Воронежский государственный университет

На правах рукописи

Вялых Александр Сергеевич

МОДЕЛИ И АЛГОРИТМЫ АНАЛИЗА И ПРОГНОЗИРОВАНИЯ НАДЕЖНОСТИ ИСПОЛЬЗОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ В УСЛОВИЯХ КОНФЛИКТНЫХ ВЗАИМОДЕЙСТВИЙ

Специальность
05.13.17 – «Теоретические основы информатики»

Диссертация
на соискание ученой степени кандидата
технических наук

Научный руководитель
доктор технических наук,
профессор А. А. Сирота

Воронеж – 2014

Оглавление

Введение

Усложнение задач, выполняемых современными информационными системами (ИС), развитие используемых в них информационных технологий (ИТ), а также возникновение условий функционирования, связанных с возможностями преднамеренных и непреднамеренных негативных воздействий (НВ), требует новых подходов к анализу и прогнозированию надежности ИС.

Под надежностью ИС понимается свойство информационной системы сохранять свои характеристики в данных условиях эксплуатации [1]. Под информационными технологиями в данной работе понимаются технологии использования компьютерных систем и телекоммуникационного оборудования для хранения, обработки, передачи и управления данными [2].

В частности, технологии хранения, обработки, передачи и управления данными (информационные технологии) реализуются в программном обеспечении (ПО) информационных систем (ИС) (то есть программное обеспечение осуществляет все вышеприведенные функции). Таким образом, вопрос о надежности использования ПО ИС, рассматриваемый в данной работе, является частным случаем более широкого вопроса о надежности использования ИТ.

Одним из важнейших аспектов исследований в области анализа и синтеза ИС является конфликтный аспект надежности использования ПО ИС, предусматривающий учет конфликтного характера информационных взаимодействий. Здесь под конфликтным взаимодействием подразумевается, с одной стороны, возможность источников негативных воздействий (ИНВ), как преднамеренных, так и непреднамеренных, оказать влияние на ИС, а, с другой стороны, работа системного администратора (персонала, обслуживающего информационную систему), направленная на сохранение работоспособности ПО ИС и предотвращения последствий данного воздействия. При этом наличие двух сторон с разными интересами и целями, разными возможностями и условиями их реализации накладывает особые, часто детерминированные, правила развития конфликтного взаимодействия [3,4,5].

Как правило, внешние негативные воздействия на ИС осуществляются за счет использования одного из классов дефектов ПО [6] (ошибок ПО [7,8]), которые в научной литературе называются уязвимостями [3,4,5,6]. Согласно [9], используя уязвимости в ПО ИС, можно нарушить конфиденциальность, доступность и/или целостность информации. При этом нарушение доступности и/или целостности информации, обычно приводит к отказам [10] в работе ИС, соответственно, можно утверждать, что используемые при этом уязвимости влияют на надежность [10] ИС.

Исходя из вышесказанного, в настоящей работе под надежностью использования ПО ИС будет пониматься сохранение работоспособности ИС и всех выполняемых ею функций в условиях наличия внутренних уязвимостей ПО и конфликтных взаимодействий. А под внутренними уязвимостями ПО – класс дефектов ПО, наличие которых может быть использовано внешними источниками негативных воздействий и привести к отказам в работе ИС, т.е., в конечном счете, оказать влияние на ее надежность.

Количество уязвимостей, обнаруживаемых в ПО, с каждым годом возрастает [11-14]. Возрастает также среднее время их устранения [11-14], и растет число эксплойтов (программ или скриптов, использующих уязвимости для негативного воздействия на ИС) [11-14]. Данная ситуация повышает возможности успешного негативного воздействия на ИС, что отражается ростом числа попыток НВ на ИС [15]. Более того, статистика показывает, что если раньше в основном преднамеренное негативное воздействие оказывалось на ИС крупных компаний, то сейчас негативному воздействию все больше и больше подвергаются ИС средних и малых компаний [15]. Соответственно, возрастают совокупные материальные, репутационные и иные потери. В связи с этим пользователи и разработчики ПО нуждаются в выработке адекватных методов оценки надежности использования ПО в ИС в условиях внутренних дефектов (уязвимостей) и конфликтных взаимодействий.

