гамма-распределением или при целочисленных значениях параметра формы – распределениями Эрланга:

$$f_1(t) = \frac{t^{n_1-1}}{\overline{\tau_1}^{n_1}(n_1-1)!} \cdot e^{-\frac{t}{\overline{\tau_1}}} \text{ и } f_2(t) = \frac{t^{n_2-1}}{\overline{\tau_2}^{n_2}(n_2-1)!} \cdot e^{-\frac{t}{\overline{\tau_2}}}\ . \tag{6.10}$$

где n_1 и n_2 – параметры формы.

Следует отметить, что параметр формы в распределении Эрланга (потоки с таким распределением часто называют эрланговскими) в ССПМ соответствует количеству простых переходов между логическими переходами по рассматриваемой траектории.

Для эрланговских входящих потоков вероятность срабатывания перехода «И» (см. таблицу 6.1) удовлетворяет требованиям, предъявляемым к функциям распределения вероятностей случайных величин $P_{И}(0)=0,\ P_{И}(\infty)=1$. В связи с этим математическое ожидание времени срабатывания перехода рассчитывается по точным формулам, приведенным в таблице 6.2.

Вместе с тем, как показано в [35], что с достаточно высокой точностью для оценки математического ожидания времени срабатывания перехода с логикой «И» при небольших значениях параметра формы эрланговских входящих потоках можно использовать сравнительно простые формулы, приведенные также в таблице 6.2. Для сравнения в таблице 6.3 даны значения математического ожидания времени срабатывания логических переходов с логиками «И» и «ИЛИ», рассчитанные по точной и приближенной формулам.

Таблица 6.2 – Соотношения для расчета математического ожидания времени срабатывания переходов с логиками «И», «ИЛИ» и «И-ИЛИ»

Логика	Математическое ожидание времени срабатывания перехода	
	По точной формуле	По приближенной формуле
«И»	$\int_0^\infty \left\{ \frac{x^{n_1} \cdot e^{-\frac{t}{\overline{\tau_1}}}}{\overline{\tau_1}^{n_1} \cdot (n_1-1)!} \cdot \gamma\left(\frac{x}{\overline{\tau_2}^{n_2}}, n_2\right) + \frac{x^{n_2} \cdot e^{-\frac{t}{\overline{\tau_2}}}}{\overline{\tau_2}^{n_2} \cdot (n_2-1)!} \cdot \gamma\left(\frac{x}{\overline{\tau_1}^{n_1}}, n_1\right) \right\} dx$	$\overline{\tau_И} = \frac{\left(\overline{\tau_1} \cdot n_1\right)^2 + \overline{\tau_1} \cdot n_1 \cdot \overline{\tau_2} \cdot n_2 + \left(\overline{\tau_2} \cdot n_2\right)^2}{\overline{\tau_1} \cdot n_1 + \overline{\tau_2} \cdot n_2}$
«ИЛИ»	$\overline{\tau_{ИЛИ}} = \int_0^\infty \left\{ \frac{x^{n_1} \cdot e^{-\frac{t}{\overline{\tau_1}}}}{\overline{\tau_1}^{n_1} \cdot (n_1-1)!} \cdot \left[1 - \gamma\left(\frac{x}{\overline{\tau_2}^{n_2}}, n_2\right)\right] + \frac{x^{n_2} \cdot e^{-\frac{t}{\overline{\tau_2}}}}{\overline{\tau_2}^{n_2} \cdot (n_2-1)!} \cdot \left[1 - \gamma\left(\frac{x}{\overline{\tau_1}^{n_1}}, n_1\right)\right] \right\} dx$	$\overline{\tau_{ИЛИ}} = \frac{\overline{\tau_1} \cdot n_1 \cdot \overline{\tau_2} \cdot n_2}{\overline{\tau_1} \cdot n_1 + \overline{\tau_2} \cdot n_2}$
«И-ИЛИ»	$\overline{\tau_{И-ИЛИ}} = \int_0^\infty \left\{ \begin{array}{l} \left[\frac{x^{n_1} \cdot e^{-\frac{t}{\overline{\tau_1}}}}{\overline{\tau_1}^{n_1} \cdot (n_1-1)!} \cdot \gamma\left(\frac{x}{\overline{\tau_2}^{n_2}}, n_2\right) + \frac{x^{n_2} \cdot e^{-\frac{t}{\overline{\tau_2}}}}{\overline{\tau_2}^{n_2} \cdot (n_2-1)!} \cdot \gamma\left(\frac{x}{\overline{\tau_1}^{n_1}}, n_1\right) \right] \cdot \left[1 - \gamma\left(\frac{x}{\overline{\tau_3}^{n_3}}, n_3\right)\right] + \\ + \frac{x^{n_3} \cdot e^{-\frac{t}{\overline{\tau_3}}}}{\overline{\tau_3}^{n_3} \cdot (n_3-1)!} \cdot \left[1 - \gamma\left(\frac{x}{\overline{\tau_1}^{n_1}}, n_1\right) \cdot \gamma\left(\frac{x}{\overline{\tau_2}^{n_2}}, n_2\right)\right] \end{array} \right\} dx$	$\overline{\tau_{И-ИЛИ}} = \frac{\overline{\tau_1} \cdot n_1 \cdot \overline{\tau_2} \cdot n_2}{\overline{\tau_1} \cdot n_1 + \overline{\tau_2} \cdot n_2} + \frac{\left(\overline{\tau_1} \cdot n_1\right)^2 \cdot \overline{\tau_3} \cdot n_3}{\left(\overline{\tau_1} \cdot n_1 + \overline{\tau_3} \cdot n_3\right)^2} + \frac{\left(\overline{\tau_2} \cdot n_2\right)^2 \cdot \overline{\tau_3} \cdot n_3}{\left(\overline{\tau_2} \cdot n_2 + \overline{\tau_3} \cdot n_3\right)^2} - \frac{\overline{\tau_1} \cdot n_1 \cdot \overline{\tau_2} \cdot n_2 \cdot \overline{\tau_3} \cdot n_3}{\overline{\tau_1} \cdot n_1 + \overline{\tau_2} \cdot n_2 + \overline{\tau_3} \cdot n_3}$

Таблица 6.3 – Математические ожидания времени срабатывания логического перехода с логиками «И» и «ИЛИ», рассчитанные по точной и приближенной формулам

Параметры первого потока		Параметры второго потока		Результат расчета	
$\overline{\tau_1}$	n_1	$\overline{\tau_2}$	n_2	По точной формуле	По приближенной формуле
Для перехода с логикой «И»					
1	1	2	1	2.33	2.33
1	1	2	3	6.1	6.5
1	2	4	5	15	15.2
1	3	7	7	35	35.2
1	5	7	10	70	70,3
1	7	7	15	105	105,4
Для перехода с логикой «ИЛИ»					
1	1	2	1	0.67	0.67
1	1	2	3	0.96	0.86
1	2	4	5	2	1.82
1	3	7	7	3	2.83
1	5	7	10	5	4.67
1	7	7	15	7	6.6

Из таблицы 6.3 видно, что использование приближенной формулы вполне оправдано. Следует отметить, что с увеличением значений параметров формы закона Эрланга точность определения математического ожидания времени срабатывания логического перехода «И» возрастает. Однако это не всегда означает близость значений вероятностей срабатывания перехода.

Расчет самой вероятности проводится по формулам, приведенным в таблице 6.4, с использованием неполной γ-функции [92],

$$\gamma\{a \cdot t, n\} = \frac{1}{(n-1)!} \cdot \int_0^{at} x^{n-1} \cdot \exp(-x) \cdot dx \,. \qquad (6.11)$$

На рисунке 6.6 приведены для сравнения зависимости вероятностей срабатывания логического перехода с логикой «И», рассчитанные по формулам таблицы 6.4.

Таблица 6.4 – Соотношения для расчета вероятностей срабатывания переходов с логиками «И», «ИЛИ» и «И-ИЛИ»

Логика срабатывания перехода	Вероятность срабатывания перехода	
	По точной формуле	По приближенной формуле
«И»	$P_И(t) = \gamma(\frac{t}{\overline{\tau}_1}, n_1) \cdot \gamma(\frac{t}{\overline{\tau}_2}, n_2)$	$P_И(t) = 1 - exp\left[-\frac{t \cdot}{\overline{\tau}_И}\right]$
«ИЛИ»	$P_{ИЛИ}(t) = 1 - \left\{1 - \gamma(\frac{t}{\overline{\tau}_1}, n_1)\right\} \cdot \left\{1 - \gamma(\frac{t}{\overline{\tau}_2}, n_2)\right\}$	$P_{ИЛИ}(t) = 1 - exp\left[-\frac{t \cdot}{\overline{\tau}_{ИЛИ}}\right]$
«И-ИЛИ»	$P_{ИЛИ}(t) = 1 - \left\{1 - \gamma(\frac{t}{\overline{\tau}_1}, n_1) \cdot \gamma(\frac{t}{\overline{\tau}_2}, n_2)\right\} \cdot \left\{1 - \gamma(\frac{t}{\overline{\tau}_3}, n_3)\right\}$	$P_{И-ИЛИ}(t) = 1 - exp\left[-\frac{t \cdot}{\overline{\tau}_{И-ИЛИ}}\right]$

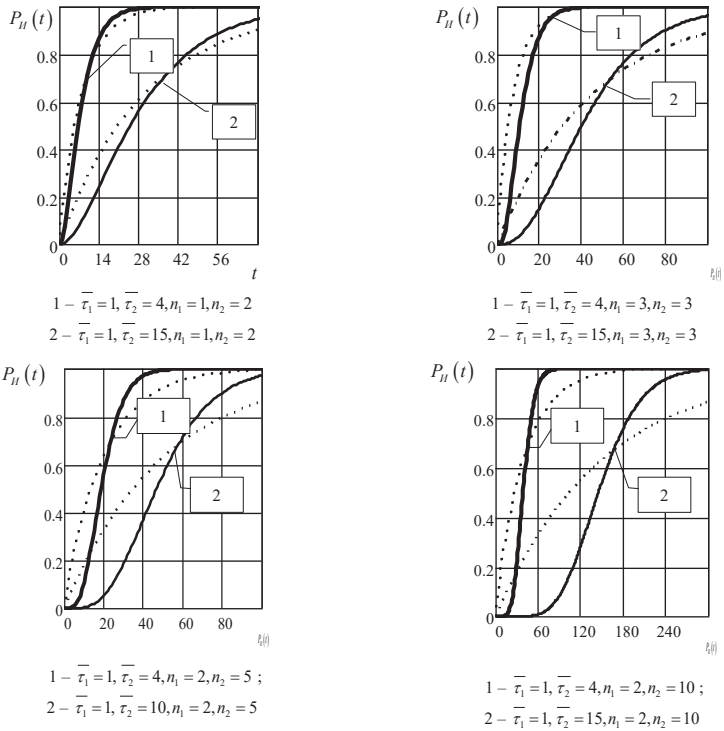


Рисунок 6.6 – Зависимости от времени вероятности срабатывания логического перехода «И», рассчитанные по точной (сплошная линия) и приближенной (пунктирная линия) формулам

По результатам сравнения зависимостей от времени вероятностей срабатывания перехода можно сделать некоторые выводы:

– при небольших значениях, параметров форм эрланговских входящих потоков, равных $1..5$, использование экспоненциального приближения для оценки вероятности срабатывания логического перехода не приводит к значительным ошибкам в оценке;

– увеличение значений параметров форм эрланговских входящих потоков может приводить к заметным ошибкам в оценке, особенно при начальных значениях времени t .

Аналогичные выводы имеют место и по логическим переходам с логикой «ИЛИ» и «И-ИЛИ», для которых вероятности срабатывания переходов тоже удовлетворяют требованиям, предъявляемым к функциям распределения времени их срабатывания.

Сравнительные значения математических ожиданий времен срабатывания переходов для логики «И-ИЛИ» приведены в таблицах 6.5.

Таблица 6.5 – Сравнительные значения математического ожидания времени срабатывания логического перехода «И-ИЛИ», рассчитанные по точной и приближенной формулам

Параметры первого потока		Параметры второго потока		Параметры третьего потока		Результат расчета	
$\overline{\tau_1}$	n_1	$\overline{\tau_2}$	n_2	$\overline{\tau_3}$	n_3	По точной формуле	По приближенной формуле
1	1	2	3	1	2	1.86	1.6
2	2	2	3	2	5	5.9	4.1
2	3	4	3	4	7	12	8
2	5	2	7	4	10	11.4	15

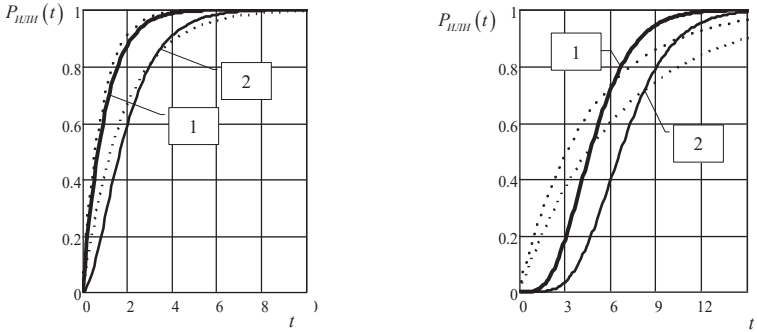
Примеры расчета зависимости от времени вероятности срабатывания логических переходов с логикой «ИЛИ» и «И-ИЛИ» – на рисунках 6.7 и 6.8.

По-иному рассчитываются вероятностно-временные характеристики срабатывания переходов с логиками «XOR», «И-НЕ» и «ИЛИ-НЕ», так как вероятности их срабатывания (см. таблица 6.1) не удовлетворяют требованиям, предъявляемым к функциям распределения вероятностей.

Логический переход «И-НЕ» применительно к двум входящим потокам срабатывает в том случае, если по первой из дуг процесс успел подойти к переходу, а по второй – нет, что соответствует условию «первый и не второй». Для расчета вероятности срабатывания такого перехода рассмотрим случайную величину $y = \tau_2 - \tau_1$, где τ_1 и τ_2 – длительности пауз между моментами поступления первого и второго потоков на переход. Если $y > 0$ к моменту времени t, то переход сработает.

Плотность вероятности величины y в соответствии с [26] определяется для двух входящих потоков следующим образом:

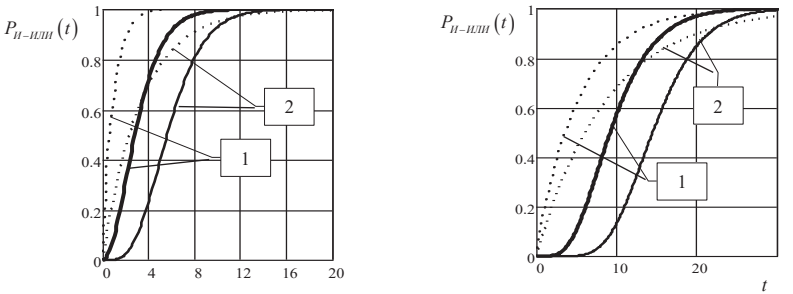
$$f_{И-НЕ}(y)=\begin{cases}\int\limits_{y}^{\infty} f_2(\tau_2)\cdot f_1(\tau_2-y)\cdot d\tau_2, \text{ если } y>0;\\ \int\limits_{0}^{\infty} f_2(\tau_2)\cdot f_1(\tau_2-y)\cdot d\tau_2, \text{ если } y<0.\end{cases} \qquad (6.12)$$



1 – $\overline{\tau}_1=1, \overline{\tau}_2=2, n_1=1, n_2=3$; 1 – $\overline{\tau}_1=1, \overline{\tau}_2=7, n_1=5, n_2=10$;

2 – $\overline{\tau}_1=1, \overline{\tau}_2=4, n_1=2, n_2=5$ 2 – $\overline{\tau}_1=1, \overline{\tau}_2=7, n_1=7, n_2=15$

Рисунок 6.7 – Зависимости от времени вероятности срабатывания логического перехода «ИЛИ», рассчитанные по точной (сплошная линия) и приближенной (пунктирная линия) формулам при различных параметрах входящих потоков



1 – $\overline{\tau}_1=1, \overline{\tau}_2=2, \overline{\tau}_3=3, n_1=1, n_2=3, n_3=2$ 1 – $\overline{\tau}_1=2, \overline{\tau}_2=3, \overline{\tau}_3=4, n_1=3, n_2=3, n_3=7$

2 – $\overline{\tau}_1=2, \overline{\tau}_2=2, \overline{\tau}_3=2, n_1=2, n_2=3, n_3=5$ 2 – $\overline{\tau}_1=2, \overline{\tau}_2=2, \overline{\tau}_3=4, n_1=5, n_2=7, n_3=10$

Рисунок 6.8 – Зависимости от времени вероятности срабатывания логического перехода «И-ИЛИ», рассчитанные по точной (сплошная линия) и приближенной (пунктирная линия) формулам при различных параметрах входящих потоков

В частности, для пуассоновских входящих потоков с учетом только положительно определенных значений времени формула (6.12) преобразуется к виду

$$f_{И-НЕ}(y) = \frac{1}{\overline{\tau_1} + \overline{\tau_2}} e^{-\frac{y}{\overline{\tau_2}}} + \frac{1}{\overline{\tau_1} + \overline{\tau_2}} \delta(y), \qquad (6.13)$$

где $\overline{\tau_1}$ и $\overline{\tau_2}$ – средние значения случайных времен τ_1 и τ_2;

$\delta(y)$ – дельта функция (функция Дирака).

Тогда усредненная вероятность того, что при каждой реализации будет выполняться неравенство $y > 0$, рассчитывается следующим образом:

$$\overline{p_y} = \overline{\tau_2} \Big/ \left(\overline{\tau_1} + \overline{\tau_2} \right). \qquad (6.14)$$

Математическое ожидание времени и вероятность срабатывания перехода с логикой «И-НЕ» определяется по формулам, приведенным в таблице 6.6.

Таблица 6.6 – Соотношения для расчета вероятности и математического ожидания времени срабатывания перехода с логикой «И-НЕ» при эрланговских входящих потоках[27]

Количество входящих потоков	Формулы для расчета математического ожидания времени срабатывания перехода	Формула для расчета вероятности срабатывания перехода	
		По точной формуле	По приближенной формуле
2 с параметрами n_1 и n_2	$\tau^{(2)}_{И-НЕ} = \overline{\tau_1} \cdot n_1 \cdot \left(1 + \frac{\overline{\tau_1} \cdot n_1}{\overline{\tau_2} \cdot n_2} \right)$	$P^{(2)}_{И-НЕ}(t) = \gamma\left(\frac{t}{\overline{\tau_1}}, n_1 \right) \cdot \left[1 - \gamma\left(\frac{t}{\overline{\tau_2}}, n_2 \right) \right]$	$P^{(2)}_{И-НЕ}(t) = 1 - exp\left(-t \Big/ \overline{\tau^{(2)}_{И-НЕ}} \right)$
3 с параметрами n_1, n_2 и n_3	$\tau^{(3)}_{И-НЕ} = \overline{\tau^{(2)}_{И-НЕ}} \cdot \left(1 + \frac{\overline{\tau^{(2)}_{И-НЕ}}}{\overline{\tau_3} \cdot n_3} \right)$	$P^{(2)}_{И-НЕ}(t) = \gamma\left(\frac{t}{\overline{\tau_1}}, n_1 \right) \cdot \gamma\left(\frac{t}{\overline{\tau_2}}, n_2 \right) \cdot \left[1 - \gamma\left(\frac{t}{\overline{\tau_3}}, n_3 \right) \right]$	$P^{(3)}_{И-НЕ}(t) = 1 - exp\left(-t \Big/ \overline{\tau^{(3)}_{И-НЕ}} \right)$

Логический переход «ИЛИ-НЕ» имеет место применительно к не менее, чем трем входящим потокам, при этом он срабатывает, если или первый поток, или второй поток подойдет к переходу, а третий нет. Математическое ожидание времени срабатывания перехода рассчитывается с использованием того же подхода, что был использован для перехода с логикой «И-НЕ».

Формулы для расчета математического ожидания времени и вероятности срабатывания переходов с логикой «ИЛИ-НЕ» приведены соответственно в таблице 6.7.

[27] Для пуассоновскиз входдящих потоков $n_1 = n_2 = n_3 = 1$

Таблица 6.7 – Соотношения для расчета вероятности и математического ожидания времени срабатывания перехода с логикой «ИЛИ-НЕ»

Количество входящих потоков	Формулы для расчета математического ожидания времени срабатывания перехода	Формула для расчета вероятности срабатывания перехода	
		По точной формуле	По приближенной формуле
3 с параметрами n_1, n_2 и n_3	$\overline{\tau_{ИЛИ-НЕ}} = \overline{\tau_{ИЛИ}^{(1,2)}} \cdot \left(1 + \frac{\overline{\tau_{ИЛИ}^{(1,2)}}}{\overline{\tau_3} \cdot n_3}\right)$ где $\overline{\tau_{Или}^{(1,2)}} = \cdot \frac{\overline{\tau_1} \cdot n_1 \cdot \overline{\tau_2} \cdot n_2}{\overline{\tau_1} \cdot n_1 + \overline{\tau_2} \cdot n_2}$	$P_{ИЛИ-НЕ}(t) = \left\{1 - \left[1 - \gamma\left(\frac{t}{\overline{\tau_1}}, n_1\right)\right] \cdot \left[1 - \gamma\left(\frac{t}{\overline{\tau_2}}, n_2\right)\right]\right\} \times$ $\times \left[1 - \gamma\left(\frac{t}{\overline{\tau_3}}, n_3\right)\right]$	$P_{ИЛИ-НЕ}(t) = 1 - exp\left(-t/\overline{\tau_{ИЛИ-НЕ}}\right)$

Вероятность того, что очередное событие первого потока поступит на переход при отсутствии события второго потока, рассчитывается по формуле:

$$P_{XOR}(t) = P_1(t) \cdot [1 - P_2(t)] + P_2(t) \cdot [1 - P_1(t)]. \quad (6.15)$$

Переход срабатывает в том случае, когда время t_1 поступления первого потока меньше времени t_2 поступления второго потока, то есть выполняется условие $t_1 < t_2$.

Условная интенсивность поступления на переход первого потока, при условии того, что второй поток не поступил, определяется из соотношения:

$$\mu_1 = \frac{1}{\overline{\tau_1}} \cdot \int_0^\infty \int_0^\infty f_2(y + \tau_1) \cdot f_1(\tau_1) \cdot d\tau_1 dy. \quad (6.16)$$

Условная интенсивность поступления на переход второго потока, при условии того, что первый поток не поступил, определяется из аналогичного соотношения:

$$\mu_2 = \frac{1}{\overline{\tau_2}} \cdot \int_0^\infty \int_0^\infty f_1(y + \tau_2) \cdot f_2(\tau_2) \cdot d\tau_2 dy. \quad (6.17)$$

Для экспоненциального приближения:

$$\mu_1 = \left(1/\overline{\tau_1}\right) \cdot \left[\overline{\tau_1} / \left(\overline{\tau_1} + \overline{\tau_2}\right)\right] = 1 / \left(\overline{\tau_1} + \overline{\tau_2}\right) \quad \text{и}$$

$$\mu_2 = \frac{1}{\overline{\tau_2}} \cdot \frac{\overline{\tau_2}}{\overline{\tau_1} + \overline{\tau_2}} = \frac{1}{\overline{\tau_1} + \overline{\tau_2}}. \quad (6.18)$$

Поскольку переход срабатывает при поступлении любого из двух потоков, суммарная интенсивность такого срабатывания составляет величину:

$$\mu_{XOR} = \mu_1 + \mu_2 = 2 / \left(\overline{\tau_1} + \overline{\tau_2}\right) \text{ и } \overline{\tau_{XOR}} = 1/\mu_{XOR} = \left(\overline{\tau_1} + \overline{\tau_2}\right) / 1. \quad (6.19)$$

Для двух пуассоновских входящих потоков вероятность срабатывания перехода рассчитывается по формуле:

$$P_{XOR}(t) = \left[1 - e^{-t/\overline{\tau_1}}\right] \cdot e^{-t/\overline{\tau_2}} + e^{-t/\overline{\tau_1}} \cdot \left[1 - e^{-t/\overline{\tau_2}}\right], \quad (6.20)$$

а для эрланговских входящих потоков – по формуле:

$$P_{XOR}(t) = \gamma(t/\overline{\tau_1}, n_1) \cdot \left\{1 - \gamma(t/\overline{\tau_2}, n_2)\right\} + \gamma(t/\overline{\tau_2}, n_2)) \cdot \left\{1 - \gamma(t/\overline{\tau_1}, n_1)\right\}. \qquad (6.21)$$

При этом в случае эрланговских потоков для расчета математического ожидания времени срабатывания перехода с логикой XOR может быть использовано приближенное соотношение:

$$\overline{\tau_{XOR}} = \left(\overline{\tau_1} \cdot n_1 + \overline{\tau_2} \cdot n_2\right)/2. \qquad (6.22)$$

Примеры применения составных сетей Петри-Маркова на основе марковских процессов с логическими переходами с пропозициональной логикой срабатывания приведены в разделе 6.4.

6.3. Составные сети Петри-Маркова на основе марковских процессов с предикатными условиями выполнения

Условия срабатывания логических переходов в сети Петри-Маркова могут формально вводиться не только с использованием аппарата пропозициональной логики, но с использованием операций с предикатами второго порядка [55, 56]. Если обозначить позиции и переходы СПМ через x_i и x_j, то бинарный предикат, отражающий условие выполнения действия между этими элементами, обозначается как $R\left(x_i, x_j\right)$.

В практике моделирования процессов реализации угроз могут найти применение следующие операции с предикатами:

$\rightarrow \left\{R(x_i, x_j), R(x_j, x_k)\right\}$ – операция последовательного выполнения или импликации (в общем случае взаимосвязанных действий по разным ветвям сети операция импликации может иметь вид $\rightarrow \left\{R(x_i, x_j), R(x_k, x_z)\right\}$);

$\xrightarrow{0}(R(x_i, x_j), R^0(x_k, x_j))$ – операция «отрицательной импликации», когда действия между элементами СПМ x_i и x_j происходят, а между x_k и x_j – нет;

$XOR(R(x_i, x_j), R(x_k, x_z)) = R(x_i, x_j) \oplus R(x_k, x_z)$ – операция эксклюзивного выбора (знак «XOR», исключающее «ИЛИ», при этом возможно ($i \equiv k$);

$OR(R(x_i, x_j), R(x_k, x_z)) = R(x_i, x_j) \vee R(x_k, x_z)$ – операция неэксклюзивного выбора (знак «OR» – логическая операция «ИЛИ», при этом возможно $i \equiv k$);

$AND(R(x_i, x_j), R(x_k, x_z)) = R(x_i, x_j) \wedge R(x_k, x_z)$ – операция параллельного выполнения (знак «AND» – логическая операция «И», при этом возможно $i \equiv k$);

$\Omega(R(x_i, x_j), R(x_k, x_z), n_\Omega)$ – операция цикла (знак «Ω») с повторением n_Ω раз (при этом возможно $i \equiv k$) или для связанных операций $\Omega\left\{\left[R(x_i, x_j), n_\Omega\right] \wedge \left[R(x_k, x_z), \left(n_\Omega - 1\right)\right]\right\}$.

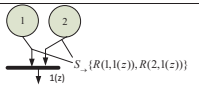
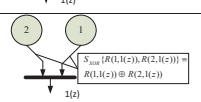
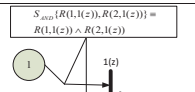
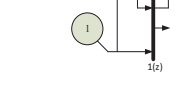
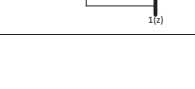
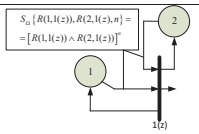
Краткое описание предикатных операций применительно к СПМ и их соответствие логическим переходам приведено в таблице 6.8.

Следует отметить, что если операции «отрицательной импликации», «эксклюзивного выбора», «неэксклюзивного выбора» и параллельного выполнения (операция AND) соответствуют ранее рассмотренным логикам срабатывания переходов СПМ, то существенно отличающимися являются операции импликации и цикла, для которых требуются пояснения.

Пусть в СПМ имеется предикатная операция импликации, определяющая следующее правило срабатывания одного из переходов: процесс должен подойти к переходу за время t сначала по первой, а затем по второй дуге.

Если вероятности перемещения процесса по первой и второй дугам равны соответственно $P_1(t)$ и $P_2(t)$, а вероятность того, что по второй дуге процесс придет позже, чем по первой – $P(t_1 < t_2)$, где t_1 и t_2 – случайные

Таблица 6.8 – Основные предикатные операции в сети Петри-Маркова

Наименование предикатной операции	Графическое представление предикатной операции	Условия срабатывания перехода	Соотношение для расчета математического ожидания времени срабатывания перехода при экспоненциальных распределениях
1	2	3	4
Импликация		Переход 1(z) срабатывает, если сначала состоялось перемещение процесса по дуге (1,1(z)), а затем по дуге (2,1(z)) со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$	Условие в виде пропозициональной логики отсутствует. $\overline{\tau_{\to}} = \frac{\overline{\tau_1}^2 + \overline{\tau_1}\cdot\overline{\tau_2} + \overline{\tau_2}^2}{\overline{\tau_2}}$
Отрицательная импликация		Переход 1(z) срабатывает, если состоялось перемещение процесса по дуге (1(a),1(z)) и не произошло по дуге (2(a),1(z)) со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$	Соответствует условию «И-НЕ» (см. таблицу 6.2). $\overline{\tau_{\not\to}} = \overline{\tau_1}\cdot\left(1+\frac{\overline{\tau_1}}{\overline{\tau_2}}\right)$
Эксклюзивный выбор («XOR»)		Переход 1(z) срабатывает, если состоялось перемещение процесса или только по дуге (1,1(z)), или только по дуге (2,1(z)) со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$	Соответствует условию «XOR» (см. раздел 2.2) $\overline{\tau_{XOR}} = \frac{\overline{\tau_1}+\overline{\tau_2}}{2}$
Неэксклюзивный выбор («OR»)		Переход 1(z) срабатывает, если состоялось перемещение процесса хотя бы по одной из дуг (1,1(z)) или (2,1(z)) со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$	Соответствует условию «ИЛИ» (см. таблицу 6.2) $\overline{\tau_{OR}} = \frac{\overline{\tau_1}\cdot\overline{\tau_2}}{\overline{\tau_1}+\overline{\tau_2}}$
Параллельное выполнение («AND»)		Переход 1(z) срабатывает, если состоялось перемещение процесса и по дуге (1,1(z)), и по дуге (2,1(z)) со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$	Соответствует условию «И» (см. таблицу 6.2) $\overline{\tau_{\wedge}} = \frac{\overline{\tau_1}^2 + \overline{\tau_1}\cdot\overline{\tau_2} + \overline{\tau_2}^2}{\overline{\tau_1}+\overline{\tau_2}}$
Операция цикла по одной дуге		Переход 1(z) срабатывает, если состоялось перемещение по дуге из позиции1(a) в переход 1(z) и n раз перемещение процесса из позиции 2(a) в переход 1(z) и обратно со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$. Примечание: обозначение $\Omega(2)$ означает, что операция цикла применяется к дуге, исходящей из второй позиции	Условие в виде пропозициональной логики отсутствует. $\overline{\tau_{\Omega}} = \frac{\overline{\tau_1}^2 + \overline{\tau_1}\cdot\overline{\tau_2}\cdot n + \overline{\tau_2}^2\cdot n^2}{\overline{\tau_1} + n\cdot\overline{\tau_2}}$,
Операция цикла по двум дугам		Переход 1(z) срабатывает, если состоялось n раз перемещение процесса по обеим дугам из позиции1(a) в переход 1(z) и перемещение из позиции 2(a) в переход 1(z) и обратно $n-1$ раз со средними временами перемещения $\overline{\tau_1}$ и $\overline{\tau_2}$	Условие в виде пропозициональной логики отсутствует. $\overline{\tau_{\Omega}} = \frac{\overline{\tau_1}^2 + \overline{\tau_1}\cdot\overline{\tau_2} + \overline{\tau_2}^2}{\overline{\tau_1}+\overline{\tau_2}}\cdot n$,

времена подхода процесса к переходу по первой и второй дуге соответственно, то это соответствует предикатной операции «импликация» (см. таблица 6.8).

В отличие от логики «И-НЕ» переход срабатывает, когда по второй дуге процесс тоже подойдет к переходу, но позже, чем по первой дуге (при логике «И-НЕ» переход сработает, когда по второй дуге процесс вообще не подойдет к переходу).

Тогда математическое ожидание времени срабатывания перехода может быть определено путем расчета математического ожидания паузы между событиями потока, полученного путем прореживания с вероятностью $P(t_1 < t_2)$ потока, соответствующего логике срабатывания «И», то есть в соответствии с теорией прореживания рекуррентных потоков следующим образом:

$$\overline{\tau_{\rightarrow}} = \frac{\overline{\tau_1}^2 + \overline{\tau_1} \cdot \overline{\tau_2} + \overline{\tau_2}^2}{(\overline{\tau_1} + \overline{\tau_2}) \cdot P(t_1 < t_2)}. \qquad (6.23)$$

Вероятность того, что при каждом подходе процесса к переходу по обеим дугам будет выполняться неравенство $t_1 < t_2$, рассчитывается по формуле

$$P(t_1 < t_2) = \int_0^{\infty}\int_0^{\infty} f_2(y+\tau) \cdot f_2(y) \cdot d\tau \cdot dy = \frac{\overline{\tau_2}}{\overline{\tau_1} + \overline{\tau_2}}. \qquad (6.24)$$

В результате

$$\overline{\tau_{\rightarrow}} = \frac{\overline{\tau_1}^2 + \overline{\tau_1} \cdot \overline{\tau_2} + \overline{\tau_2}^2}{\overline{\tau_2}}. \qquad (6.25)$$

Длительность операции цикла определяется суммарным временем перемещения процесса к переходу и от перехода к инцидентной позиции при выполнении цикла n раз.

Приведенные соотношения для среднего времени срабатывания переходов в соответствии с операциями над предикатами второго порядка позволяют существенно расширить возможности моделирования процессов реализации угроз безопасности информации с использованием СПМ на основе составных марковских процессов.

6.4. Примеры применения аппарата составных сетей Петри-Маркова на основе марковских процессов для моделирования угроз безопасности информации

При моделировании угроз безопасности информации в ИС с использованием аппарата ССПМ на основе марковских процессов необходимо определить состав процедур и функций и последовательность их выполнения, имеющих место при реализации угроз. В описательных моделях такая детальность описания моделируемых процессов, как правило, не предусматривается. Для такого описания более подходят так называемые функциональные модели, широко применяемые в иных областях исследований. Так, например, весьма развитой сегодня считается методология

функционального моделирования DFD – методология построения диаграмм потока данных (Data flow diagramming, DFD). Она применяется при проведении обследований предприятий в рамках проектов управленческого консалтинга, а также в проектах автоматизации крупных объектов при экспресс-обследовании. Нотация диаграмм потоков данных позволяет отображать на диаграмме как шаги процесса, так и поток, например, документов и команд управления. Ее развитием стала методология IDEF0 (Icam DEFinition), которая является основной частью программы ICAM (Integrated Computer Aided Manufacturing – Интеграция компьютерных и промышленных технологий), проводившейся по инициативе ВВС США. В IDEF0 рассматриваются логические отношения между работами, а не их временные характеристики. Функциональная модель IDEF0 – это набор блоков, каждый из которых представляет собой «черный ящик» со входами и выходами, управлением и механизмами, которые детализируются (декомпозируются) до необходимого уровня.

Несмотря на достаточно широкое применение в иных областях исследований, эта методология для моделирования угроз безопасности информации пока применяется весьма ограниченно. Вместе с тем идея разработки функциональных моделей для описания процесов реализации угроз плодотворна, но такие модели отличаются от разрабываемых в соответствии с методологией IDEF0. Они должны позволять описывать процессы реализации угроз с такой детальностью, которая дает возможность непосредственно переходить к определению состава позиций ССПМ и выполняемых функций для разработки адекватных ССПМ в ходе моделирования угроз. Кроме того, они должны содержать сведения, касающиеся логических условий протекания процесса реализации угрозы, и временные характеристики выполнения функций.

Такие функциональные модели угроз могут строиться двумя путями.

Первый путь заключается в том, что выполняемые при реализации угрозы действия привязываются к аппаратным или программным средствам, которые задействованы при их реализации. В этом случае формально функциональная модель процесса реализации каждой угрозы представляется в виде совокупности пяти множеств:

$$\Psi_u = \left\{ A_u, D_u, M(D_u), L(D_u) \right\}, u = \overline{1,U}, \tag{6.26}$$

где A_u – множество аппаратных и программных элементов, которые задействованы при реализации u-й угрозы;

D_u – множество функций (действий) $d_u^{(k)} \in D_u, k = \overline{1, \aleph(D_u)}$, которые нужно выполнить для реализации u-й угрозы, $\aleph(D_u)$ – мощность множества D_u, при этом имеется взаимосвязь между каждым действием и соответствующим ему аппаратным или программным элементом;

$M(D_u)$ – матрица взаимосвязей действий (функций) $d_u^{(k)}$, отражающая последовательность выполнения;

$L(D_u)$ – совокупность условий успешного выполнения действий (функций) $d_u^{(k)}$, при которых возможна реализация u-й угрозы.

Такие модели разрабатываются чаще всего для угроз, реализуемых на сетевом, системном или прикладном системно-техническом уровне.

Второй путь имеет место, когда действия при реализации угрозы выполняются одним и тем же элементом (например, вредоносной программой). В этом случае нет необходимости в функциональной модели указывать задействованные программные или аппаратные элементы, при этом используется традиционный подход, когда модель представляет совокупность выполняемых в определенном порядке функций. Формально функциональная модель в этом случае представляет собой совокупность лишь трех множеств:

$$\Psi_u = \{D_u, M(D_u), L(D_u)\}, u = \overline{1,U}. \quad (6.27)$$

Примеры разработанных с использованием данного подхода функциональных моделей некоторых угроз безопасности информации в ИС приведены в таблице 6.9. В приведенных примерах отсутствуют только сроки выполнения каждого действия, которые могут различными для разных ИС.

Следует отметить, что в приведенных функциональных моделях матрицы взаимосвязей действий все являются диагональными, что объясняется принятой нумерацией действий и отсутствием возвратных действий (с возвратом к предыдущим состояниям).

В общем случае возможны и такие взаимосвязи действий, которые приводят к недиагональному виду матрицы $M(D_u)$.

Таким образом, порядок формирования функциональных моделей угроз безопасности информации в ИС состоит в следующем:

1) определяется перечень аппаратных или программных элементов ИС, которые используются в процессе реализации угрозы;

2) определяется перечень, порядок и при необходимости сроки выполнения действий (функций) в процессе реализации угрозы;

3) формируется ориентированный граф, иллюстрирующий функциональную модель угроз;

4) устанавливаются условия, если таковые имеются, при которых выполняется каждое действие;

5) формируется при необходимости матрица взаимосвязей действий.