Разработка данных методов ведется уже достаточно давно, в том числе, в работах И.И. Застрожнова, В.В. Липаева, А.Ю. Щеглова, O.H. Alhazmi, S. Frei,

Y.K. Malaiya, J.D. Musa, A. Ozment, M.L. Shooman. Тем не менее, все подходы, существующие на данный момент [16-31], обладают рядом принципиальных недостатков. Подходы, используемые в государственном регулировании вопросов надежности использования информационных технологий в рамках действующих систем на территории Российской Федерации, не учитывают как динамику уязвимостей в информационных системах, так и динамику преднамеренного негативного воздействия на информационные системы. Те же подходы (не закрепленные в государственных нормативных документах), которые учитывают данные динамические процессы, моделируют их без учёта ряда важных факторов, которые проявляются именно в условиях конфликтного взаимодействия. В связи с этим, с одной стороны, остается открытым вопрос об оценке параметров, характеризующих эти процессы, то есть не ясно, каким образом с помощью этих подходов численно оценить надежность конкретных информационных систем, а с другой стороны, не ясно, насколько вообще данные методы адекватны практике реальных ситуаций.

В качестве универсальной синтетической методологии исследований в сфере надежности информационных систем и технологий целесообразно рассматривать методы математического и компьютерного моделирования динамики изменения состояний информационных процессов и систем, опирающиеся на концептуальные модели конфликтных взаимодействий и позволяющие учитывать все наиболее значимые факторы таких взаимодействий [32]. Таким образом, тема диссертации, посвященная разработке моделей и алгоритмов анализа и прогнозирования надежности использования программного обеспечения в информационных системах в условиях конфликтных взаимодействий, представляется актуальной.

Тема входит в план научно-исследовательских работ ВГУ по кафедре технологий обработки и защиты информации и непосредственно связана с научным направлением Воронежского государственного университета «Математическое моделирование, программное и информационное обеспечение,

методы вычислительной и прикладной математики и их применение к фундаментальным и прикладным исследованиям в естественных науках».

Целью работы является разработка моделей и алгоритмов анализа и прогнозирования надежности использования программного обеспечения в информационных системах в условиях конфликтных взаимодействий.

Объектом исследования выступают информационные процессы и структуры, оказывающие непосредственное влияние на надежность использования ПО в ИС.

Предметом исследования является математическое и программное обеспечение для моделирования и анализа процессов с целью оценки надежности использования ПО в ИС и оценки данных характеристик.

Для достижения цели в работе рассматриваются и решаются следующие задачи.

1. Анализ наиболее важных факторов, влияющих на надежность использования ПО в ИС, определение основных требований к разрабатываемым алгоритмам и моделям анализа надежности использования ПО в ИС в условиях конфликтных взаимодействий, анализ современных подходов к оценке надежности использования ПО в ИС на предмет учета данных факторов и требований.
2. Разработка моделей функционирования информационных систем при наличии внутренних уязвимостей.
3. Разработка алгоритмов и моделей оценки надежности использования ПО в ИС в условиях конфликтных взаимодействий, учитывающих наиболее важные факторы и соответствующих основным требованиям, определенным ранее.

Методы проведения исследования. При решении поставленных в диссертации задач использовались аппарат теории вероятностей и математической статистики, модели и методы теории систем массового обслуживания, математический аппарат цепей Маркова, аппарат искусственных

нейронных сетей, а также технологии компьютерного имитационного моделирования.

Основные результаты, выносимые на защиту, и их научная новизна. На защиту выносятся следующие результаты, впервые достаточно подробно развитые или полученные в настоящей работе:

1. Двухэтапный нейросетевой алгоритм статистического анализа и прогнозирования нестационарных временных последовательностей, используемый для оценки динамики обнаружения дефектов (уязвимостей) программного обеспечения.

2. Математические модели динамики изменения состояний программного обеспечения с учетом возможных уязвимостей и общий алгоритм оценки надежности использования программного обеспечения.

3. Объектно-ориентированные и математические модели оценки надежности использования программного обеспечения информационных систем в динамике конфликтного взаимодействия.

4. Компьютерные имитационные модели использования программного обеспечения информационных систем в динамике конфликтного взаимодействия с коалицией внешних источников.

Научная новизна полученных результатов определяется следующим.

1. . Разработанный двухэтапный нейросетевой алгоритм статистического анализа и прогнозирования нестационарных временных последовательностей, используемый для оценки интенсивности обнаружения уязвимостей ПО, отличается применением на первом этапе специальной процедуры интерполяции экспериментальных данных в виде разложения по радиально-базисным функциям с нахождением коэффициентов разложения с использованием метода регуляризации, а на втором этапе – процедуры прогнозирования на основе комитета нейронных сетей (многослойных персептронов), обученных по интерполированным данным.

2. Предложенные математические модели и общий алгоритм оценки надежности использования ПО, основанные на представлении процесса

появления и устранения уязвимостей как процесса функционирования системы массового обслуживания, отличаются учетом зависимостей интенсивности обнаружения уязвимостей от времени, полученных по данным прогноза, учетом временных характеристик закрытия уязвимостей, а также характера действий производителя ПО и администратора информационной системы и реальных данных, которые могут быть получены из открытых источников.

3. Разработанные объектно-ориентированные и математические модели оценки надежности использования ПО информационных систем в условиях конфликтного взаимодействия в виде цепи Маркова с непрерывным временем, отличаются введением пространства состояний, учитывающих динамику обнаружения и закрытия уязвимостей и основные этапы организации негативного воздействия в дуэльных ситуациях, что позволяет повысить обоснованность оценки надежности использования программного обеспечения информационных систем в условиях конфликтных взаимодействий.

4. Предложенные компьютерные имитационные модели использования программного обеспечения информационных систем отличаются использованием формализма гибридных автоматов (карт состояний Харела) для исследования ситуативных изменений в динамике конфликтного взаимодействия, позволяют рассматривать ситуации без ограничений на характер распределения времени переходов между состояниями ПО информационной системы и для произвольной коалиции источников внешних воздействий, что дает возможность оценки надежности использования программного обеспечения в ситуациях конфликтного взаимодействия любого вида.

Научная новизна полученных результатов работы определяется также тем, что разработанные модели и алгоритмы оценки надежности использования ПО в ИС одновременно:

- позволяют учесть характер протекания реальных процессов функционирования информационных систем в условиях конфликтных взаимодействий;

- опираются не только на текущее состояние информационной системы, но и позволяют учитывать прогноз ее будущего состояния;
- учитывают наиболее важные факторы, влияющие на надежность информационных систем;
- используют данные, которые могут быть получены из открытых источников и статистики, опубликованной в сети Интернет;
- достаточно просто могут быть расширены и модифицированы под конкретные условия функционирования исследуемых информационных процессов и систем.

Достоверность результатов работы. Результаты исследований, сформулированные в диссертации, получены на основе корректного использования взаимно дополняющих друг друга теоретических и экспериментальных (имитационное моделирование, обработка данных реальной статистики уязвимостей программного обеспечения) методов исследований. Их достоверность также определяется совпадением результатов, полученных различными методами, между собой, а, в ряде частных случаев, с известными, наглядной физической трактовкой установленных закономерностей и соотношений.

Значимость для науки и практики заключается в том, что полученные модели и алгоритмы отвечают потребностям важного направления – развития методического обеспечения анализа и прогноза надежности использования программного обеспечения информационных систем в условиях конфликтных взаимодействий.

Особенностью разработанных моделей и алгоритмов является возможность их адаптации к новым условиям применения информационных технологий, а также к различным вариантам конфликтных ситуаций, без принципиальных ограничений на характер вероятностных распределений интервалов между событиями, происходящими в системе. Поэтому данные модели и алгоритмы могут быть использованы как основа для последующих исследований в области

надежности использования программного обеспечения в информационных системах.