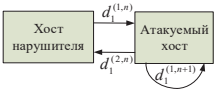
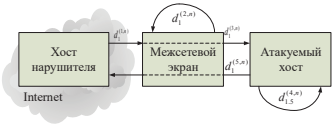
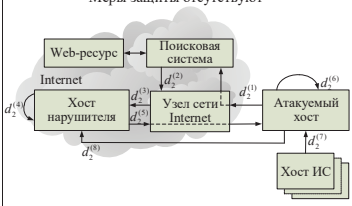
Далее на основе функциональных моделей разрабатываются соответствующие ССПМ. Порядок разработки состоит в следующем:

1) определяются начальные (или начальное), промежуточные и конечные (или конечное) состояния, в которых может находиться моделируемый процесс реализации угрозы по мере выполнения действий, определенных в функциональной модели, и таким образом формируется перечень позиций ССПМ;

2) определяются логические переходы. К таким переходам относятся все переходы, содержащие более одной входящей дуги (выходящих дуг как у

логического, так и у простого перехода может быть более одной). Указывается место каждого логического перехода (после каких состояний процесса должен выполняться переход) и логика его срабатывания или предикатная операция;

Таблица 6.9 – Примеры функциональных моделей процессов реализации некоторых угроз безопасности в информационных системах в условиях отсутствия мер защиты

Наименование угрозы и ее индекс u	Графическое представление совокупности действий по реализации угрозы	Идентификатор действия	Содержание действия (функции)
Угроза атаки «Отказ в обслуживании» из внешней сети. Вариант атаки – «шторм TCP-запросов» («syn-flooding»), $u=1$	Меры защиты отсутствуют  $M(D_1)=$ (1 → 2, 2 → 1) В левом столбце матрицы указаны номера первичных действий, в верхней строке – номера последующих действий. Условия сводятся к ограничению количества принимаемых пакетов по одному порту – не более n раз.	$d_1^{(1,n)}$	Хост нарушителя направляет через маршрутизатор для атакуемого хоста TCP-запрос на соединение. После получения квитанции от атакуемого хоста нарушитель вместо передачи ответной квитанции о начале связи передает на атакуемый хост еще один пакет с запросом, и это повторяется n раз
		$d_1^{(2,n)}$	Атакуемый хост направляет пакет с квитанцией согласия на установление виртуального канала, записывает данные о запросившем соединение хосте в буфер и ждет ответной квитанции. В ходе атаки действие повторяется n раз пока буфер не переполнится и ОС атакуемого компьютера не зависнет
		$d_1^{(3,n)}$	При превышении количества пакетов с запросами величины n буфер атакуемого хоста переполняется и операционная система зависает система атакуемого компьютера не сможет помещать сведения о запросившем соединение хосте в буфер
	Установлен межсетевой экран, введено условие: предельно допустимое количество пакетов, принимаемых по одному порту  $M(D_1)=$ (1 → 1, 2 → 1, 3 → 1, 4 → 1) 1 2 3 4 5	$d_1^{(1,n)}$	Атакуемый хост направляет пакет с квитанцией согласия на установление виртуального канала, записывает данные о запросившем соединение компьютере в буфер и ждет ответной квитанции с сообщением о начале сеанса связи. В ходе атаки действие повторяется n раз пока операционная система атакуемого компьютера не сможет помещать сведения о запросившем соединение хосте в буфер
		$d_1^{(2,n)}$	При превышении количества пакетов с запросами величины n буфер атакуемого хоста переполняется и операционная система зависает. Компьютер перестает участвовать в системе электронного документооборота
		$d_1^{(3,n)}$	Хост нарушителя направляет через маршрутизатор для атакуемого хоста TCP-запрос на соединение. Количество таких запросов может составлять величину n. Остальные пакеты отбрасываются
		$d_1^{(4,n)}$	TCP-пакет с запросом на соединение попадает на межсетевой экран. Процедура повторяется n раз
		$d_1^{(5,n)}$	Атакуемый хост направляет пакет с квитанцией согласия на установление виртуального канала, записывает данные о запросившем соединение компьютере в буфер и ждет ответной квитанции с сообщением о начале сеанса связи. В ходе атаки действие повторяется n раз пока операционная
Угроза копирования нужного нарушителю файла путем запуска скрипта, внедряемого с использованием файлов cookies, $u=2$	Меры защиты отсутствуют  	$d_2^{(1)}$	Атакуемый хост отправил через некий транспортный узел на поисковую систему запрос на Internet-ресурс. По запросу осуществляется поиск запрашиваемого ресурса сети Internet
		$d_2^{(2)}$	Поисковик Internet формирует файл cookie, который посылается на атакуемый хост
		$d_2^{(3)}$	Файл cookie перехватывается хостом нарушителя с помощью, например, «сниффера», установленного предварительно на узле Internet, расположенного по маршруту передачи запроса ресурса атакуемым хостом, и блокируется передача файла cookie на атакуемый хост

Наименование угрозы и ее индекс u	Графическое представление совокупности действий по реализации угрозы	Идентификатор действия	Содержание действия (функции)
	Условия выполнения действий не вводятся: множество $L(D_2)$ пустое $M(D_2)=\begin{array}{c\|cccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \hline 1 & & 1 & & & & & & \\ 2 & & & 1 & & & & & \\ 3 & & & & 1 & & & & \\ 4 & & & & & 1 & & & \\ 5 & & & & & & 1 & & \\ 6 & & & & & & & 1 & \\ 7 & & & & & & & & 1 \end{array}$	$d_2^{(4)}$	В файл cookie вставляется скрипт с вредоносной программой и подготавливается к передаче на атакуемый хост
		$d_2^{(5)}$	Файл cookie передается на атакуемый хост
		$d_2^{(6)}$	Браузер атакуемого хоста, записывая файл cookie, запускает вредоносную программу
		$d_2^{(7)}$	С одного из хостов ИС на атакуемый хост передается файл с нужной нарушителю информацией
		$d_2^{(8)}$	Вредоносной программой файл копируется и передается на хост нарушителя
	Установлено на атакуемом хосте средство анализа скриптов в web-браузере для выявления наличия вредоносной программы База данных сигнатур и правил для эвристического анализа ИС Модуль сканирования Модуль эвристического анализа Консоль администратора безопасности Web-ресурс Поисковая система Internet Хост нарушителя Узел сети Internet Атакуемый хост Хост ИС $M(D_2)=\begin{array}{c\|ccccccccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ \hline 1 & & 1 & & & & & & & & & & & & & \\ 2 & & & 1 & & & & & & & & & & & & \\ 3 & & & & 1 & & & & & & & & & & & \\ 4 & & & & & 1 & & & & & & & & & & \\ 5 & & & & & & 1 & & & & & & & & & \\ 6 & & & & & & & 1 & & & & & & & & \\ 7 & & & & & & & & 1 & & & & & & & \\ 8 & & & & & & & & & 1 & & & & & & \\ 9 & & & & & & & & & & 1 & & & & & \\ 10 & & & & & & & & & & & 1 & & & & \\ 11 & & & & & & & & & & & & 1 & & & \\ 12 & & & & & & & & & & & & & 1 & & \\ 13 & & & & & & & & & & & & & & 1 & \\ 14 & & & & & & & & & & & & & & & 1 \end{array}$ Условия выполнения действий не вводятся: множество $L(D_2)$ – пустое	$d_2^{(1)}$	Атакуемый хост отправил на web-сервер через маршрутизатор ИС запрос на Internet-ресурс. Запрос переправляется на поисковую машину в сеть Internet
		$d_2^{(2)}$	Поисковик Internet формирует файл cookie и осуществляет информационный поиск по запросу. Файл cookie посылается на атакуемый хост
		$d_2^{(3)}$	Файл cookie перехватывается хостом нарушителя с помощью, например, «сниффера», установленного предварительно на узле Internet, расположенного по маршруту передачи запроса ресурса атакуемым хостом, и блокируется передача файла cookie на атакуемый хост
		$d_2^{(4)}$	В файл cookie вставляется скрипт с вредоносной программой и подготавливается к передаче на атакуемый хост
		$d_2^{(5)}$	Файл cookie передается на атакуемый хост
		$d_2^{(6)}$	Браузер атакуемого хоста открывает файл cookie и запускает вредоносную программу
		$d_2^{(7)}$	С хостов ИС на атакуемый хост передаются файлы, среди которых есть те, которые представляют интерес для нарушителя
		$d_2^{(8)}$	Модуль сканирования сканирует все потенциальные объекты воздействия, в том числе атакуемый хост
		$d_2^{(9)}$	Модуль эвристического анализа собирает данные, необходимые для выявления вредоносной программы
		$d_2^{(10)}$	В модуле сканирования осуществляется сравнение выявленных фрагментов кода с сигнатурами вредоносных программ из базы сигнатур
		$d_2^{(11)}$	Модуль эвристического анализа сравнивает полученные данные с признаками наличия ВП
		$d_2^{(12)}$	При обнаружении модулем сканирования вредоносной программы сигнал об этом передается на консоль администратора безопасности
		$d_2^{(13)}$	При обнаружении модулем эвристического анализа вредоносной программы сигнал об этом передается на консоль администратора безопасности
		$d_2^{(14)}$	Администратором безопасности блокируется выполнение вредоносной программы
		$d_2^{(15)}$	При отсутствии команды с консоли администратора о блокировании вредоносной программы осуществляется копирования нужных нарушителю файлов и их передача на хост нарушителя

3) определяются для каждого участка ССПМ между начальным состоянием и первым логическим переходом, между логическими переходами и, наконец, между последним логическим переходом и конечным состоянием все простые переходы;

4) для каждого простого и логического перехода формируются входные и выходные дуги, соединяющие переход с инцидентными входными и выходными позициями ССПМ;

5) осуществляется разметка ССПМ путем помещения маркера в начальную позицию (или начальные позиции, если их несколько).

Разбиение ССПМ на участки позволяет рассчитать время, в течение которого моделируемый процесс достигнет логического перехода на данном участке, а затем времена срабатывания логических переходов и, наконец, время выполнения процесса в целом.

Примеры ССПМ применительно к угрозам, функциональные модели реализации которых даны в таблице 6.9, приведены в таблице 6.10. Там же приведены соотношения для расчета математического ожидания времени и вероятности реализации угроз без мер и в условиях применения мер защиты.

6.5. Составные сети Петри-Маркова на основе полумарковских процессов и примеры их применения для моделирования процессов реализации угроз безопасности информации

Аппарат полумарковских процессов (ПМП) сегодня широко применяется в теории надежности, при анализе и построении систем массового обслуживания. Находит он применение также и при моделировании угроз безопасности информации в информационных системах.

Однако в традиционном представлении этот аппарат имеет определенные ограничения, не позволяющие создавать целый ряд важных для практики моделей.

К основным из них относятся:

невозможность учета логических условий[28], которые могут иметь место в моделируемых процессах;

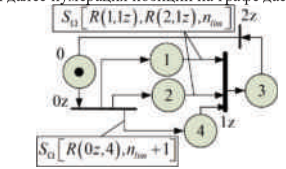
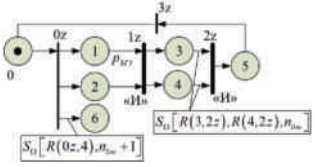
сложность моделирования циклических операций с задаваемыми условиями завершения их выполнения (например, по количеству выполненных циклов);

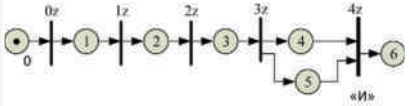
наличие только одного начального состояния процесса, в то время как могут иметь место несколько состояний, из которых может начинаться процесс с определенными (возможно случайными) задержками во времени.

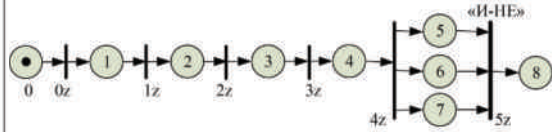
В [35] предложены некоторые расширения аппарата ПМП, связанные с применением составных СПМ на основе полумарковских процессов, позволяющие значительно расширить его моделирующие возможности. Под составной сетью Петри-Маркова на основе ПМП здесь понимается процесс, фрагменты которого объединены между собой логическими переходами и являются полумарковскими процессами. Пример составного ПМП приведен на рисунке 6.9.

[28] За исключением условия, определяющего срабатывания перехода и состоящего в том, что количество фишек (меток) в каждой входящей позиции, *инцидентной переходу,* должно быть не меньше количества входящих дуг из этой позиции

Таблица 6.10 – Модели реализации угроз с использованием аппарата сетей Петри-Маркова и соотношения для расчета среднего времени такой реализации

Наименование и идентификатор угрозы	Графическое представление модели реализации угрозы с использованием сети Петри-Маркова	Соотношение для расчета среднего времени и вероятности реализации угрозы
Угроза атаки «syn-flooding», ($u=1$)	Без мер защиты (здесь и далее нумерация позиций на графе дается без буквы «а»)  0 – начальное состояние процесса, нарушитель начал отправку запроса на соединение с атакуемым хостом; 1 – на атакуемый хост поступил TCP-запрос на соединение с хостом нарушителя; 2 – количество запросов на соединение не превышает допустимое значение n_{lim}; 3 – атакуемый хост начал передачу квитанцию на хост нарушителя о готовности к взаимодействию; 4 – количество запросов превысило допустимый уровень, буфер, содержащий записи о хостах, приславших запросы на соединение, переполнился, операционная система «зависла», атака реализована; 0(z) – передача запроса с хоста нарушителя на атакуемый хост; 1(z) – логический переход, срабатывающий при условии, что количество запросов на соединение не превышает предельное значение n_{lim}; 2(z) – ответ-квитанция отправляется на хост нарушителя	$\overline{\tau_u^{(0)}}=\overline{\tau_{0,0}}\cdot(n_{lim}+1)+(\frac{\overline{\tau_{1,1}}^2+\overline{\tau_{1,1}}\cdot\overline{\tau_{2,1}}+\overline{\tau_{2,1}}^2}{\overline{\tau_{1,1}}+\overline{\tau_{2,1}}}+\overline{\tau_{3,2}})\cdot n_{lim}$, где n_{lim} – предельное количество пакетов с запросами на соединение (определяется размером буфера операционной системы). $P_u^{(0)}(t)=1-\exp\left[-t/\overline{\tau_u^{(0)}}\right]$. Примечания: 1. Здесь и далее первый индекс – это номер события, второй – номер перехода (обозначения (a) и (z) опущены). 2. $s_\Omega\left[R(1,1z),R(2,1z),n_{lim}\right]$ – предикатная операция цикла с предикатами второго порядка $R(1,1z)$ и $R(2,1z)$, указывающая, что перемещение процесса из позиций 1 и 2 в переход 1z происходит n_{lim} раз. 3. $s_\Omega\left[R(0z,4),n_{lim}+1\right]$ – предикатная операция с предикатом второго порядка $R(0z,4)$, указывающая, что перемещение процесса из перехода 0z в позицию 4 происходит $n_{lim}+1$ раз
	Установлен межсетевой экран  0 – начальное состояние процесса, нарушитель начал отправку запроса на соединение с атакуемым хостом; 1 – на межсетевой экран поступил TCP-запрос на соединение с хостом нарушителя; 2 – межсетевой экран пропустил запрос на атакуемый хост с вероятностью $p_{M\Phi}$;	$\overline{\tau_u^{(M)}}=\overline{\tau_{0,0}}\cdot(n_{lim}+1)+\left[\frac{(\frac{\overline{\tau_{1,1}}}{p_{M\Phi}})^2+(\frac{\overline{\tau_{1,1}}}{p_{M\Phi}})\overline{\tau_{2,1}}+\overline{\tau_{2,1}}^2}{\frac{\overline{\tau_{1,1}}}{p_{M\Phi}}+\overline{\tau_{2,1}}}+\frac{\overline{\tau_{3,2}}^2+\overline{\tau_{3,2}}\cdot\overline{\tau_{4,2}}+\overline{\tau_{4,2}}^2}{\overline{\tau_{3,2}}+\overline{\tau_{4,2}}}+\overline{\tau_{5,3}}\right]\cdot n_{lim}$; $P_u^{(M)}(t)=1-\exp\left[-t/\overline{\tau_u^{(M)}}\right]$, где $p_{M\Phi}$ – вероятность преодоления межсетевого экрана запросом на соединение (например, вероятность того, что хост, запросивший соединение от имени доверенного хоста, принят как доверенный), $p_{M\Phi}>0$.

Наименование и идентификатор угрозы	Графическое представление модели реализации угрозы с использованием сети Петри-Маркова	Соотношение для расчета среднего времени и вероятности реализации угрозы
	3 – на атакуемый хост поступил TCP-запрос на соединение с хостом нарушителя; 4 – количество запросов на соединение не превышает допустимое значение n_{lim}; 5 – ответ-квитанция подготовлена к отправке на хост нарушителя; 6 – количество пакетов с запросом на соединение превысило величину n_{lim}, операционная система «зависает», угроз реализована; 0(z) – передача запроса с хоста нарушителя на межсетевой экран; 1(z) – логический переход, срабатывающий; 2(z) – логический переход, срабатывающий если на атакуемый хост поступил запрос на соединение и количество запросов не превышает величину n_{lim}; 3(z) – ответ-квитанция отправляется на хост нарушителя	
Угроза копирования нужного нарушителю файла путем запуска скрипта, внедряемого с использованием файлов cookies, $u=2$	Меры защиты не применяются  0 – начальное состояние процесса, нарушитель готов к проведению атаки внедрения скрипта. Атакуемый хост подготовил запрос к поисковой системе для поиска нужного ему информационного ресурса; 1 – поисковая система получила запрос и подготовила к передаче на атакуемый хост файла cookie; 2 – файл cookie перехвачен «сниффером» нарушителя на одном из узлов сети Internet и передан на хост нарушителя; 3 – в файл cookie на хосте нарушителя внедрен вредоносный скрипт и файл передан на атакуемый хост; 4 – файл cookie принят web-браузером на атакуемом хосте, открыт и запущен вредоносный скрипт для копирования файла; 5 – на атакуемом хосте запущена штатная программа поиска файла по имени и с вероятности p_{search} найден нужный нарушителю файл и подготовлен к передаче по сети; 6 – скопирован и передау на хост нарушителя нужный ему файл; 0z – передача запроса с атакуемого хоста на поисковую систему; 1z – перехват файла cookie на одном из узлов сети Internet и передача его на хост нарушителя; 2z – внедрение вредоносного скрипта в файл cookie и передача его на атакуемый хост; 3z – web-браузером на атакуемом хосте открыт и запущен вредоносный скрипт, запущена штатная программа поиска нужного нарушителю файла; 4z – копирование нужного нарушителю файла и передача его на хост нарушителя	$\overline{\tau_u^{(0)}} = \overline{\tau_{0,3}} + \frac{\overline{\tau_{4,4}}^2 + \overline{\tau_{4,4}} \cdot \overline{\tau_{5,4}} \cdot p_{search} + \overline{\tau_{5,4}}^2 \cdot p_{search}^2}{\overline{\tau_{4,4}} \cdot p_{search} + \overline{\tau_{5,4}} \cdot p_{search}^2}$, $p_{search} > 0$; $\overline{\tau_{0,3}} = \overline{\tau_{0,0}} + \overline{\tau_{1,1}} + \overline{\tau_{2,2}} + \overline{\tau_{3,3}}$; $P_u^{(0)}(t) = 1 - \exp\left[-t / \overline{\tau_u^{(0)}}\right]$