Результаты диссертационной работы имеют практическое значение для исследования реальных информационных систем и отдельного программного обеспечения. Оценка результатов данных исследований позволит:

- пользователям информационных систем - выявить слабые места в политике обеспечения надежности (оценить работу системного администратора, выявить программное обеспечение, использование которого нежелательно, и т.п.), оценить материальные и иные риски, которым может подвергнуться информационная система, а также выработать рекомендации по их снижению;

- разработчикам программного обеспечения - оценить надежность использования их продуктов, выявить наиболее уязвимые из них, и, соответственно, рациональнее распределить финансовые и иные ресурсы при поддержке уже существующего программного обеспечения и разработке нового;

- организациям, осуществляющим аттестацию информационных систем и сертификацию программного обеспечения – точнее оценить реальные процессы функционирования информационных систем в условиях конфликтных взаимодействий, выработать на основе разработанных моделей и алгоритмов новую методологию, более полно учитывающую данные процессы.

Соответствие диссертации паспорту научной специальности. Диссертация соответствует специальности 05.13.17 – «Теоретические основы информатики» по следующим областям исследований:

- разработка и анализ моделей информационных процессов и структур (п. 2 паспорта специальности 05.13.17);

- разработка методов обеспечения высоконадежной обработки информации и обеспечения помехоустойчивости информационных коммуникаций для целей передачи, хранения и защиты информации; разработка основ теории надежности и безопасности использования информационных технологий (п. 11 паспорта специальности 05.13.17);

Реализация результатов работы. Полученные в диссертации результаты реализованы в департаменте связи и массовых коммуникаций Воронежской области при оценке надежности работы удостоверяющего центра правительства Воронежской области, а также в Воронежском государственном университете при выполнении исследований по гранту РФФИ в рамках научного проекта № 13-01-97507 р_центр_а.

Апробация работы. Основные положения диссертационной работы докладывались и обсуждались: на XII, XIII, XIV Международных научно-технических конференциях «Кибернетика и высокие технологии XXI века» (Воронеж) в 2011, 2012 и 2013 годах; на XI, XII, XIII Международных конференциях «Информатика: проблемы, методология, технологии» (Воронеж) в 2011, 2012 и 2013 годах; на X Международной научно-технической конференции «Физика и технические приложения волновых процессов» (Самара) в 2011 году.

Публикации и личный вклад автора. По теме диссертации опубликовано 11 работ [33-43], из них 4 работы – в изданиях, рекомендованных ВАК. В совместных работах соавторам принадлежит постановка задачи и определение направления исследований, автору – проведение рассуждений, необходимых для решения поставленных задач, разработка концептуальных, математических и имитационных моделей информационных процессов, обоснование и разработка алгоритмов анализа данных, вывод формул для оценки вероятностных характеристик надежности информационных технологий, а также анализ и интерпретация полученных результатов.

Объем и структура диссертационной работы. Диссертация состоит из введения, четырех разделов, заключения и списка литературы, включающего 94 наименования. Объем диссертации составляет 167 страниц, включая 157 страниц основного текста и 10 страниц списка литературы.

В первой главе диссертации дается общая характеристика условий функционирования современных информационных систем и технологий в условиях преднамеренных негативных воздействий, определяются наиболее важные факторы, влияющие на надежность использования ПО в ИС в условиях

конфликтных взаимодействий, а также основные требования к алгоритмам и моделям анализа надежности использования ПО в ИС в данных условиях и проводится анализ современных подходов к оценке надежности использования ПО в ИС на предмет учета данных факторов и требований. На основе полученных результатов разрабатывается технологическая схема построения моделей и алгоритмов анализа и прогнозирования надежности использования ПО в ИС в условиях внутренних уязвимостей (дефектов) и преднамеренных негативных воздействий.