Наименование и идентификатор угрозы	Графическое представление модели реализации угрозы с использованием сети Петри-Маркова	Соотношение для расчета среднего времени и вероятности реализации угрозы
	Установлена программа «cookie-менеджер» для обнаружения вредоносного скрипта  0 – начальное состояние процесса, нарушитель готов к проведению атаки внедрения скрипта, атакуемый хост направил запрос на поиск нужного ему web-ресурса; 1 – поисковик получил запрос и подготовил к передаче на хост пользователя файла cookie; 2 – файл cookie перехвачен «сниффером» и передан на хост нарушителя; 3 – в файл cookie внедрен вредоносный скрипт для передачи на атакуемый хост; 4 – файл cookie принят и открыт web-браузером на атакуемом хосте; 5 – на атакуемом хосте найден с вероятностью p_{search} нужный нарушителю файл пользователя, скопирован и подготовлен к передаче по сети; 6 – запущена программа «cookie-менеджер» для обнаружения с вероятностью p_{det} вредоносного скрипта; 7 – подана команда на запуск вредоносного скрипта; 8 – нужный нарушителю файл скопирован и передан на хост с определенным нарушителем сетевым адресом; 0(z) – передача запроса на искомый ресурс сети Internet с атакуемого хоста на Web-сервер; 1(z) – передача Web-сервером файла cookie на хост пользователя; 2(z) – перехват файла cookie «сниффером», передача его на хост нарушителя и внедрение в него нужного нарушителю скрипта; 3(z) – передача файла cookie с внедренным скриптом на атакуемый хост; 4(z) – запуск вредоносного скрипта; 5(z) – логический переход, срабатывающий, если запущен вредоносный скрипт, завершено копирование файла пользователя и не обнаружен к этому моменту времени вредоносный скрипт, результатом срабатывания перехода является передача файла пользователя на хост нарушителя	$\overline{\tau_u^{(0)}} = \overline{\tau_{0,4}} + \overline{\tau_{5,6,7}} \cdot \left(1 + \frac{\overline{\tau_{5,6,7}}}{\overline{\tau_6} \cdot (1 - p_{det})}\right),$ $\overline{\tau_{5,6,7}} = \frac{\overline{\tau_{5,5}}^2 + \overline{\tau_{5,5}} \cdot \overline{\tau_{7,5}} \cdot p_{search} + \overline{\tau_{7,5}}^2 \cdot p_{search}^2}{\overline{\tau_{5,5}} \cdot p_{search} + \overline{\tau_{7,5}} \cdot p_{search}^2}$ $p_{search} > 0;$ $P_u^{(ШГ)}(t) = 1 - \exp\left[-t / \overline{\tau_u^{(ШГ)}}\right].$ Примечание: 1.Если $p_{det} \to 1$, то $\overline{\tau_{ш}} \to \infty$ и $P_u^{(ШГ)}(t) \to 0$. 2.Если $p_{search} \to 0$, то $\overline{\tau_{ШГ}} \to \infty$ и $P_u^{(ШГ)}(t) \to 0$

Для этого графа составляются системы интегро-дифференциальных уравнений для парциальных полумарковских процессов:

1) между нулевым состоянием и первым переходом «ИЛИ» с охватом состояний 0, 2, 3 и 5;

2) между нулевым состоянием и первым переходом «И» с охватом состояний 0, 1, 3 и 4;

3) между двумя переходами «ИЛИ» с охватом состояний 6, 7, 10 и 11;

4) между нулевым состоянием и вторым переходом «ИЛИ» с охватом состояний 0, 3, 8 и между переходами «И» и «ИЛИ» с охватом состояния 9.

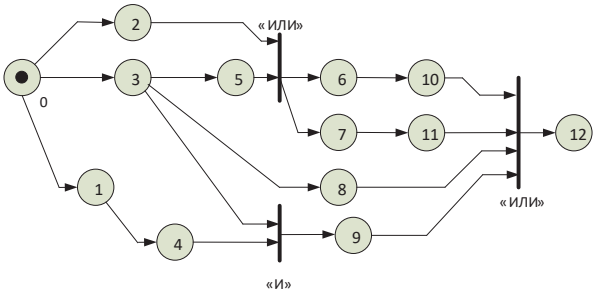


Рисунок 6.9 – Пример графа составного полумарковского процесса с логическими условиями «И» и «ИЛИ»

При этом полагается, что процесс выходит из перехода мгновенно, а из состояния за ограниченное случайное время.

Так как в рассматриваемых ССПМ имеются переходы с логическими условиями, то ССПМ так же, как и ССПМ на основе марковских процессов, разбивается на участки между логическими переходами, каждый из которых может иметь более одной траектории перемещения (см. рисунок 6.4).

При разбиении СПМ рассматриваются участки, соответствующие перемещениям:

а) из начального состояния до первого логического перехода;

б) между логическими переходами;

в) после последнего логического перехода до конечного состояния.

Пусть на участке h, завершающемся логическим переходом с некоторой логикой, имеется несколько траекторий и на этот переход с других участков не подходит ни одна дуга. Для такого участка плотность вероятности времени прохождения процесса из начальной позиции с номером 1 в логический переход с номером J представляет собой свертку функций распределения времен перемещения из каждой позиции в инцидентный ей переход (см. формулу 6.48), то есть:

$$f_{1,J}^{(h)}(t) = f_{1,1}^{(h)}(t) * f_{2,2}^{(h)}(t) * \ldots * f_{J,J}^{(h)}(t), \qquad (6.28)$$

где $f_{j,j}^{(h)}(t)$ – плотность вероятности для времени перемещения процесса из j-й позиции на траектории h в следующий за ней (инцидентный ей) j-й переход;

*– операция свертки [92].

При этом предельная вероятность достижения перехода, то есть вероятность перемещения по вложенной цепи Маркова, если вероятность перемещения из i-й позиции в следующий за ней i-й переход равна π_{ii}, определяются произведением[29]

$$\pi_{1,J} = \prod_{i=1}^{J} \pi_{ii} . \qquad (6.29)$$

Так как случайные времена перемещения из позиций в переходы являются статистически независимыми, то математическое ожидание суммарного времени перемещения процесса из начальной позиции траектории в конечный переход определяется простым суммированием математических ожиданий времен ($\overline{\tau_{ii}^{(h)}}$) перемещения процесса из каждой позиции в каждый переход по данной траектории

$$\overline{\tau_{1,J}^{(h)}}(t) = \sum_{i=1}^{J} \overline{\tau_{i,i}^{(h)}} . \qquad (6.30)$$

Если к логическому переходу на данном участке подходят дуги с других участков, то для них тоже вычисляется время и предельная вероятность перемещения процесса по этим дугам. Аналогичным образом рассчитываются средние времена перемещения по другим участкам.

Далее СПМ преобразуется следующим образом: каждый ее участок между логическими переходами заменяется позицией с дугой, входящей в логический переход. Затем для полученного графа рассчитываются времена срабатывания логических переходов.

Рассмотрим некоторые особенности такого расчета для различных логических условий, а также построения ССПМ в случае наличия альтернативных траекторий перемещения моделируемого процесса.

Математические ожидания времен срабатывания переходов для ССПМ на основе ПМП рассчитываются так же, как и в случае марковских парциальных процессов.

Однако при расчете вероятностей срабатывания необходимо учитывать срабатывание вложенной марковской цепи. Вложенная марковская цепь характеризуется вероятностями π_{ij} того, что моделируемому процессу «разрешено» перемещаться из i-й позиции в j-й переход, при этом перемещения по марковской цепи не зависят от времени и не влияют на временные характеристики перемещения моделируемого процесса из позиции в переход. Это обстоятельство позволяет рассматривать процедуры оценки вероятностностей срабатывания логических переходов следующим образом.

[29] Здесь для упрощения применена последовательная нумерация позиций и переходов в пределах участка, в общем случае может быть единая нумерация в рамках всей СПМ.

Логический переход «И» для двух входящих дуг срабатывает в том случае, когда процесс как по первой, так и по второй дуге подошел к переходу. Вероятности перемещения по вложенной марковской цепи по каждой из двух дуг (рисунок 6.10) рассчитываются как произведения:

$$\pi_1 = \prod_{j=1^{(1)}}^{J^{(1)}} \pi_{jj}^{(1)} \text{ и } \pi_2 = \prod_{j=1^{(2)}}^{J^{(2)}} \pi_{jj}^{(2)}, \tag{6.31}$$

где верхние индексы соответствуют номеру траектории на участке СПМ.

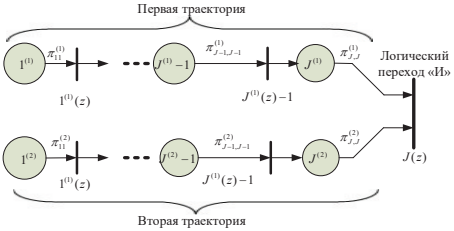


Рисунок 6.10 – Траектории перемещения моделируемого процесса к логическому переходу «И»

Вероятность того, что переход при двух входящих потоках сработает за время t с учетом срабатывания вложенной марковской цепи, определяется из соотношения:

$$P_И(t) = \pi_1 \cdot \pi_2 \cdot P_1(t) \cdot P_2(t). \tag{6.32}$$

При экспоненциальном приближении вероятность срабатывания перехода за время t определяется следующим образом:

$$P_И(t) = \pi_1 \cdot \pi_2 \cdot \left\{1 - \exp\left[-t \cdot \left(\overline{\tau_1} + \overline{\tau_2}\right) \Big/ \left(\overline{\tau_1}^2 + \overline{\tau_1} \cdot \overline{\tau_2} + \overline{\tau_2}^2\right)\right]\right\}. \tag{6.33}$$

Логический переход «ИЛИ» для двух входящих дуг срабатывает в том случае, когда процесс или по первой, или по второй дуге подошел к переходу. Вероятность того, что переход сработает за время t без учета вложенной марковской цепи, определяется по формуле (4.65), а с ее учетом – из соотношения

$$P_{ИЛИ}(t) = \left[1 - \left(1 - \pi_1\right) \cdot \left(1 - \pi_2\right)\right] \cdot \left\{1 - \left[1 - P_1(t)) \cdot (1 - P_2(t)\right]\right\}. \tag{6.34}$$

С учетом экспоненциального приближения вероятность срабатывания перехода за время t определяется следующим образом:

$$P_{ИЛИ}(t) = \left[1 - \left(1 - \pi_1\right) \cdot \left(1 - \pi_2\right)\right] \cdot \left\{1 - \exp\left[-t \cdot \left(\overline{\tau_1} + \overline{\tau_2}\right) \Big/ \left(\overline{\tau_1} \cdot \overline{\tau_2}\right)\right]\right\}. \tag{6.35}$$

Логический переход «И-НЕ» применительно к двум входящим потокам срабатывает в том случае, если по первой из дуг процесс подошел к переходу, а по второй – нет, при этом моделируемому процессу «разрешено» с вероятностями π_1 и π_2 войти в логический переход. Вероятность того, что

переход сработает за время t без учета вложенной марковской цепи, определяется по формуле (4.80), а с ее учетом – из соотношения

$$P_{И-НЕ}(t) = \pi_1 \cdot \pi_2 \cdot P_1(t)\left[1 - P_2(t)\right] + \pi_1 \cdot \left(1 - \pi_2\right) \cdot P_1(t)\,. \qquad (6.36)$$

С учетом экспоненциального приближения вероятность срабатывания перехода за время t определяется следующим образом:

$$P_{И-НЕ}(t) = \pi_1 \cdot \pi_2 \cdot \left\{1 - \exp\left[-t \Big/ \overline{\tau}_1 \cdot \left(1 + \frac{\overline{\tau}_1}{\overline{\tau}_2}\right)\right]\right\} + \pi_1 \cdot \left(1 - \pi_2\right) \cdot \left\{1 - \exp\left[-t/\overline{\tau}_1 \cdot\right]\right\}. \qquad (6.37)$$

Логический переход «И-ИЛИ» имеет место применительно к не менее, чем трем входящим потокам. Он срабатывает, если процесс подошел к переходу или по первой и второй дуге, или по третьей. Вероятность того, что переход сработает за время t с учетом вложенной марковской цепи, определяется из соотношения

$$P_{И-ИЛИ}(t) = \pi_1 \cdot \pi_2 \cdot \pi_3 \cdot \left\{1 - \left[1 - P_1(t) \cdot P_2(t)\right] \cdot (1 - P_3(t))\right\}. \qquad (6.38)$$

При экспоненциальном приближении вероятность срабатывания перехода за время t определяется следующим образом:

$$P_{И-ИЛИ}\left(t\right) = \pi_1 \cdot \pi_2 \cdot \pi_3 \left\{ - \frac{t}{\dfrac{\overline{\tau}_1 \overline{\tau}_2}{\overline{\tau}_1 + \overline{\tau}_2} + \dfrac{\overline{\tau}_1^2 \overline{\tau}_3}{\left(\overline{\tau}_1 + \overline{\tau}_3\right)^2} + \dfrac{\overline{\tau}_2^2 \overline{\tau}_3}{\left(\overline{\tau}_2 + \overline{\tau}_3\right)^2} - \dfrac{\overline{\tau}_1 \overline{\tau}_2 \overline{\tau}_3}{\overline{\tau}_1 \overline{\tau}_2 + \overline{\tau}_1 \overline{\tau}_3 + \overline{\tau}_2 \overline{\tau}_3}} \right\}. \qquad (6.39)$$

Наконец, при построении СПМ на ССПМ возникает необходимость учета альтернативных вариантов развития моделируемого процесса (рисунок 6.11).

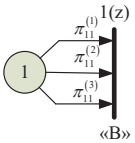


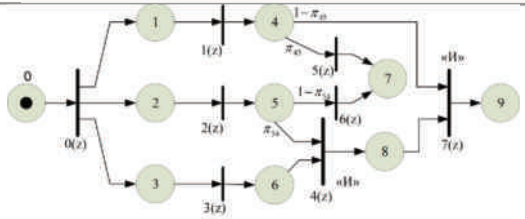
Рисунок 6.11 – Фрагмент сети Петри-Маркова с альтернативными вариантами развития моделируемого процесса

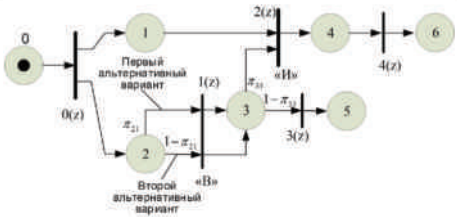
В этом случае или для каждого варианта строится своя СПМ, или на графе СПМ с альтернативными вариантами вводится дополнительный логический переход «Выбор» («В»), при поступлении на который процесса с вероятностями $\pi_{ij}^{(h)}, h = \overline{1,H}, \sum_h \pi_{ij}^{(h)} = 1$ осуществляется выбор одной из альтернатив и далее для нее проводится расчет, затем выбираются другие альтернативы и по ним также проводятся расчеты. В результате расчета

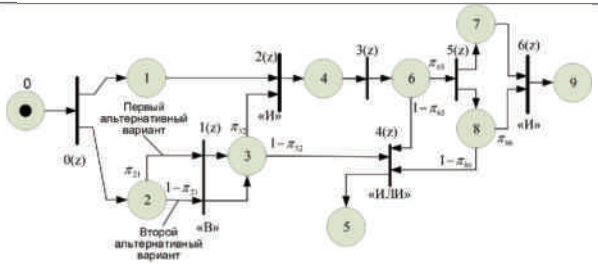
получаются альтернативные результаты, которые затем могут быть свернуты с вероятностями $\pi_{ij}^{(h)}$ в один результирующий показатель оценки возможности реализации угрозы. Примеры применения составных СПМ на основе ПМП при моделировании процессов реализации угроз безопасности информации в ИС приведены в таблице 6.11.

Необходимо отметить в этом случае важность корректного задания вероятностей перемещения по вложенной цепи Маркова. Как правило, для этого приходится разрабатывать дополнительные модели, связанные, например, с динамикой преодоления мер защиты, внедрения вредоносной программы, своевременным обнаружением сетевой атаки и т.д. Для разработки таких моделей, в свою очередь, могут быть использованы ССПМ на основе как марковских, так и полумарковских процессов.

Таблица 6.11 – Примеры моделей динамики реализации основных угроз безопасности информации, связанных с сетевыми атаками, с использованием составных сетей Петри-Маркова на основе полумарковских процессов

Наименование сетевой атаки	Сеть Петри-Маркова, моделирующая динамику реализации угрозы	Обозначения, примечания
Подмена доверенного объекта (IP- spoofing)	 0 – нарушитель включился в сеть общего пользования; 1 – нарушитель в готовности к проведению «шторма TCP-запросов» на хост доверенного пользователя; 2 – нарушитель в готовности к подбору (прогнозу) номера ответного пакета и порта взаимодействия хоста доверенного пользователя; 3 – нарушитель в готовности к организации связи с атакуемым хостом; 4 – хост доверенного пользователя атакован «штормом TCP-запросов»; 5 – нарушитель завершил выявление порта взаимодействия и номера пакета подтверждения соединения и с вероятностью $\pi_{5,4}$ создал условия для соединения с атакуемым хостом; 6 – атакуемый хост получил TCP-запрос; 7 – атака сорвалась из-за неправильного подбора порта взаимодействия или номера пакета подтверждения соединения, или из-за срыва подавления доверенного хоста; 8 – установлено TCP соединение с атакуемым хостом от имени доверенного объекта; 9 – осуществлен НСД к атакуемому хосту от имени доверенного пользователя; 0(z) – подготовка к проведению атаки; 1(z) – передача запроса на соединение с атакуемым хостом; 2(z) – подбор (прогноз) порта взаимодействия и номера пакета подтверждения соединения; 3(z) – осуществление «шторма» TCP-запросов для подавления доверенного сервера (блокирование хоста доверенного пользователя); 4(z) – создание условий для установления соединения с атакуемым хостом; 5(z) – срыв атаки из-за неправильного подбора порта или номера пакета доверенного хоста; 6(z) – прекращение (срыв) атаки из-за срыва блокировки хоста доверенного пользователя, отправка от имени доверенного объекта на атакуемый хост ответа S-SYN и S-ACK на установку соединения; 7(z) – получение прав доверенного пользователя и несанкционированный доступ к защищаемой информации	$\bar{\tau}_{\text{и}} = \bar{\tau}_{0,0} + \overline{\tau_{7(z)}}$; $\overline{\tau_{7(z)}} = \dfrac{\bar{\tau}^2_{1,7} + \bar{\tau}_{1,7} \cdot \bar{\tau}_{(2,3),7} + \bar{\tau}^2_{(2,3),7}}{\bar{\tau}_{1,7} + \bar{\tau}_{(2,3),7}}$; $\overline{\tau_{17}} = \overline{\tau_{11}} + \overline{\tau_{47}}$; $\bar{\tau}_{(2,3)7} = \dfrac{\overline{\tau^2_{24}} + \overline{\tau_{24}} \cdot \overline{\tau_{14}} + \overline{\tau^2_{34}}}{\overline{\tau_{24}} + \overline{\tau_{34}}} + \overline{\tau_{87}}$; $\overline{\tau_{24}} = \overline{\tau_{22}} + \overline{\tau_{54}}$; $\overline{\tau_{34}} = \overline{\tau_{33}} + \overline{\tau_{64}}$; $P_{\text{и}}(t) = \pi_{5,4} \cdot (1 - \pi_{4,5}) \cdot (1 - e^{-\frac{t}{\tau_{\text{и}}}})$ <u>Примечание:</u> а) первый номер в индексе соответствует номеру позиции, второй – номеру перехода