Во второй главе дается описание разработанного нейросетевого алгоритма прогнозирования интенсивности обнаружения уязвимостей (дефектов) в ПО, обосновывается преимущество его прогностических способностей над существующими аналитическими моделями обнаружения уязвимостей, описываются разработанные математическая модель динамики уязвимостей (дефектов) в ПО, а также математические модели функционирования информационной системы в условиях конфликтных взаимодействий. Приводится общий алгоритм анализа вероятностных характеристик надежности использования ПО в ИС без учета характера негативных воздействий, основанный на ранее полученном нейросетевом алгоритме прогноза интенсивности обнаружения уязвимостей в ИС и математических моделях динамики уязвимостей в ПО и функционирования ИС в целом.

В третьей главе описываются разработанные объектно-ориентированные модели конфликтного взаимодействия ИС и источника негативных воздействий, использующие аппарат языка UML, математические модели конфликтного взаимодействия ИС и ИНВ на основе цепей Маркова и компьютерные имитационные модели конфликтного взаимодействия ИС и ИНВ, реализованные в интегрированной среде Matlab+Simulink+Stateflow, а также общий алгоритм анализа вероятностных характеристик надежности использования ПО в ИС в условиях преднамеренных негативных воздействий, использующий данные модели и результаты, полученные во 2-й главе.

Четвертая глава посвящена практическому применению результатов диссертационной работы, а именно, в ней на основе предложенных моделей и алгоритмов оценки надежности ПО выполнены исследования для базовых элементов типовой ИС удостоверяющего центра и типовой ИС пользователя удостоверяющего центра и предложены рекомендации для повышения их надежности.

Автор выражает глубокую признательность научному руководителю проф. А.А. Сироте за постоянное внимание и руководство, а также всему коллективу кафедры технологий обработки и защиты информации ВГУ за оказанную поддержку.

Глава 1. Общая характеристика проблемы обеспечения надежности информационных систем и технологий при наличии внутренних уязвимостей и взаимодействий конфликтного характера

1.1. Характеристика условий функционирования современных информационных систем и технологий

ПО, установленное в ИС, представляет собой операционную систему, утилиты и различные прикладные программы, в том числе средства защиты информации (СЗИ), отличающиеся от остального ПО малым количеством уязвимостей и позволяющие закрыть доступ к остальному ПО в ИС в условиях конфликтных взаимодействий.

Успех негативного воздействия (НВ) на ИС практически полностью определяется моментом времени, в который используется уязвимость ПО. В связи с этим особо важным становится точное определение и исследование жизненного цикла уязвимостей [44-50]. Этот жизненный цикл описывается определенными событиями (или датами этих событий). В разных работах [44-50] списки таких событий различны как по количеству событий, так и по их составу, кроме того, некоторых важных событий нет ни в одном из таких списков (либо они включены в другие события). В связи с этим, здесь будет приведен список событий, определяющий жизненный цикл уязвимостей, представляющий из себя объединение уже имеющихся списков с добавлением отсутствующих событий:

1. Дата инъекции - дата, когда уязвимый код был впервые зарегистрирован в репозитории исходного кода разработчика. Если репозиторий не используется, то это - первая дата, когда уязвимый код был добавлен к сборке или скомпилирован.

2. Дата выпуска - дата общедоступного выпуска системы, которая впервые содержит определенную уязвимость.

3. Дата обнаружения - дата, когда была впервые обнаружена уязвимость.

4. Дата раскрытия - дата, когда человек или организация, обнаружившая уязвимость, впервые уведомляет о ней вендора (поставщика ПО) или специальные учреждения, занимающиеся раскрытием уязвимостей.

5. Дата публикации - дата, когда существование уязвимости делается публично известным (например, через общедоступные форумы или выпуск патча). Дата публикации уязвимости часто совпадает с датой выпуска патча, закрывающего ее.

6. Дата выпуска временного решения - дата, когда выпускается первое временное решение, описывающее, как устранить уязвимость, независимо от того, официально ли оно выпущено (от поставщика) или корректно ли оно (недостающие отказы).

7. Дата выпуска патча (обновления ПО, закрывающего уязвимость) - дата, когда выпускается первое исправление для уязвимости, независимо от того, официально ли исправление (от поставщика) или корректно ли оно (недостающие отказы).