Наименование сетевой атаки	Сеть Петри-Маркова, моделирующая динамику реализации угрозы	Обозначения, примечания
Парольная атака на хост сети	 0 – нарушитель включился в сеть общего пользования; 1 – абонентом проведен запрос пароля; 2 – нарушитель, не зная пароля, решил подобрать его по специальному словарю (первый альтернативный вариант) с вероятностью π_{21} или путем прямого перебора (второй альтернативный вариант) с вероятностью $1-\pi_{21}$; 3 – нарушитель завершил подбор пароля; 4 – пароль подобран правильно с вероятностью π_{33}; 5 – пароль подобран неправильно с вероятностью $1-\pi_{33}$; 6 – атака завершена; 0(z) – абонент – объект атаки запрашивает пароль; 1(z) – подбор пароля по одному из альтернативных вариантов (логический переход «Выбор»); 2(z) – предоставление доступа нарушителю по паролю с вероятностью π_{33} при попытке получения доступа со стороны нарушителя (логический переход «И») ; 3(z) – пресечение несанкционированного доступа из-за неправильно подобранного пароля; 4(z) – завершение атаки	$\overline{\tau_{и}} = \overline{\tau_{04}} = \overline{\tau_{и^*}} + \overline{\tau_{44}}$; $\overline{\tau_{и^*}} = \overline{\tau_{00}} + \frac{\overline{\tau_{12}}^2 + \overline{\tau_{12}} \cdot \overline{\tau_{22}} + \overline{\tau_{22}}^2}{\overline{\tau_{12}} + \overline{\tau_{12}}}$; $\overline{\tau_{22}} = \overline{\tau_{21}} + \overline{\tau_{32}}$; $P_{и}(t) = \begin{cases} \pi_{21} \cdot \pi_{33} \cdot (1 - e^{-\frac{t}{\tau_{и}}}) - \text{в первом варианте;} \\ (1-\pi_{21}) \cdot \pi_{33} \cdot (1 - e^{-\frac{t}{\tau_{и}}}) - \text{во втором варианте} \end{cases}$

Наименование сетевой атаки	Сеть Петри-Маркова, моделирующая динамику реализации угрозы	Обозначения, примечания
Несанкционированный запуск приложения	 0 – нарушитель сделал запрос на соединение с атакуемым хостом; 1 – абонентом проведен запрос пароля; 2 – нарушитель, не зная пароля, решил подобрать его путем прямого перебора (первый вариант) или по специальному словарю (второй вариант); 3 – нарушитель завершил подбор пароля; 4 – пароль подобран правильно; 5 – атака сорвана; 6 – поиск приложения завершен; 7 – предпринята попытка запуска приложения; 8 – проведен анализ трафика на предмет обнаружения атаки; 9 – атака реализована; 0(z) – абонент – объект атаки запрашивает пароль; 1(z) – подбор пароля по словарю или прямым перебором (логический переход «Выбор»); 2(z) – при наличии запроса на соединение на атакуемом хосте вводится пароль доступа для нарушителя (логический переход «И»); 3(z) – получение доступа в операционную среду объекта атаки и поиск приложения; 4(z) – формирование сообщения об отказе в доступе или из-за неправильно подобранного пароля, или из-за обнаружения атаки, или из-за срыва поиска нужного приложения (логический переход «ИЛИ»); 5(z) – передача команды на запуск приложения, включение системы обнаружения вторжений на атакуемом хосте; 6(z) – запуск приложения при условии необнаружения атаки	$\overline{\tau_0}=\overline{\tau_{10}}=\overline{\tau_{10}}+\overline{\tau_{2(z)}}+\overline{\tau_{6(z)}};\quad \overline{\tau_{2(z)}}=\frac{\overline{\tau_{12}^2}+\overline{\tau_{12}}\cdot(\overline{\tau_2}+\overline{\tau_{22}})+(\overline{\tau_2}+\overline{\tau_{22}})^2}{\overline{\tau_{12}}+\overline{\tau_2}+\overline{\tau_{22}}};$ $\overline{\tau_{6(z)}}=\overline{\tau_{43}}+\overline{\tau_{65}}+\frac{\overline{\tau_{76}^2}+\overline{\tau_{76}}\cdot\overline{\tau_{86}}+\overline{\tau_{76}^2}}{\overline{\tau_{76}}+\overline{\tau_{86}}};$ $P_n(t)=\begin{cases}\pi_{21}\cdot\pi_{32}\cdot\pi_{65}\cdot\pi_{85}\cdot(1-e^{-\frac{t}{\tau_n}})-\text{в первом варианте;}\\(1-\pi_{21})\cdot\pi_{32}\cdot\pi_{65}\cdot\pi_{85}\cdot(1-e^{-\frac{t}{\tau_n}})-\text{во втором варианте}\end{cases}$

7. МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ ФОРМИРОВАНИЯ ТРЕБОВАНИЙ К ЭФФЕКТИВНОСТИ ЗАЩИТЫ ИНФОРМАЦИИ В ИНФОРМАЦИОННЫХ СИСТЕМАХ

7.1. Особенности формирования требований к эффективности защиты информации в информационных системах

Сегодня в стране действуют более 80 нормативных правовых документов и более 40 государственных стандартов в области ТЗИ в ИС. Структура системы таких требований весьма обширна, поскольку они охватывают самые различные аспекты регулирования деятельности по защите информации, такие как формирование терминологии в предметной области, категорирование ИС и информационных ресурсов, построение систем защиты в функционирующих ИС и вновь создаваемых ИС в защищенном исполнении, создание программных и программно-аппаратных средств защиты и контроля защищенности информации в ИС и др. Охватить все эти аспекты в одной работе не представляется возможным, поэтому в данной монографии рассматривается только то, что касается требований к эффективности защиты информации в ИС.

Сегодня формирование требований по ТЗИ в ИС фактически не связывается с количественной оценкой эффективности защиты. При этом защита считается эффективной, если выполняются требования, установленные действующим нормативным правовым документом государственного регулятора или стандартами (международными или национальными), а неэффективной, – если эти требования не выполняются. Состав требований сформирован экспертно без каких-либо количественных обоснований достигаемой защищенности информации в ИС.

Несомненным достоинством такого подхода является простота и ясность процедуры формирования требований. Однако ему присущи весьма существенные недостатки, к основным из которых относятся:

наличие лишь опосредованной связи требований с оценками возможностей реализации угроз (по сути, угроза рассматривается как актуальная, если экспертно определено, что она реализуема в данной ИС и наносит значимый ущерб для обладателя информации);

отсутствие учета фактора времени при задании требований;

отсутствие какого-либо обоснования при соотнесении классов защищенности ИС и классов защиты применяемых или подлежащих применению в ИС средств защиты информации;

неопределенности, возникающие при попытке применения новых мер и средств защиты в интересах парирования новых угроз безопасности информации, не известных при создании данного нормативного документа;

неопределенности, возникающие при настройках программных или программно-аппаратных средств и систем защиты[30] (с указанными

[30] К таким системам часто относят, например, распределенные по ИС системы антивирусной защиты, системы обнаружения вторжений, DLP-системы, SIEM-системы, SOC-системы и т.д. []

настройками, как правило, не связываются установленные требования по ТЗИ в ИС) и т.д.

В этих условиях несомненно важной является разработка количественных методов обоснования требований по ТЗИ в ИС и, в частности, требований к эффективности ТЗИ, которые учитывали бы возможности реализации угроз в конкретных ИС и уровень наносимого при этом ущерба. Такие требования сегодня можно и целесообразно разрабатывать с использованием теории рисков при анализе возможностей реализации угроз безопасности информации без применения и с применением мер и средств защиты. В этом случае требования должны формироваться путем установления предельных значений или требуемого значения показателя эффективности защиты, или предельно допустимого риска реализации рассматриваемой угрозы (совокупности угроз). Вместе с тем сегодня фактически отсутствует подход к определению как требуемого значения показателя эффективности, так и предельно допустимого риска реализации угроз.

Наиболее простой путь для определения предельных значений указанных параметров состоит в установлении их обладателем информации, возможно с градацией, например, по требуемым классам защищенности ИС. В этом есть резон, так как каждой ИС соответствует своя шкала ценностей обрабатываемой в ней информации для ее обладателя, существенно зависящая не только от того, какая информации, подлежащая защите, хранится и обрабатывается в ИС, какие имеют место объемы и сроки ее обработки и т.д., но и от потребностей и интересов самого обладателя. Это, с одной стороны, приводит к значительному упрощению процедуры обоснования, но, с другой стороны, обусловливает необходимость разработки состоятельных, обоснованных и проверенных на практике применительно к разным классам ИС рекомендаций по реализации указанной процедуры.

Значительно более перспективным является подход к формированию *количественных требований* к эффективности защиты информации и к предельно допустимому риску реализации угроз. Именно этот подход рассматривается далее.

Соответствующая структура системы количественных требований к эффективности защиты информации и к предельно допустимому риску реализации угроз приведена на рисунке 7.1.

Процедуры обоснования приведенной системы требований имеют определенные особенности, суть которых состоит в следующем.

1. Процедура обоснования требований является иерархической, осуществляемой «сверху-вниз», например, а) начиная с требований к эффективности ЗИ в течение всего жизненного цикла ИС и заканчивая этапом вывода ее из эксплуатации, б) начиная с требований к эффективности защиты информации в ИС в целом и заканчивая требованиями к эффективности защиты информации, содержащейся в системных файлах и файлах с пользовательской информацией и т.д.

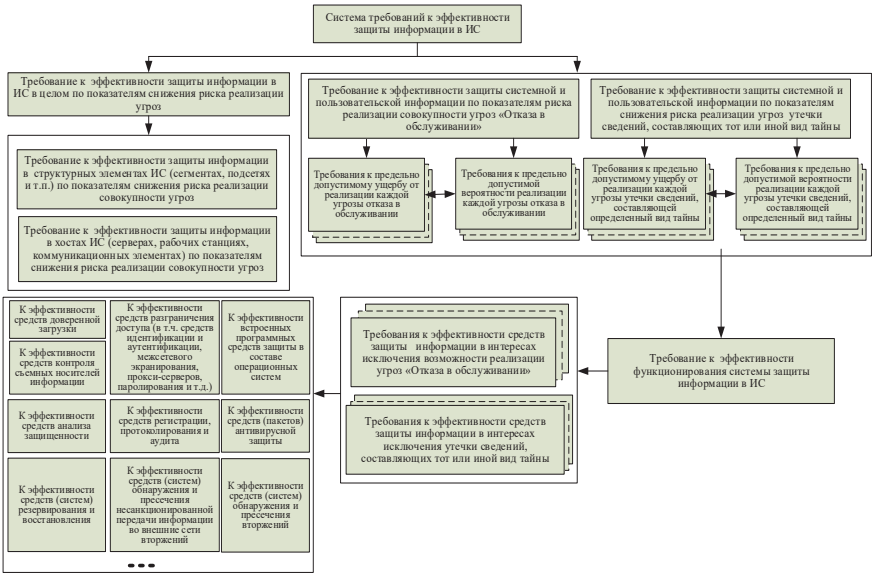


Рисунок 7.1 – Система требований к эффективности защиты информации в информационной системе

Необходимо отметить, что в данной монографии не рассматривается вопрос распределения требований к эффективности защиты информации по этапам жизненного цикла ИС, а только – формирование таких требований в пределах этапа эксплуатации ИС.

2. Должна быть обеспечена преемственность подхода к заданию требований по эффективности в части их дифференциации по классам защищенности ИС и средств вычислительной техники (СВТ) в их составе.

3. Формирование требований к эффективности защиты ИС и ее элементов от угроз «Отказа в обслуживании» (угроз вывода из строя, нарушения функционирования) связаны с определением предельно допустимых показателей возможности нарушения функционирования операционных систем хостов (серверов, рабочих станций), функционирования коммуникационных элементов ИС (маршрутизаторов, коммутаторов, концентраторов, повторителей и т.д.), что возможно сегодня путем, во-первых, воздействия на системные файлы, во-вторых, использования уязвимостей операционных систем и организации обмена сообщениями между хостами, в-третьих, использования специально сформированных вредоносных программ с соответствующими деструктивными функциями и т.д. Кроме того, рассматриваются вопросы формирования требований к эффективности защиты пользовательских прикладных программ и данных, без которых применение ИС оказывается невозможным. Сегодня фактически отсутствует подход к заданию требований к эффективности защиты

информации, нарушение безопасности которой приводит к нарушению функционирования ИС. В данной монографии излагается один из возможных подходов к формированию таких требований на основе использования теории риска.

4. Формирование требований к эффективности защиты от угроз утечки конфиденциальной информации (информации ограниченного доступа) в ИС и ее элементах обусловлено содержанием сведений, составляющих тот или иной вид тайны (государственной, служебной, коммерческой, профессиональной, личной) и персональные данные. При этом должен быть обеспечен дифференцированный подход к требованиям по эффективности защиты в зависимости от того, какие сведения защищаются от утечки. Кроме того, ущерб от утечки сведений, как было отмечено в разделе 1.3, существенно зависит от того, к кому и в какие сроки попадут эти сведения, а также, насколько быстро они устаревают, что, как правило, неизвестно. Все это приводит к тому, что при формировании требований к эффективности ТЗИ чаще всего полагается, что любая утечка конфиденциальной информации приводит к недопустимому ущербу, а требования к эффективности ее защиты формируются только применительно к предельно допустимой возможности ее утечки (перехвата, копирования, несанкционированной передачи и т.д.).

7.2. Формирование требований к эффективности защиты информационной системы и ее элементов от угроз отказа в обслуживании на основе их сопоставления с требованиями по надежности функционирования информационных систем

Формирование требований к эффективности защиты информационной системы от угроз отказа в обслуживании сводится к определению требуемого значения показателя эффективности защиты (см. раздел 2):

при использовании безотносительного показателя эффективности

$$\eta_{dif}^{(треб)}(T) = R_u^{(0)}(T) - R_u^{(доп)}; \qquad (7.1)$$

при использовании относительного показателя эффективности

$$\eta_{rel}^{(треб)}(T) = R_u^{(доп)}(T) \big/ R_u^{(0)}(T),\ R_u^{(0)}(T) > 0; \qquad (7.2)$$

при использовании относительного разностного показателя эффективности

$$\eta_{reldif}^{(треб)}(T) = \left| R_u^{(0)}(T) - R_u^{(доп)}(T) \right| \big/ R_u^{(0)}(T),\ R_u^{(0)}(T) > 0, \qquad (7.3)$$

где $R_u^{(0)}(T)$ – риск, возникающий для ИС при реализации угрозы отказа в обслуживании при отсутствии мер защиты за период оценки T;

$R_u^{(доп)}(T)$ – допустимое значение риска, определяемое при организации защиты от угрозы отказа в обслуживании за период оценки T;

Если при отсутствии мер защиты риск нарушения функционирования близок к единице, то формулы значительно упрощаются:

$$\eta_{dif}^{(треб)}(T)=\eta_{reldif}^{(треб)}\left(T\right)=1-R_{u}^{(доп)}(T);\ \eta_{rel}^{(треб)}(T)=R_{u}^{(доп)}(T)\,. \qquad (7.4)$$

Весьма существенным обстоятельством при задании требуемых значений показателей эффективности является учет фактора времени, то есть необходимость определения периода времени, в течении которого должна сохраняться эффективность защиты, как это принято в теории надежности [105–108]. Таким периодом может быть, например, период времени до пересмотра состава актуальных угроз и мер защиты в связи с появлением новых угроз, изменением условий функционирования ИС, состава и характеристик ее аппаратного обеспечения и др. Это может быть и плановый период времени: для одних ИС это может быть один год, для других три и даже пять лет. Вместе с тем следует иметь в виду, что отказ от учета фактора времени приводит к несостоятельности требований, предъявляемых к эффективности защиты информации в ИС.

Таким образом, при формировании требований к эффективности защиты ИС от угрозы отказа в обслуживании достаточно задать предельно допустимое значение риска реализации такой угрозы $R_{u}^{(доп)}\left(T\right)$ за период оценки T. Далее задаются требования к предельно допустимому среднему ущербу $\overline{\zeta_{u}^{(доп)}}$ и к предельно допустимой вероятности реализации угрозы отказа в обслуживании $P_{u}^{(доп)}\left(T\right)$ для ИС в целом так, чтобы:

$$\overline{\zeta_{u}^{(доп)}}\cdot P_{u}^{(доп)}\left(T\right)\le R_{u}^{(доп)}\left(T\right). \qquad (7.5)$$

При этом, если требования устанавливаются для многих ИС в рамках одного ведомства или государственным регулятором для ИС различной важности, для которых установлены определенные классы защищенности, то требования к эффективности защиты должны задаваться дифференцированно с учетом класса защищенности данной конкретной ИС, то есть

$$\overline{\zeta_{u}^{(доп)}}\left(k\right)\cdot P_{u}^{(доп)}\left(k,T\right)\le R_{u}^{(доп)}\left(k,T\right),k=\overline{1,K}\,, \qquad (7.6)$$

где k – номер класса защищенности, заданный для данной ИС. Самая высокая защищенность соответствует первому классу ($k=1$), а самая низкая – последнему ($k=K$);

K – общее количество классов защищенности, установленное государственным регулятором или в ведомстве.

Если установлен предельно допустимый ущерб от угрозы отказа в обслуживании для первого класса $\overline{\zeta_{u}^{(доп)}}\left(1\right)$, то для $k>1$ имеет место соотношение:

$$\overline{\zeta_{u}^{(доп)}}\left(k\right)=\varepsilon\left(k\right)\overline{\zeta_{u}^{(доп)}}\left(1\right),\varepsilon\left(k\right)\ge 1\,, \qquad (7.7)$$

где $\varepsilon\left(k\right)$ – коэффициент увеличения предельно допустимого ущерба от угрозы отказа в обслуживании для k-го класса защищенности по сравнению с

первым (функция $\varepsilon(k)$ устанавливается в ведомстве или государственным регулятором и может быть как линейной, так и нелинейной зависимостью от k).

В основе излагаемого подхода к формированию требования к *предельно допустимой вероятности реализации угрозы отказа в обслуживании* лежит идея о том, что нарушение функционирования ИС одинаково опасно для обладателя и пользователей ИС независимо от того, произошло ли оно из-за преднамеренного воздействия на программно-аппаратные элементы ИС или из-за отказа этих элементов в условиях отсутствия указанного воздействия, то есть из-за возникновения неисправностей, сбоев, ошибок в настройках, ошибок пользователей и т.д., характеризуемых таким понятием как «функциональная надежность» ИС [1, 105, 106], которая оценивается вероятностью безотказной работы [107] или в соответствии с [105] вероятностью безошибочного выполнения процесса, то есть функционирования ИС. Если учесть, что сегодня требования к функциональной надежности задаются соответствующими стандартами, например, стандартами [107, 108], то имеется возможность обосновать предельно допустимую вероятность реализации угрозы отказа в обслуживании.

Если положить, что для первого класса защищенности вероятность $P_{ud}^{(\partial on)}(1,t)$ того, что угроза отказа в обслуживании в результате преднамеренного воздействия будет реализована, окажется меньше, чем вероятности выхода из строя ИС из-за отказа элемента или программы, то

$$P_{ud}^{(\partial on)}(1,T) \le P_{func}^{(\partial on)}(T), \tag{7.8}$$

где $P_{func}^{(\partial on)}(T)$ – предельно допустимая вероятность того, что ИС выйдет из строя из-за неисправностей, сбоев, ошибок и т.д. за время T.

Для других классов защищенности по аналогии с формулой (7.7) требования несколько снижаются:

$$P_{ud}^{(\partial on)}(k,T) \le \varepsilon_p(k) \cdot P_{func}^{(\partial on)}(T), \varepsilon_p(k) \ge 1,\ k = \overline{1,K}, \tag{7.9}$$

где $\varepsilon_p(k)$ – коэффициент увеличения предельно допустимой вероятности реализации угрозы отказа в обслуживании для k-го класса защищенности по сравнению с первым.

При таком подходе учитывается, является ли источник u-й угрозы внутренним или внешним в соответствующей модели процесса реализации угрозы. Если источников (внутренних или внешних) много и ими используется один и тот же способ реализации угрозы, то это учитывается в интенсивности возникновения угрозы. Если количество угроз с одинаковым по содержанию несанкционированным действием составляет величину U, то формула (7.9) преобразуется к виду:

$$1 - \prod_{u=1}^{U}\left[1 - P_{ud}^{(\partial on)}(k,T)\right] \le \varepsilon_p(k) \cdot P_{func}^{(\partial on)}(T), \varepsilon_p(k) \ge 1,\ k = \overline{1,K}, u = \overline{1,U}, \tag{7.10}$$

откуда при одинаковых требованиях по защите от каждой u-й угрозы

$$P_{ud}^{(\partial on)}\left(k,T\right)\le 1-\sqrt[U]{1-\varepsilon_p\left(k\right)\cdot P_{func}^{(\partial on)}\left(T\right)},\varepsilon_p\left(k\right)\ge 1,\ k=\overline{1,K},u=\overline{1,U}\ , \qquad (7.11)$$

Более сложной оказывается процедура задания требований к эффективности защиты элементов ИС (серверов, рабочих станций, коммутационных устройств и т.п.) от угрозы нарушения их функционирования. При этом полагается, что ИС выходит из строя, когда выходит из строя:

1) любой из некоторого множества элементов ИС, относящихся к критически важным, то есть нарушение функционирования любого из этих элементов неизбежно приводит к отказу в обслуживании всей ИС;

2) любая из некоторых совокупностей элементов ИС, при этом в каждой совокупности выходят из строя все входящие в нее элементы, то есть нарушение функционирования ИС происходит, когда деструктивное воздействие происходит на все элементы хотя бы одной из критически важных совокупностей элементов ИС.

Так как последствия примерно одинаковы, независимо от того, какой критически важный элемент выводится из строя, то требования предъявляются только к предельно допустимой вероятности вывода из строя каждого критически важного элемента.

В первом из указанных случаев сначала определяются предельно допустимее значение вероятности $P_{ud}^{(\partial on)}\left(n_{kr}\ge 1,T\right)$ того, что в результате преднамеренного деструктивного воздействия выйдет из строя хотя бы один из критически важных элементов ИС, которая без учета класса защищенности определяется из уравнения:

$$P_{ud}^{(\partial on)}\left(n\ge 1,T\right)=1-\prod_{n=1}^{N_{кр}}\left[1-P_{ud}^{(\partial on)}\left(n,T\right)\right]\le P_{func}^{(\partial on)}\left(T\right), \qquad (7.12)$$

где $P_{ud}^{(\partial on)}\left(n,T\right)$ – предельно допустимая вероятность вывода из строя n-го критически важного элемента в составе ИС за время T, откуда при равных требованиях применительно ко всем критически важным элементам имеет место (рисунок 7.2) для любого n-го критически важного элемента соотношение

$$P_{ud}^{(\partial on)}\left(n,T\right)\le 1-\sqrt[N_{кр}]{1-P_{func}^{(\partial on)}\left(T\right)}\ . \qquad (7.13)$$

С учетом k-го класса защищенности ИС в соответствии с соотношением (7.9) формула (7.13) преобразуется к виду

$$P_{ud}^{(\partial on)}\left(k,n,T\right)\le 1-\sqrt[N_{кр}]{\left[1-\varepsilon_p\left(k\right)\cdot P_{func}^{(\partial on)}\left(T\right)\right]},\ \varepsilon_p\left(k\right)\ge 1,\ k=\overline{1,K}\ . \qquad (7.14)$$

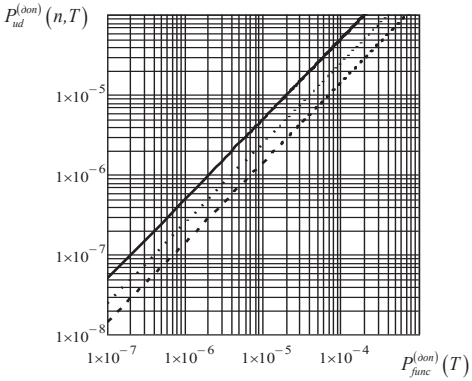


Рисунок 7.2 – Зависимость допустимой вероятности нарушения функционирования элементов ИС от допустимой вероятности нарушения функционирования ИС в целом

Если предельно допустимая вероятность реализации угрозы за оцениваемый период времени установлена и в соответствии с моделью угроз известны возможные интенсивности (λ_{ud}) попыток реализации угроз, то можно установить допустимые вероятности ($P_{ud}^{(доп)}$) реализации угрозы в каждой попытке. При экспоненциальном приближении имеет место соотношение:

$$P_{ud}^{(доп)}(k,T) = 1 - \exp\left(-T \cdot \lambda_{ud} \cdot P_{ud}^{(доп)}\right), \tag{7.15}$$

откуда следует

$$P_{ud}^{(доп)} \le \frac{1}{T \cdot \lambda_{ud}} \ln\left[\frac{1}{1 - P_{ud}^{(доп)}(k,T)}\right],\ P_{ud}^{(доп)}(k,T) < 1. \tag{7.16}$$

Пример 7.1. Пусть атаки на ИС проводятся в среднем один раз в сутки, то есть $\lambda_{ud} = 0.042 \frac{1}{час}$. Необходимо определить предельно допустимую вероятность реализации угрозы в каждой попытке, если период оценки составляет 1 год, то есть $T = 8760$ часов, в зависимости от предельно допустимой вероятности реализации угрозы $P_{ud}^{(доп)}$. Результаты расчета в виде графика приведены на рисунке 7.3.

Если количество угроз с одинаковым по содержанию несанкционированным действием составляет величину U, то по аналогии с формулами (7.10) и (7.11) предельно допустимая вероятность реализации угрозы отказа в обслуживании определяется при разных требованиях по защите от разных угроз из соотношения:

$$\prod_{u=1}^{U}\left[1 - P_{ud}^{(доп)}(k,T)\right] \ge \sqrt[N_{кр}]{\left[1 - \varepsilon_p(k) \cdot P_{func}^{(доп)}(T)\right]},\ \varepsilon_p(k) \ge 1,\ k = \overline{1,K}, u = \overline{1,U}. \tag{7.17}$$

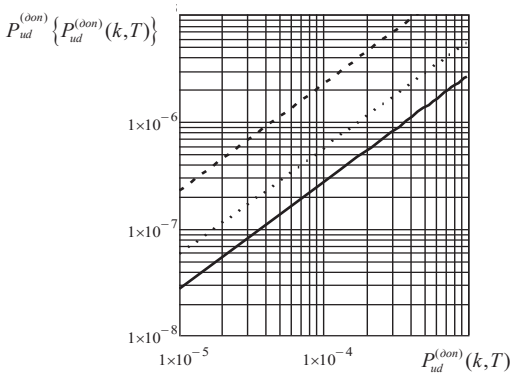


Рисунок 7.3 – Зависимость предельно допустимой вероятность реализации угрозы в каждой попытке от предельно допустимой вероятности реализации угроз за весь оцениваемый период

Отсюда без учета класса защищенности ИС

$$P_{ud}^{(\partial on)}\left(S_{kr},T\right)=1-\prod_{s=1}^{S_{kr}}\left[1-\prod_{n=1}^{N_s}P_{ud}^{(\partial on)}\left(n,T\right)\right]\le P_{func}^{(\partial on)}\left(T\right), \qquad (7.20)$$

и требования к каждому элементу $P_{ud}^{(\partial on)}\left(n,T\right)$ определяются из решения уравнения

$$\prod_{s=1}^{S_{kr}}\left[1-\prod_{n=1}^{N_s}P_{ud}^{(\partial on)}\left(n,T\right)\right]\le 1-P_{func}^{(\partial on)}\left(T\right). \qquad (7.21)$$

С учетом класса защищенности ИС в соответствии с соотношением (7.9) это уравнение преобразуется к виду

$$1-\prod_{s=1}^{S_{kr}}\left\{1-\left[P_{ud}^{(\partial on)}\left(k,T\right)\right]^{N_s}\right\}\le\varepsilon_p\left(k\right)P_{func}^{(\partial on)}\left(T\right),\ \varepsilon_p\left(k\right)\ge 1,\ k=\overline{1,K}. \qquad (7.22)$$

В этом случае так же, как и в предыдущих случаях, можно установить допустимые вероятности ($P_{ud}^{(\partial on)}$) реализации угрозы в каждой попытке (см. формулу 7.14) и предельно допустимые вероятности реализации угрозы при наличии нескольких таких угроз (см. формулы 7.15 и 7.16).

Аналогичным образом могут быть детализированы требования, предъявляемые к предельно допустимой вероятности нарушения целостности или доступности системных файлов в конкретном программно-аппаратном элементе ИС или пользовательских файлов, без которых перестает нормально функционировать прикладное ПО, что для пользователя равносильно выводу из строя самой ИС.

Пусть установлены требования по защите i-го элемента ИС заданного класса защищенности, например, одной из рабочих станций ИС, от угроз нарушения функционирования этого элемента за заданное время $P_{ud}^{(\partial on)}\left(k,T\right)$. Вывод из строя ИС возможен путем нарушения целостности или доступности:

1) хотя бы одного критически важного системного файла из некоторой совокупности таких файлов в i-м хосте ИС;

2) некоторой совокупности системных файлов из определенного набора таких совокупностей (как правило, с разным количеством системных файлов в каждой совокупности), при этом только нарушение целостности или доступности всех файлов, входящих в совокупность, приводит к нарушению функционирования ИС;

3) некоторой совокупности пользовательских файлов (исполняемых файлов прикладных программ или файлов данных, без которых прикладные программы не работают).

Если составлен перечень критически важных системных файлов в i-м хосте ИС, количество которых составляет величину F_{syst} и нарушение целостности или доступности каждого из которых приводит к выводу из строя рабочей станции (первый из указанных случаев), то по аналогии с формулами (7.9) и (7.10) применительно к любому системному файлу с номером f_{syst} имеет место неравенство:

$$\prod_{f_{syst}=1}^{F_{syst}}\left[1-P_{ud}^{(\partial on)}\left(f_{syst},k,T\right)\right]\geq 1-P_{ud}^{(\partial on)}\left(k,T\right),\ k=\overline{1,K},f_{syst}=\overline{1,F_{syst}}\ , \qquad (7.23)$$

из которого определяется допустимая вероятность нарушения целостности или доступности любого критически важного системного файла f_{syst} в ИС k-го класса защищенности.

При равных требованиях к защищенности всех критически важных файлов:

$$P_{ud}^{(\partial on)}\left(f_{syst},k,T\right)\leq 1-\sqrt[F_{syst}]{1-P_{ud}^{(\partial on)}\left(k,T\right)},f_{syst}=\overline{1,F_{syst}}\ . \qquad (7.24)$$

Если определены совокупности системных файлов (второй случай), при этом количество файлов в c-й совокупности составляет величину F_c, а количество таких совокупности составляет величину C_{syst}, то по аналогии с формулой (7.17) для i-го хоста ИС k-го класса защищенности имеет место неравенство:

$$1-\prod_{c=1}^{C_{syst}}\left[1-\prod_{f_{syst}=1}^{F_c}P_{ud}^{(\partial on)}\left(f_{syst},k,T\right)\right]\leq P_{ud}^{(\partial on)}\left(k,T\right)\ . \qquad (7.25)$$

При одинаковых требованиях к защищенности каждого файла формула преобразуется к виду:

$$1-\prod_{c=1}^{C_{syst}}\left\{1-\left[P_{ud}^{(\partial on)}\left(f_{syst},k,T\right)\right]^{F_c}\right\}\leq P_{ud}^{(\partial on)}\left(k,T\right)\ . \qquad (7.26)$$

На рисунке 7.4 показана область значений, в рамках которой выбирается предельно допустимая вероятность реализации угрозы нарушения функционирования ИС за счет воздействия на совокупности системных файлов.

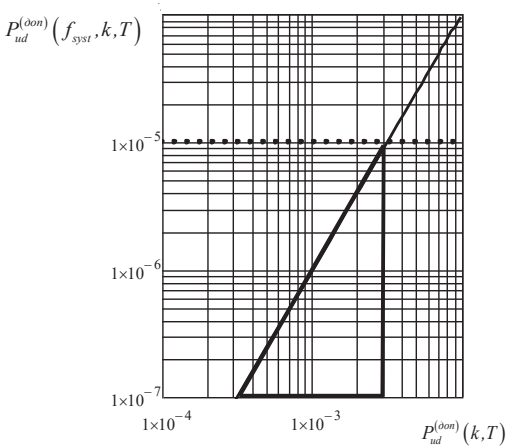


Рисунок 7.4 – Зависимость предельно допустимой вероятности нарушения целостности или доступности системных файлов от допустимой вероятности реализации угрозы нарушения функционирования системы в целом: $C_{syst}=2, F_1=2, F_2=3$

Из этого условия определяются допустимые вероятности реализации угроз относительно каждого критически важного системного файла.

Если количество угроз отказа в обслуживании, реализуемых путем нарушения целостности или доступности системных или пользовательских файлов, велико, что приводит к значительным затратам времени на проведение оценки возможностей реализации угроз, то полагается, что с единичной вероятностью будут выполняться деструктивные действия относительно этих файлов или определенных их сочетаний, когда нарушителем или инициированным им процессом получен доступ в операционную среду хоста ИС.В этом случае устанавливается предельно допустимая вероятность $P_{OS}^{(доп)}\left(k,T\right)$ получения несанкционированного доступа в операционную среду:

$$P_{OS}^{(доп)}\left(k,T\right) \le P_{ud}^{(доп)}\left(k,T\right) . \qquad (7.27)$$

Изложенный подход позволяет задавать количественные требования к эффективности защиты информации от угроз нарушения функционирования ИС, в том числе с учетом установленного для нее класса защищенности.

7.3. Формирование требований к эффективности защиты конфиденциальной информации от угроз, реализуемых путем несанкционированного доступа к ней

К конфиденциальной информации относится информация, содержащая сведения, составляющие тот или иной вид тайны: государственной, служебной, коммерческой, профессиональной (банковской, врачебной, тайны связи и др.), а также персональные данные. Вопросы оценки допустимого риска нарушения безопасности информации, содержащей сведения, составляющие государственную тайну, регулируются специальными документами и здесь не рассматриваются.

Предельно допустимая вероятность утечки иной информации ограниченного доступа должна устанавливаться обладателем защищаемой информации [7]. При этом должен проводиться анализ обрабатываемой или подлежащей обработке информации на предмет отнесения содержащейся в ней сведений к тому или иному виду тайны или персональным данным. Соответственно разделяется все множество файлов с пользовательской информации ограниченного доступа по видам тайн.

Далее в зависимости от класса (уровня) защищенности системы (класс по-прежнему обозначается буквой k) и от того, к какому виду тайны относятся сведения, раскрываемые в результате возможной утечки информации, определяются предельно допустимое значение вероятности $P_{uv}^{(\partial on)}\left(f_{conf},k,T\right)$ реализации угрозы утечки информации, соответствующей v-му виду тайны и содержащейся в файле с номером f_{conf}. Для этого сначала устанавливается предельно допустимое значение вероятности реализации угрозы применительно к классу, соответствующему самой высокой защищенности информации для одного из видов тайн (например, или коммерческой тайны, или служебной тайны, или персональным данным). При определении предельно допустимого значения указанной вероятности могут применяться несколько вариантов обоснования.

Первый вариант состоит в ориентации на предельно допустимую вероятность вывода из строя элемента ИС (рабочей станции, сервера и т.д.), при этом обладатель защищаемой от утечки сведений, составляющих тот вид тайны, для защиты которой предъявляются наиболее высокие требования (пусть это соответствует виду тайны с номером $v=1$), полагает, что предельно допустимая вероятность утечки не должна быть меньше предельно допустимой вероятности нарушения функционирования элемента ИС при каждой попытке реализации угрозы, то есть:

$$P_{u,1}^{(\partial on)}\left(f_{conf},k\right)\geq P_{ud}^{(\partial on)} \quad . \tag{7.28}$$

Затем устанавливаются коэффициенты, определяющие изменение требований применительно к иным видам тайн $\zeta_v,\ v=1,2...V$. При этом

самые высокие требования соответствуют наиболее важной информации, тогда $\zeta_1 = \zeta_{\min}$, (как правило, полагается, что $\zeta_{\min} = 1$, и $\zeta_V = \zeta_{\max} \geq 1$).

Если рассматривать требования по защите только одного файла с конфиденциальной информацией с v-м видом тайны, имеющегося в хосте ИС, для которой установлен k-й класс защищенности, то допустимая вероятность $P_{ufv}^{(\partial on)}$ того, что в каждой попытке угроза утечки информации будет реализована, определяется из соотношения:

$$P_{ufv}^{(\partial on)} = \zeta_v \cdot P_{uf1}^{(\partial on)}, \ \zeta_v > 1, v = 2,3...V, \tag{7.29}$$

при этом полагается, что в каждой попытке будет иметь место угроза утечки информации, содержащей сведения, относящиеся к виду тайны ($v = 1$), для которого устанавливаются наиболее жесткие требования по защите.

Тогда допустимая вероятность реализации угроз нарушения конфиденциальности за оцениваемый период времени T для одного файла с информацией, содержащей сведения, относящиеся к v-му виду тайны, рассчитывается при экспоненциальном приближении по формуле:

$$P_{ufv}^{(\partial on)}(k,T) \leq \varepsilon_p(k) \cdot \left[1 - \exp\left(-T \cdot \lambda_{ufv} \cdot P_{ufv}^{(\partial on)}\right)\right], \tag{7.30}$$

где λ_{uf} – интенсивность попыток реализации угрозы относительно f-го файла с информацией, содержащей сведения, относящиеся к v-му виду тайны.

Если таких файлов N_v, но в каждой попытке реализации угрозы может быть скопирован только один файл с конфиденциальной информацией, то для случая, когда нужно защитить все файлы за время T, предельно допустимая вероятность реализации угроз за это время определяется из соотношения:

$$P_{ufv}^{(\partial on)}(N_v,k,T) \leq \varepsilon_p(k) \cdot \left[1 - \exp\left(-T \cdot \lambda_{ufv} \cdot P_{ufv}^{(\partial on)}\right)\right]^{N_v}. \tag{7.31}$$

Для случая, когда нужно защитить не менее n_v из N_v файлов за время T и в каждой попытке реализации угрозы может быть скопирован только один файл с конфиденциальной информацией, формула для расчета предельно допустимой вероятности имеет вид:

$$P_{ufv}^{(\partial on)}\left(N_v,k,T\right) \leq \varepsilon_p\left(k\right) \sum_{j=n_v}^{N_v} C_{N_v}^{j} \left\{1 - exp\left[-T \cdot \lambda_{ufv} \cdot P_{ufv}^{(\partial on)}\right]\right\}^{j} \times$$
$$\times \left\{exp\left[-T \cdot \lambda_{ufv} \cdot P_{ufv}^{(\partial on)}\right]\right\}^{N_v - j}. \tag{7.32}$$

Если в каждой попытке реализации угроз копирования файлов с конфиденциальной информацией может быть скопировано более одного файла, то количество скопированных файлов зависит от времени, которое имеет нарушитель для реализации такой угрозы.