8. Дата инсталляции патча или применения временного решения — дата, когда на ИС был установлен патч, закрывающий уязвимость, или было использовано временное решение, также устраняющее уязвимость.

9. Дата создания эксплойта (программы или скрипта, использующего уязвимость ПО для НВ на ИС) - дата, когда был выпущен первый автоматизированный эксплойт (скрипт или программа), использующий определенную уязвимость, для негативного воздействия на ИС.

Соответственно, в зависимости от того, наступило ли уже то или иное событие, уязвимости можно присвоить следующие статусы:

1. Неизвестная уязвимость: Неизвестная уязвимость существует в программном обеспечении, но еще не была обнаружена.

2. Секретная уязвимость: Секретная уязвимость была обнаружена, но тот, кто ее обнаружил, не сообщил о ней вендору (поставщику ПО), общественности или специальному учреждению, занимающемуся раскрытием уязвимостей. Если

человек, обнаруживший уязвимость – источник НВ (ИНВ), она может быть использована для негативного воздействия на ИС.

3. Раскрытая уязвимость: Раскрытая уязвимость была обнаружена, и тот, кто ее обнаружил, раскрыл информацию о ней вендору или учреждению, занимающемуся раскрытием уязвимостей.

4. Опубликованная уязвимость: Опубликованная уязвимость была обнаружена и обнародована или через патч или через общедоступный интернет-ресурс (сайт, форум и т.п.), или через средства массовой информации.

5. Уязвимость, для которой существует временное решение: Уязвимость, для которой существует временное решение – это уязвимость, для которой было создано хотя бы одно временное решение, закрывающее ее.

6. Уязвимость, для которой существует патч. Уязвимость, для которой существует патч – это уязвимость, для которой был создан хотя бы один патч, закрывающий ее.

7. Закрытая уязвимость: Закрытая уязвимость – уязвимость, которая была удалена из информационной системы посредством инсталляции патча, или же посредством применения временного решения.

8. Уязвимость, для которой существует эксплойт: Уязвимость, для которой существует эксплойт - это уязвимость, для которой был создан хотя бы один эксплойт (программа или скрипт, использующий эту уязвимость).

При этом одна уязвимость может обладать сразу несколькими статусами. Например, уязвимость может одновременно иметь патч и иметь эксплойт. Определяющим же условием для потенциальной надежности той или иной ИС являются 2 ключевых события в жизненном цикле уязвимости – обнаружение уязвимости и закрытие уязвимости. Если уязвимость еще не обнаружена или уже закрыта, то она не может быть использована ИНВ, если же она обнаружена, но еще не закрыта, то, соответственно – может.

От количества известных уязвимостей в информационной системе, от скорости их устранения, от быстроты их нахождения и легкости их использования

для негативных воздействий и от последствий такого использования зависит надежность этой информационной системы.

То есть, для того чтобы охарактеризовать условия функционирования современных информационных систем при наличии преднамеренных негативных воздействий, необходимо описать процесс того, как обнаруживаются уязвимости, как они используются ИНВ для преднамеренного НВ (ПНВ) на ИС и как они устраняются. Далее приведено описание этих процессов.

Обнаружение уязвимостей. Поиском уязвимостей в программном обеспечении занимаются ИНВ, использующие их для негативных воздействий на ИС, разработчики программного обеспечения, специальные фирмы, работающие в области безопасности информационных систем, в том числе и так называемые “Провайдеры информации о безопасности” (ПИБ), и другие заинтересованные в этом люди и организации. Кроме того, уязвимости могут быть обнаружены и случайно в процессе использования какого-либо программного обеспечения [45].

Скорость обнаружения новых уязвимостей в ПО зависит как от заранее определенных факторов, таких как: уровень проверки ПО на наличие уязвимостей до его официального выпуска и количество строк в программном коде (или размер, занимаемый ПО), так и от факторов, изменяющихся во времени, например, от популярности ПО (количество ИС, в которых это ПО используется) или от каких-то случайных факторов, характеризующих работу исследователей уязвимостей. Из последнего следует, что скорость обнаружения новых уязвимостей также зависит от времени. Ниже приведена таблица, содержащая среднегодовые скорости обнаружения уязвимостей (количество уязвимостей в месяц) в Windows XP за 10 лет [51].