Пусть для копирования f-го файла необходимо в общем случае случайное время τ_f, при этом копирование осуществляется с вероятностью $P_{ufv} > 0$. Тогда математическое ожидание суммарного времени копирования n_v файлов определяется следующим образом:

$$\overline{\tau_{n_v}} = \sum_{f=1}^{n_v} \frac{\overline{\tau_f}}{P_{ufv}}. \qquad (7.33)$$

Вероятность копирования n_v файлов при каждой попытке при экспоненциальном приближении рассчитывается по формуле:

$$P_{ufv}(n_v, \overline{t_{copy}}) \approx \overline{t_{copy}} \Big/ \left(\overline{t_{copy}} + \overline{\tau_{n_v}}\right), \qquad (7.34)$$

где $\overline{t_{copy}}$ – среднее время, которым может располагать нарушитель для реализации угрозы.

Если $P_{ufv}\left(n_v, \overline{t_{copy}}\right) \le P_{ufv}^{(\partial on)}$, то необходимые требования выполняются, в противном случае надо принимать дополнительные меры защиты, направленные на увеличение среднего времени копирования файлов $\overline{\tau_f}$, снижение среднего времени, которое может иметь нарушитель для реализации угрозы $\overline{t_{copy}}$. При этом, если в каждой попытке нужно защитить не менее n_v из N_v файлов при каждой попытке реализации угрозы, то допустимая вероятность реализации угрозы за время T рассчитывается из соотношения:

$$P_{ufv}^{(\partial on)}\left(n_v, k, T\right) \le \varepsilon_p\left(k\right) \cdot \left\{1 - exp\left[-T \cdot \lambda_{ufv} \sum_{j=n_v}^{N_v} C_{N_v}^{n_v} \cdot \left(P_{ufv}^{(\partial on)}\right)^j \cdot \left(1 - P_{ufv}^{(\partial on)}\right)^{N_v - j}\right]\right\}. \qquad (7.35)$$

Если файлов с конфиденциальной информацией много и сложно проводить оценки возможностей реализации угроз ее утечки применительно ко всем таким файлам, то можно воспользоваться приемом, аналогичным ранее предложенному для определения предельно допустимой вероятности реализации угроз отказа в обслуживании, когда полагается, что с единичной вероятностью будет выполняться несанкционированное действие относительно любого из файлов с конфиденциальной информацией, если нарушителем или инициированным им процессом получен доступ в операционную среду хоста ИС. При этом требование задается относительно предельно допустимой вероятности получения несанкционированного доступа в операционную среду, то есть с учетом соотношений (7.29) и (7.30):

$$P_{fvOS}^{(\partial on)}\left(k, T\right) \le P_{ufv}^{(\partial on)}\left(k, T\right). \qquad (7.36)$$

Таким образом могут быть сформированы требования к эффективности защиты конфиденциальной информации от угроз, реализуемых путем несанкционированного доступа к ней.

7.4. Некоторые аспекты обоснования количественных требований к средствам защиты информации в информационных системах

Сегодня имеется ряд документов, содержащих требования к средствам защиты, таким как межсетевые экраны [109], системы обнаружения вторжений [110], средства антивирусной защиты [111], средства доверенной загрузки [112], средства контроля съемных машинных носителей информации [113]. Эти требования привязаны к классам и уровням защищенности ИС и к классам защиты, устанавливаемым для средств защиты. Сами требования установлены на основе опыта специалистов, в том числе те, которые прописаны в международных стандартах ИСО/ МЭК 15408 [20, 18, 30]. При этом они не связаны с количественными показателями требуемой эффективности защиты самих ИС, хотя качественно связь между классом (уровнем) защищенности ИС и требуемыми классами средств защиты установлена на основе экспертных оценок. Ниже предлагается подход к формированию требуемой эффективности защиты информации средствами защиты, исходя из требуемой эффективности такой защиты в ИС.

Разнообразие средств защиты информации, разрабатываемых и применяемых в ИС различных классов защищенности с различным составом и структурой построения, и реализуемых этими средствами функций защиты обусловливает необходимость учета целого ряда факторов при обосновании количественных требований к их эффективности. К таким факторам относятся следующие:

– принцип построения системы защиты информации в ИС и управления ею (централизованный, то есть с консоли администратора безопасности, децентрализованный в виде упомянутой ранее многоагентной системы защиты [27–29] или смешанный, то есть централизованно-децентрализованный). Сегодня системы защиты пока строятся только по централизованному принципу;

– принятая стратегия защиты информации в ИС (например, стратегии периметровой защиты, эшелонированной защиты, тотального контроля и т.д. [26]). Сегодня реализуется в основном стратегия эшелонированной защиты;

– класс защищенности ИС, в составе которой создается система защиты информации, и необходимость соответствующей дифференциации требований к средству защиты;

– задачи защиты, решаемые с применением средства, и реализуемые им функции, направленные на решение указанных задач;

– наличие других средств защиты, выполняющих такие же задачи по защите информации в ИС (например, при реализации эшелонированной защиты информации в ИС могут выполняться функции разграничения доступа в разных эшелонах системы защиты);

– место установки средства защиты в составе ИС и др.

С учетом этих факторов необходимо связать требуемую эффективность защиты информации в ИС с показателем влияния рассматриваемого средства защиты на эффективность защиты информации в ИС, то есть необходимо

оценить по сути, степень влияния выполнения средством одной или группы функций защиты на возможность реализации рассматриваемой угрозы безопасности информации. Сложность заключается в том, что, во-первых, степень влияния выполнения разных функций на возможность реализации угрозы, как правило, оценивается разными показателями, во-вторых, при применении средства защиты в ходе решения некоторых задач может выполняться совокупность нескольких функций защиты. Пример таких функций защиты[31] и применяемых показателей для оценки их влияния на возможность реализации угроз приведен в таблице 7.1.

Таблица 7.1 – Примеры некоторых функций защиты и показателей оценки их влияния на возможность реализации угроз безопасности информации

Содержание функции защиты	**Наименование показателя оценки влияния выполнения функции защиты на возможность реализации угрозы**
Идентификация пользователя (процесса, носителя)	Вероятность подбора идентификатора пользователя (процесса, носителя)
Аутентификация пользователя (процесса, сообщения)	Вероятность преодоления (например, подбора или обхода пароля) процедуры аутентификации пользователя (процесса, сообщения)
Обнаружение вторжения (попытки вторжения) в операционную среду	Вероятность правильного обнаружения сетевой атаки по ее сигнатуре
Обнаружение вторжения (попытки вторжения) по аномалии сетевого трафика	Вероятность правильного обнаружения сетевой атаки по аномалии сетевого трафика
Обнаружение вторжения (попытки вторжения, проникновения в операционную среду) по несоответствию действий абонента стандартному сценарию взаимодействия с абонентами сети	Вероятность обнаружения проникновения в операционную среду по несоответствию действий нарушителя абонента стандартном сценарию действий пользователя или абонента
Обнаружение вредоносной программы или попытки ее внедрения	Вероятность обнаружения вредоносной программы (попытки внедрения)
Фильтрация трафика по служебной информации в соответствии с протоколами межсетевого взаимодействия (блокирование несанкционированного доступа)	Вероятность обнаружения попытки обхода процедуры фильтрации
Трансляция сетевых адресов межсетевым экраном (прокси-сервером) для защиты от внешнего источника угрозы проникновения в операционную среду хоста ИС	Вероятность выявления сетевого адреса хоста ИС при наличии процедуры трансляции адресов межсетевым экраном
Выдача ложной информации на запрос (команду) для отвлечения нарушителя на ложный ресурс	Вероятность отвлечения нарушителя на ложный информационный ресурс

Пусть каждым d -м средством защиты выполняется определенный набор $\{\varphi_{ud}\}, d=\overline{1,D}, \varphi_{ud}=\overline{1,\Phi_{ud}}, u=\overline{1,U}$, функций защиты относительно защищаемой информации в ИС при реализации u -й угрозы. Для того чтобы оценить, как влияет выполнение функции защиты на риск реализации угроз, необходимо провести моделирование процесса реализации данной угрозы с использованием аппарата марковских, полумарковских процессов или сетей Петри-Маркова (см. разделы 5 и 6).

Если функция защиты не влияет на размер возможного ущерба (что может иметь место, например, при реализации функции восстановления

[31] Следует заметить, что в ряде публикаций по аналогии со стандартом ИСО/ МЭК 15408 [18] эти функции называют функциями безопасности.

информации) или ущерб определен обладателем как неприемлемый, то показатель снижения риска $\theta_u\left(d,\varphi_{ud},t\right)$ оценивается следующим образом:

$$\theta_u\left(d,\varphi_{ud},t\right)=R_u^{(ЗИ)}\left(d,\varphi_{ud},t\right)\Big/R_u^{(0)}\left(t\right),R_u^{(0)}\left(t\right)>0, \qquad (7.37)$$

где $R_u^{(0)}\left(t\right)$ и $R_u^{(ЗИ)}\left(d,\varphi_{ud},t\right)$ – риски реализации угрозы при отсутствии d-го средства защиты и при наличии d-го средства, реализующего φ_{ud}-ю функцию защиту.

По своей сути этот показатель совпадает с относительным показателем эффективности защиты информации применительно к условиям применения d-го средства, реализующего φ_{ud}-ю функцию защиту от u-й угрозы (см. раздел 2.4). В простейших случаях в качестве таких показателей снижения уровня риска может служить, например:

– при идентификации пользователя (процесса, носителя) вероятность того, что идентификатор не будет подобран в ходе атаки;

– при аутентификации пользователя (процесса, сообщения) вероятность того, что нарушителю не удастся преодолеть процедуру аутентификации;

– при обнаружении вторжения (попытки вторжения) вероятность того, что не будет обнаружено вторжение (попытка вторжения) и т.д.

С учетом соотношения (7.37) требуемое значение показателя $\theta_u^{(треб)}\left(d,\varphi_{ud},t\right)$ снижения риска реализации угрозы d-м средством с φ_{ud}-й функцией защиту за оцениваемый период времени T определяется следующим образом:

$$\theta_u^{(треб)}\left(d,\varphi_{ud},T\right)=R_u^{(доп)}\left(d,\varphi_{ud},T\right)\Big/R_u^{(0)}\left(T\right),\ R_u^{(0)}\left(T\right)>0\,, \qquad (7.38)$$

где $R_u^{(0)}\left(T\right)$ и $R_u^{(доп)}\left(d,\varphi_{ud},T\right)$ – риски реализации u-й угрозы за время T в условиях отсутствия мер защиты и в условия применения d-го средства с φ_{ud}-й функцией защиты соответственно.

Совокупность показателей оценки влияния этих функций на риск реализации угроз можно представить в виде матрицы показателей снижения рисков реализации каждой u-й угрозы (таблица 7.2):

Таблица 7.2 – Пример матрицы значений показателей оценки влияния функций защиты на риски реализации угроз

Номер угрозы	Значение показателя влияния функции защиты φ_{ud}, выполняемой средством защиты с номером d, на снижение риска реализации u-й угрозы												
	$d=1$ для φ_{u1}				$d=2$ для φ_{u2}				▪▪▪	D для φ_{uD}			
	1_{u1}	2_{u1}	▪▪▪	Φ_{u1}	1_{u2}	2_{u2}	▪▪▪	Φ_{u2}	▪▪▪	1_{uD}	2_{uD}	▪▪▪	Φ_{uD}
$u=1$	1	0.5	▪▪▪	0.9	0.6	0.7	▪▪▪	1	▪▪▪	0.8	0.2	▪▪▪	1
$u=2$	1	1	▪▪▪	1	1	1	▪▪▪	0.2	▪▪▪	1	0.9	▪▪▪	1
▪▪▪	▪▪▪				▪▪▪					▪▪▪			
u	0.01	1	▪▪▪	1	0.8	1	0.9	1	▪▪▪	1	1	▪▪▪	1

Значение элемента этой матрицы лежит в пределах от 0 до 1 и показывает, во сколько раз уменьшается риск реализации угрозы за счет

выполнения φ_{ud}-й функции защиты, при этом значение 0 соответствует отсутствию в условиях применения функции защиты риска реализации данной угрозы, а значение 1 – отсутствию влияния функции защиты на возможности реализации данной угрозы.

Суммарное влияние всех функций защиты (Φ_u), выполняемых совокупностью D средств защиты от u-й угрозы оценивается показателем $\theta_u\left(D,\Phi_{ud},t\right)$, рассчитываемым в первом приближении следующим образом

$$\theta_u\left(D,\Phi_{ud},t\right)=\prod_{d=1}^{D}\prod_{\varphi_{ud}=1}^{\Phi_{ud}}\theta_u\left(d,\varphi_u,t\right), \tag{7.39}$$

где Φ_{ud} – количество функций защиты от u-й угрозы, выполняемых d-м средством защиты, откуда следует соотношение:

$$\theta_u^{(треб)}\left(D,\Phi_{ud},t\right)=\prod_{d=1}^{D}\prod_{\varphi_{ud}=1}^{\Phi_{ud}}\theta_u^{(треб)}\left(d,\varphi_u,t\right). \tag{7.40}$$

Для ориентировочной оценки требуемых показателей влияния каждой функции защиты каждого средства защиты может быть использована формула:

$$\theta_u^{(треб)}\left(d,\varphi_u,t\right)\le\sqrt[D+\Phi_{uD}]{\theta_u^{(треб)}\left(D,\Phi_{uD},T\right)}. \tag{7.41}$$

Таким образом, на основе изложенного подхода можно связать между собой требуемые эффективности защиты ИС с требуемыми показателями влияния реализуемых средствами защиты функций на снижение риска реализации угроз безопасности информации в ИС. Задавая требуемую эффективность защиты ИС можно определить, средства какой эффективности требуются для данной ИС.

ЗАКЛЮЧЕНИЕ

Обеспечение эффективной защиты информации в отечественных ИС в условиях расширяющегося спектра угроз и постоянного роста риска их реализации, большая востребованеность в значительном повышении обоснованности требований к такой защите, в изыскании рациональных путей построения систем защиты с применением наиболее эффективных мер и средств и др. связано с необходимостью разработки соответствующего методического обеспечения. Основной тенденцией развития такого обеспечение является переход к количественным методам оценки эффективности и к разработке системы моделей, позволяющих рассчитывать показателя такой оценки.

В данной монографии изложены существующие подходы к оценке эффективности техничесокой защиты информации в ИС и показано, что наиболее востребованным и перспективным сегодня является подход, основанный на теории рисков. При этом рассмотрена только весьма небольшая часть проблемных вопросов, имеющих место при разработке методологии оценки эффективности. В частности, практически не затронуты вопросы оценки эффективности при реализации стратегии обмана (отвлечения на ложный ресурс), стратегии адаптивной защиты, особенно на основе применения многоагентных системы защиты и т.д. Вместе с тем, в монографии акцентировано внимание читателя на целом ряде факторов, которые необходимо учитывать при анализе рисков реализации угроз и оценке эффективности защиты информации от них, таких как:

– неопределенность и нечеткость сведений об угрозах безопасности, осведемленности, возможности, намерениях и мотивах нарушителя, составе, размещении и текущих изменениях в системной и пользовательской информации защищаемой ИС и т.д.;

– большие объемы подлежащей защите информации и высокая динамика изменения ее состава;

– необходимость учета временного фактора, то есть динамики реализации угроз безопасности информации в ИС как в условиях отсутствия, так и применения разнообразных мер защиты и др.

Отсутствие в практике анализа угроз безопасности информации количественных методов обусловлено не только сложностью учета указанных факторов и отсутствием соответствующего методического обеспечения, но и полным отсутствием исследований, направленных на создание программных комплексов и экспертных систем, в которых могли бы быть реализованы предложенные или иные показатели и алгоритмы, позволившие бы перейти к количественным методам анализа угроз безопасности информации в ИС, автоматизированной подготовки необходимых исходныхданных, обоснования требований по ТЗИ и построения эффективных систем защиты.

Авторы надеятся, что данная монография внесет определенный вклад в решение комплекса проблем, связанных с внедрением в практику указанных количественных методов.

ЛИТЕРАТУРА

1. Тарасов, А.А. Функциональная устойчивость систем обработки информации/ А.А. Тарасов. – М.: МИНИТ ФСБ России, 2009. – 184 с.: ил.

2. Требования к обеспечению защиты информации в автоматизированных системах управления производственными и технологическими процессами на критически важных объектах, потенциально опасных объектах, а также объектах, представляющих повышенную опасность для жизни и здоровья людей и для окружающей природной среды. Утверждены приказом ФСТЭК России от 14 марта 2014 г. №31.

3. Концепции управления компьютеризированными предприятиями. CIO-менеджмент на современном предприятии, http://studopedia.ru/7_56978.html.

4. Денисенко, В.В. Компьютерное управление технологическим процессом, экспериментом, оборудованием/ В.В. Денисенко. – М.: Горячая линия – Телеком, 2009. – 608с.

5. Российская Федерация. Законы. Об информации, информационных технологиях и о защите информации: Федеральный закон №149 ФЗ от 27.07.2006 г.

6. Р 50922-2006. Защита информации. Основные термины и определения.

7. Р 50.1.053-2005. Информационные технологии. Основные термины и определения в области защиты информации.

8. Руководящий документ. Гостехкомиссия России. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации – М.: Воениздат, 1992 г.

9. Руководящий документ. Гостехкомиссия России. Руководящий документ. Средства вычислительной техники защита от несанкционированного доступа к информации показатели защищенности от несанкционированного доступа к информации – М.: Воениздат, 1992 г.

10. Состав и содержание организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных. Утверждены приказом ФСТЭК России от 18 февраля 2013 г. №21.

11. Требования к защите персональных данных при их обработке в информационных системах персональных данных. Постановление Правительства Россий Федерации от 1 ноября 2012 г. №1119.

12. Требования к обеспечению защиты информации в автоматизированных системах управления производственными и технологическими процессами на критически важных объектах, потенциально опасных объектах, а также объектах, представляющих повышенную опасность для жизни и здоровья людей и для окружающей природной среды. Утверждены приказом ФСТЭК России от 14 марта 2014 г. №31.

13. Требования о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах. Утверждены приказом ФСТЭК России от 11 февраля 2013 г. №17.

14. Доктрина информационной безопасности Российской Федерации. Утверждена Указом Президента Российской Федерации от 5 декабря 2016 г. N 646.

15. ГОСТ Р 53114-2008. Защита информации. Обеспечение информационной безопасности в организации. Основные термины и определения.

16. Российская Федерация. Законы. О внесении изменений в Федеральный закон «О персональных данных. Федеральный закон N 261-ФЗ г. от 25 июля 2011 г.

17. ГОСТ ИСО 27005 – 2009 г. Информационная технология. Методы и средства обеспечения безопасности. Менеджмент риска информационной безопасности.

18. ГОСТ Р ИСО/МЭК 15408-2-2013 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные компоненты безопасности.

19. ГОСТ Р ИСО 9241-11-2010: Эргономические требования к проведению офисных работ с использованием видеодисплейных терминалов (VDT). Часть 11. Руководство по обеспечению пригодности использования.

20. ГОСТ Р ИСО/МЭК 15408-1-2012 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель.

21. Методический документ «Методика оценки угроз безопасности информации». Утвержден Федеральной службой по техническому и экспортному контролю 5 февраля 2021 г.

22. Кофман, А. Введение в теорию нечетких множеств. Пер. с франц./ А.Кофман. – М.: Радио и связь, 1982. – 432 с.

23. Борисов, А.Н. Обработка нечеткой информации в системах принятия решений/ А.Н. Борисов, А.В. Алексеев, Г.В. Меркурьева и др. – М.: Радио и Связь, 1989.

24. Корченко, А.Г. Построение систем защиты информации на нечетких множествах. Теория и практические решения/ А.Г. Корченко. – Киев: «МК-Пресс», 2006. – 390с.

25. Пегат, А. Нечеткое моделирование и управление/ А.Пегат; пер. с англ. – 2-е изд. – М.: БИНОМ. Лаборатория знаний, 2015. – 798 с.: ил. – (Адаптивные интеллектуальные системы).

26. Язов, Ю. К., Организация защиты информации в информационных системах от несанкционированного доступа: монография/ Ю. К. Язов, С.В. Соловьев. – Воронеж: Кварта, 2018. – 588 с.

27. Городецкий, В.И. Многоагентные системы (обзор)/ В.И. Городецкий, М.С. Грушинский, А.В. Хабалов. – URL: http://do.gendocs.ru/docs/index-234450.html. (дата обращения 10.09.2022 г.)

28. Швецов, А.Н. Агентно-ориентированные системы: от формальных моделей к промышленным приложениям/ А.Н. Швецов. – Издательство: Федеральное государственное учреждение «Государственный научно-исследовательский институт информационных технологий и телекоммуникаций» (ФГУ ГНИИ ИТТ «Информика»), 2008. [Электронный ресурс]. – URL: http://window.edu.ru/resource/179/56179/files/62333e1-st20.pdf. (дата обращения 10.09.2022 г.).

29. Городецкий, В.И. Многоагентные технологии комплексной защиты информации в телекоммуникационных сетях. Труды 7-й международной конференции по информационным сетям и системам/ В.И. Городецкий, И.В. Котенко, О.В. Карсаев, А.В. Хабалов. – Санкт-Петербург, октябрь, 2000, 122-134 с. [Электронный ресурс]. – URL: http://ru.wikipedia.org/wiki/Городецкий,_Владимир_Иванович.

30. ГОСТ Р ИСО/МЭК 15408-3-2013 «Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Требования доверия к безопасности.

31. Милославская, Н.Г. Управление рисками информационной безопасности: Учебное пособие для вузов/ Н.Г. Милославская, М.Ю. Сенаторов, А.И. Толстой – М.: РиС, 2014. – 130с. Теория риска.

32. Емельянов, С.В. Труды ИСА РАН: Алгоритмы. Решения. Математическое моделирование. Управление рисками и безопасностью/ С.В. Емельянов. – М.: Ленанд, 2014. – 102 с.

33. Язов, Ю.К. Основы методологии количественной оценки эффективности защиты информации в компьютерных системах/ Ю.К. Язов. – Ростов-на-Дону: СКНЦ ВШ, 2006. – 274 с.

34. ГОСТ Р ИСО/МЭК 31010-2011 Менеджмент риска. Методы оценки риска. https://hseblog.ru/kb/document/1409/files/8183/ГОСТ%20Р%20ИСОМЭК%2031010-2011%20Менеджмент%20риска.%20Методы%20оценки%20риска_Текст.pdf

35. Язов, Ю.К. Сети Петри-Маркова и их применение для моделирования процессов реализации угроз безопасности информации в информационных системах: монография/ Ю.К. Язов, А.В. Анищенко. – Воронеж: Кварта, 2020. – 173 с.

36. Петренко, С.А. Управление информационными рисками. Экономически оправданная безопасность/ С.А.Петренко, С.В.Симонов – М.: Компания АйТи; ДМК Пресс, 2004.

37. Шанкин, Г.П. Ценность информации. Вопросы теории и приложений, Г.П. Шанкин. – М.: Филоматис, 2004. – 128с.

38. Харкевич, А.А. О ценности информации. А.А. Харкевич. – Проблемы кибернетики, вып.4, 1960.

39. Стратанович, Р.Л. О ценности информации/ Р.Л.Стратанович. – Изв. АН ССр, техн. Кибернетика, 1965, №5, с.3 – 12.

40. Баранович, А.Е. «Автоматная модель интеллектуального процесса оценки ценности информации и Х-гиперграфы»/ А.Е. Баранович. – М., 1997 г.

41. Единая межведомственная методика оценки ущерба от чрезвычайных ситуаций техногенного, природного и террористического характера, а также классификации и учета чрезвычайных ситуаций. ФГУ «ВНИИ по проблемам ГО и ЧС. Федеральный центр науки и высоких технологий». М., 2004.

42. Российская Федерация. Законы. О безопасности критической информационной инфраструктуры Российской Федерации: Федеральный закон №187-ФЗ от 27 июля 2017 г.

43. Власов, М.П. Моделирование экономических процессов/ М.П. Власов, П.Д. Шимко. – Ростов на/Д: Феникс, 2005. – 409.

44. Тарасов, В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика/ В.Б. Тарасов. – М.: Эдиториал УРСС, 2002. – (Науки об искусственном.).

45. Девидсон, М. Многомерное шкалирование/ М. Девидсон. – М.: Финансы и статистика, 1987.

46. Fishburn P. Utility Theory for Decision-Making. N.Y., Wiley, 1970.

47. Батищев, Р.В. Разработка методического обеспечения комплексной оценки эффективности и обоснование требований для систем защиты информации в системах электронного документооборота: диссертация … канд. техн. наук: 051319. Батищев Роман Вячеславович. Воронежский государственный технический университет. Воронеж, 2002 г.

48. Язов, Ю.К. К оценке эффективности защиты информации в телекоммуникационных системах посредством нечетких множеств/ Р.В. Батищев, А.П. Панфилов, Ю.К. Язов. – Приборостроение. №7, 2003, с. 22-29.

49. Методический подход к комплексной оценке эффективности технической защиты информации в информационно – телекоммуникационных системах с использованием аппарата теории нечетких множеств/ Жижелев А.В., Панфилов А.П., Седых И.М., Язов Ю.К.: коллективная монография / Под ред. Е.М. Сухарева. Кн.3. – М.: Радиотехника, 2003.

50. Девянин, П.Н. Модели безопасности компьютерных систем. Управление доступом и информационными потоками: учебное пособие для вузов. – 2-е изд., испр. и доп./ Девянин, П.Н. – М.: Горячая линия – Телеком. 2016. – 338с.: илл.

51. Олифер, В.Г. Безопасность компьютерных систем/ В.Г.Олифер, Н.А.Олифер – М.: Горячая линия – Телеком, 2017.– 644 с.: илл.

52. Поспелов, Д.А. Логико-лингвистические модели в системах управления/ Д.А. Поспелов. – М.: Энергоиздат, 1981. – 232 с., ил.

53. Новосельцев, В.И. Системный анализ: современные концепции/ В.И. Новосельцев – Воронеж: Издательство «Кварта», 2002. – 320 с.

54. Язов, Ю.К. Логико-лингвистическое моделирование угроз безопасности информации в информационных системах// Ю.К. Язов, С.В. Соловьев, М.А. Тарелкин/ Вопросы кибербезопасности.2022, №4 (50), с. 13 – 23.

55. Мендельсон, Э. Введение в математическую логику/ Эллиот Мендельсон. – М.: Изд-во «Наука», Главная редакция физико-математической литературы. 1976. – 320с., илл.

56. Рудометкина, М.Н. Модель логической сети с предикатными операциями/ Международный журнал прикладных и фундаментальных исследований// М.Н. Рудометкина, Спицын В.Г. – 2014. – № 10-3. – С. 21-26; URL: http://applied-research.ru/ru/article/view?id=6019.

57. Тихонов, В.И. Марковские процессы/ В.И.Тихонов, М.А. Миронов– М.: Сов. Радио, 1977.

58. Чечулин, А. А. Построение и анализ деревьев атак на компьютерные сети с учетом требования оперативности: диссертация … канд. техн. наук: 05.13.19 / Чечулин Андрей Алексеевич; С.-Петерб. ин-т информатики и автоматизации РАН. – Санкт-Петербург, 2013. – 152 с. : ил.

59. Дойникова, Е.В. Оценка защищенности и выбор защитных мер в компьютерных сетях на основе графов атак и зависимостей сервисов: диссертация … кандид. техн. наук: 051319. Дойникова Елена Викторовна. ФГБОУН Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН). Санкт-Петербург, 2013 г.

60. Бутрик, Е.Е. Подход к определению актуальных угроз безопасности информации в автоматизированных системах управления технологическими процессами с применением банка данных угроз безопасности информации ФСТЭК России / Е.Е. Бутрик, С.В. Соловьев // Информация и безопасность. – Воронеж, 2018. – Выпуск 19 (2). – С. 203-210.

61. Степашкин, М. В. Модели и методика анализа защищенности компьютерных сетей на основе построения деревьев атак [Текст]: диссертация кандидата технических наук : 05.13.11, 05.13.19 / Степашкин, Михаил Викторович; [Место защиты: С.-Петерб.]. – Санкт-Петербург, 2007. – 196 с. : ил. Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей.

62. Котенко, И. В. Вычисление и анализ показателей защищенности на основе графов атак и зависимостей сервисов / И. В. Котенко, Е. В. Дойникова // Проблемы информационной безопасности. Компьютерные системы. – СПб. : СПбПУ, 2014. – № 2. – С. 19–36.

63. Пупков, К.А. Основы кибернетики. Теория кибернетических систем/ К.А. Пупков, Ю.И.Дегтярев, Б.Н Калини и др. Под ред. К.А. Пупкова. Учебное пособие для вузов. М., «Высш. школа», 1976.

64. Горбатов, В.А. Семантическая теория проектирования автоматов/ В.А. Горбатов. – М.: Энергия, 1979. – 264 с.

65. Глушков, В.М. Синтез цифровых автоматов/ В.М. Глушков. – М.: Физматгиз, 1962.

66. Трахтенброт, Б.А. Конечные автоматы (поведение и синтез)/ Б.А. Трахтенброт, Я.М. Бардзинь Я.М. – М.: Наука, 1970.

67. Джексон, П. Введение в экспертные системы : пер. с англ. : уч. пособие// Питер Джексон. – М.: Издательский дом «Вильямс», 2001. – 624 с. ISBN 5-8459-0150-2.

68. Новосельцев, В.И. Введение в системное проектирование интеллектуальных баз знаний/ В.П.Балан, А.В. Душкин., В.И. Новосельцев, В.И. Сумин; под ред. В.И. Новосельцева. – М.: Горячая линия – Телеком, 2016. – 107 с. ISBN 978-5-9912-0589-4.

69. Поспелов, Д.А. Логические методы анализа и синтеза схем: изд. 3-е, перераб. и доп./ Д.А. Поспелов. – М., «Энергия», 1974.

70. Коршунов, Ю.М. Математические основы кибернетики: учебное пособие для втузов/ Ю,М. Коршунов. – М., «Энергия», 1972.

71. Алгазинов, Э.К. Анализ и компьютерное моделирование информационных процессов и систем/ Э.К. Алгазинов, А.А. Сирота; под общ. ред. д.т.н. А.А.Сироты. – М.: Диалог-МИФИ. 2009. – 416.

72. Кузин, Л.Т. Основы кибернетики: в 2-х томах; т.2. Основы кибернетических моделей; учеб. пособие для вузов/ Л.Т. Кузин. – М.: Энергия. 1979. – 584 с., ил.

73. Petri C.A. Kommunikation mit Automaten. – Technische Hochschule Darmstadt, 1962.

74. Котов, В.Е. Сети Петри/ В.Е. Котов – М.: Наука. Главная редакция физико-математической литературы. 1984. – 160с.

75. Питерсон, Дж. Теория сетей Петри и моделирование систем: пер. с англ./ Дж. Питерсон. – М.: Мир, 1984. – 264 с., илл.

76. Ачасова, С.М. Корректность параллельных вычислительных процессов/ С.М. Ачасова, О.Л. Бандман. – М.: Наука, 1990. – 253 с.

77. Knuth, D. E. Semantics of context-free languages. MST 2, 1968.

78. Сети Петри. WF-сети. [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/WF-сеть.

79. Проститенко, О.В. Моделирование дискретных систем на основе сетей Петри: учебное пособие/ О.В. Проститенко, В.И. Халимов, А.Ю. Рогов. – Спб.: СПб. ГТИ(ТУ),_2017 – 69с. [Электронный ресурс]. – Режим доступа: http://sa.technolog.edu.ru/repository/mmodtp_petri.pdf

80. Таненбаум, Э. Компьютерные сети/ Э. Таненбаум. – СПб.: Питер,2002. – 848 с.

81. Сердечный, А.Л. Методическое обеспечение оценивания эффективности ложных информационных систем как средств защиты от несанкционированного доступа: диссертация … канд. техн. наук: 051319. Сердечный Алексей Леонидович. Воронежский государственный технический университет. Воронеж, 2014 г.

82. Котенко, И. В. Оценка рисков в компьютерных сетях критических инфраструктур [Текст] / И. В. Котенко, И.Б. Саенко, Е. В. Дойникова // Инновации в науке: материалы XVI международной заочной научно-практической конференции (Новосибирск, 2013). – Новосибирск: СибАК, 2013. – № 16-1. – С. 84–88.

83. Гнеденко, Б.В. Введение в теорию массового обслуживания. Издание второе, переработанное и дополненное/ Б.В.Гнеденко, И.Н. Коваленко. – М.:

Издательство "Наука", Главная редакция физико-математической литературы, 1987.

84. Сильверстов, Д.С. Полумарковские процессы с дискретным множеством состояний/ Д.С. Сильверстов. – М.: Издательство "Советское радио", 1980.

85. Королюк, В.С. Полумарковские процессы и их приложения./ В.С.Королюк, А.Ф.Турбин. – Киев.: Издательство "Наукова думка", 1976.

86. Хинчин, А.Я. Работы по математической теории массового обслуживания/ А.Я. Хинчин. Физматгиз,1962.

87. Климов, Г.П. Стохастические системы массового обслуживания/ Г.П. Климов. – М.: Наука, 1966.

88. Математические методы теории массового обслуживания//Тр. мат. института им. Стеклова АН СССР. – 1955. – Т.49.– с. 1–23.

89. Седякин, Н.М. Элементы теории случайных импульсных потоков/ Н.М. Седякин. – М.: Издательство "Советское радио", 1965.

90. Большаков, И.А. Прикладная теория случайных потоков/ И.А. Большаков, В.С. Ракошиц. – М.: Сов. радио. 1978, 248 с.

91. Игнатьев, В.М. Сети Петри-Маркова/ В.М. Игнатьев, Е.В. Ларкин. – Тула: ТулГТУ, 1994, 163 с.

92. Справочник по теории вероятностей и математической статистике/ В.С. Королюк, Н.И. Портенко, А.В. Скороход, А.Ф. Турбин – М: Наука. Главная редакция физико-математической литературы, 1985. – 640 с.

93. Тараканов, К.В. Аналитические методы исследования систем/ К.В. Тараканов, Л.А. Овчаров Л.А., А.Н. Тырышкин. – М: Изд-во «Советское радио», 1974. 240 с.

94. Кокс, Д.Р. Теория восстановления/ Д.Р.Кокс, В.Л.Смит – М.: Изд-во «Советское радио», 1967.

95. Кучер, В.А. Использование методов теории вероятности и математической статистики для оценки вероятностей обнаружения уязвимостей в информационных автоматизированных системах/ В.А. Кучер, В.С. Агранович// Материалы 7 Международной научно-практической конференции «Информационная безопасность». – Таганрог: Изд-во ТРТУ, 2005, с 130-133.

96. Тихонов, В.И. Статистическая радиотехника/ В.И. Тихонов. М.: "Сов. радио", 1966.

97. Корн, Г. Справочник по математике для научных работников и инженеров/ Г.Корн, Т.Корн. – М.: Изд-во «Наука», главная редакция физико-математической литературы, 1970 г., 720 с. с илл.

98. Хоффман, Л.Дж. Современные методы защиты информации / Л.Дж. Хоффман; пер. с англ. – М.: Сов. Радио. 1980.

99. Медведовский, И.Д. Атака на Internet/ И.Д. Медведовский, В.Д. Семьянов, В.В. Платонов; под научной редакцией проф. Зегжды П.Д. – НПО «Мир и семья-95».- Спб.:- 1997.

100. Милославская, Н.Г. Информационная безопасность открытых систем: Учебник для вузов. В 2-х томах. Том 1 – Угрозы, уязвимости, атаки и

подходы к защите/ Н.Г. Милославская, С.В. Запечников, А.И. Тостой, Д.В. Ушаков. – Изд-во: Горячая Линия – Телеком, 2006. – 536 с. ISBN: 5-93517-291-1, 5-93517-319-0.

101. WebKnacRer, A. Быстро и легко. Хакинг и антихакинг: защита и нападение/ Alex WebKnacRer; учебное пособие. – М.: Лучшие книги, 2004 – 400с. : ил.

102. Конявский, В.А. Компьютерная преступность. В 2-х томах. Т.1./ В.А. Конявский, С.В. Лопаткин. – М.: РФК-Имидж Лаб, 2006. – 560 с.

103. Петренко, С.А. Аудит безопасности Intranet/ С.А.Петренко, А.А.Петренко.– М.: ДМК Пресс, 2002. – 416 с.

104. Гаценко, О.Ю. Защита информации. Основы организационного управления/ О.Ю. Гаценко. – СПб.: Изд. дом «Сентябрь», 2001 г.

105. Казарин, О.В. Надежность и безопасность программного обеспечения: учебное пособие для баклавриата и магистратуры/ Щ.В. Казарин, И.Б. Шубиинский. – М.: Мздательство Юрайт, 2018. – 342 с.

106. Шубинский, И.Б. Функциональная надежность информационных систем. Методы анализа/ И.Б. Шубинский. – Ульяновск: Областная типография «Печатный двор», 2012. – 296 с., ил. ISBN 978-5-7572-0327-0/

107. Половко, А.М. Основы теории надежности. – 2-е изд., перераб. И доп. – СПб.: БХВ-Петербург, 2006. – 704 с.: ил. ISBM 5-94157-541-6.

108. ГОСТ Р 27.003. Надежность в технике управление надежностью. Руководство по заданию технических требований к надежности. Утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 14 декабря 2011 г. № 1490-ст.

109. Требования к межсетевым экранам. Утверждены приказом ФСТЭК России от 9 февраля 2016 г. № 9.

110. Требования к системам обнаружения вторжений. Утверждены приказом ФСТЭК России от 6 декабря 2011 г. № 638.

111. Требования к средствам антивирусной защиты. Утверждены приказом ФСТЭК России от 20 марта 2012 г. № 28

112. Требования к средствам доверенной загрузки. Утверждены приказом ФСТЭК России от 27 сентября 2013 г. № 119.

113. Требования к средствам контроля съемных машинных носителей информации). Утверждены приказом ФСТЭК России от 28 июля 2014 г. № 87.

Научное издание

Язов Юрий Константинович
Соловьев Сергей Вениаминович

Методология оценки эффективности защиты информации в информационных системах от несанкционированного доступа

Монография

Издательство «Наукоемкие технологии»
ООО «Корпорация «Интел Групп»
https://publishing.intelgr.com
E-mail: publishing@intelgr.com
Тел.: +7 (812) 945-50-63

Подписано в печать 28.12.2022
Формат 60×84/16
Бумага офсетная. Печать офсетная
Объем 16,25 печ.л. Тираж 500 экз.

ISBN 978-5-907618-36-7