Таблица 1.1 – Среднегодовая скорость обнаружения уязвимостей (количество уязвимостей в месяц) в Windows XP

Период жизни ПО, год	Среднегодовая скорость обнаружения уязвимостей, ед./месяц
1	1,5
2	3,17
3	2,75
4	5,5
5	5,25
6	9,58
7	5,17
8	9,5
9	17,08
10	21,92

То, как человек, обнаруживший уязвимость, поступит с информацией о ней, зависит от его внутренних побуждений и внешних обстоятельств. Вариантов поведения неограниченное количество, но в общем виде они сводятся к пяти (рис.1.1) [45]:

- выставить информацию об уязвимости на продажу на “черном” рынке уязвимостей;
- передать бесплатно информацию об уязвимости ИНВ (или же самому ее использовать в криминальных целях);
- самостоятельно опубликовать уязвимость;
- передать бесплатно информацию об уязвимости вендору;
- выставить уязвимость на продажу на “белом” рынке уязвимостей.

Пути возможного распространения информации об уязвимости и варианты ее возможного использования показаны на рисунке 1.1.

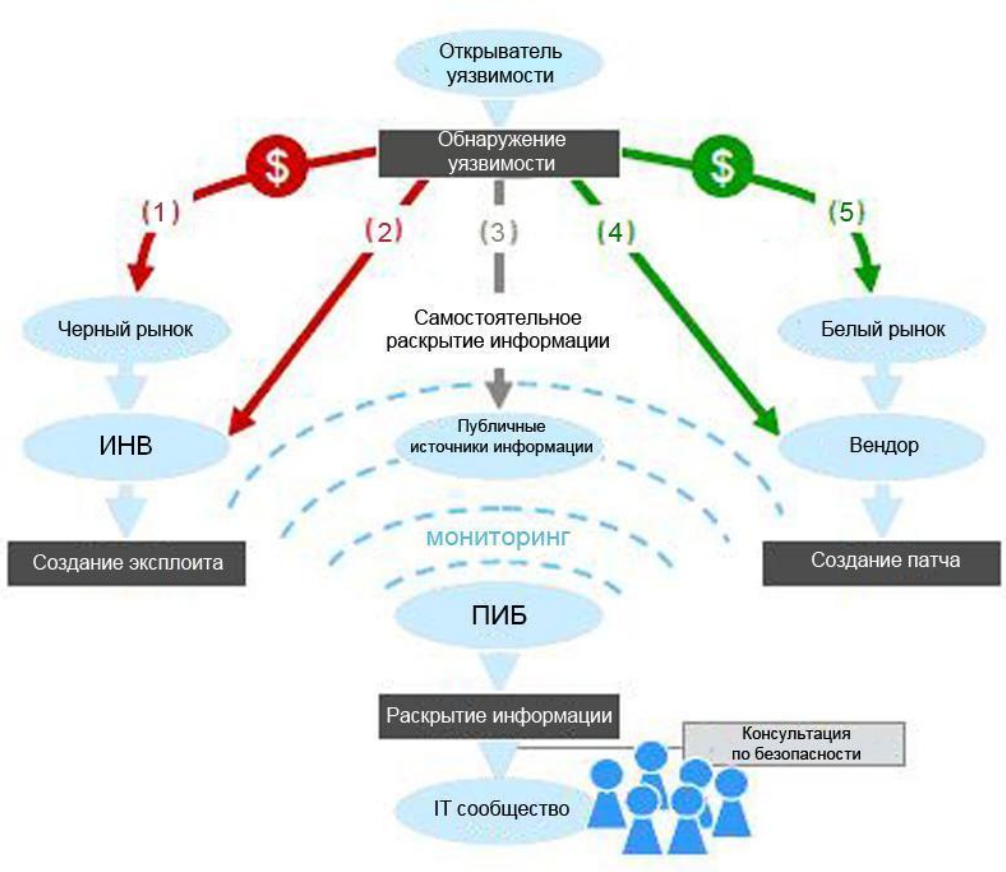

Рисунок 1.1 – Распространение информации об уязвимостях и ее возможное использование

Информация об уязвимостях может раскрываться как согласованно с вендором, так и не согласованно. В первом случае организация или человек, открывший уязвимость, сообщает о ней вендору, разработавшему ПО, в котором была найдена уязвимость, и публикует информацию о ней только после того, как вендор создаст патч, закрывающий эту уязвимость [44,45].

Информация об уязвимостях продается или подпольно ("черный рынок"), или как коммерческая услуга ("белый рынок"). Эти рынки существуют уже достаточно долго, тем не менее, коммерциализация уязвимостей до сих пор остается горячо обсуждаемой темой, главным образом потому, что она связана с вопросом раскрытия уязвимостей. Согласованному раскрытию не удается удовлетворить исследователей в области надежности, так как они считают, что информация, которую они поставляют вендорам, должна денежно компенсироваться. Поставщики же рассматривают этот процесс не иначе как

вымогательство. ИНВ, в свою очередь, не связаны юридическими или этическими соображениями и готовы инвестировать значительные суммы в подходящую информацию об уязвимостях [45]. Как сообщается в [52-54], Х. Д. Мур утверждает, что ему предложили между 60,000$ и 120,000$ за информацию о критических уязвимостях в продуктах Microsoft. Исследователи, которые намереваются продать уязвимость, боятся возможности того, что та же уязвимость будет обнаружена, исправлена и опубликована независимо. Эта угроза независимого обнаружения заставляет их продавать уязвимости самому быстрому претенденту вместо самого серьезного. Рыночную цену уязвимости определяют следующие факторы [45]:

1. Исключительность информации. Это - ключевой фактор. Когда уязвимость становится широко известной, значение информации теряет полностью свою цену или почти полностью.

2. Влияние на надежность ИС. Чем больший вред можно причинить, используя данную уязвимость, тем больше ее цена.

3. Популярность продукта. Уязвимости в популярных продуктах имеют более высокую цену, чем в непопулярных.

Для понимания роли рынка уязвимостей в контексте надежности ИС предлагается классифицировать рынки уязвимостей как “черные” и “белые” рынки.

Черный рынок. Черный рынок создан вокруг недопустимого или вредоносного использования информации об уязвимостях. Торговля открыто не ведется, и информация используется таким образом, что риск ее публикации невелик. Нехватка доверия между продавцом и покупателями делает возможным мошенничество с обеих сторон. Соответственно, из-за природы этого бизнеса точная информация о числе и типе завершенных сделок не является полностью доступной. Только немногие исследования обеспечивают некоторое понимание работы черного рынка уязвимостей, например, отчет Symantec "Underground Economy Report" [55].

Белый рынок. Игроки на белом рынке предлагают коммерческие услуги и открыто распространяют свои политики обработки уязвимостей. Демонстрация и обеспечение того, что и у покупателей и у продавцов нет злого умысла, являются основной проблемой для игроков на коммерческом рынке уязвимостей. Покупатели белого рынка обычно покупают информацию об уязвимости, чтобы защитить своих клиентов прежде, чем уязвимость станет достоянием общественности, и для информирования поставщика программного обеспечения, в котором были обнаружены уязвимости. Такие покупатели пытаются распространять свои понятия об этике в области обнаружения уязвимостей и просят, чтобы исследователи надежности ПО просили за свои услуги меньше денег (чем они могли бы получить на черном рынке) с обещанием, что информация не будет использоваться в криминальных целях [54].

У покупателей информации об уязвимостях на белом рынке есть следующие стимулы [45]:

1. Освещения в печати опубликованных ими уязвимостей увеличивает интерес к их коммерческим услугам.
2. Провайдеры систем обнаружения проникновения и предотвращения включают дополнительную защиту, которую клиенты воспринимают как преимущество.
3. Они предоставляют своим клиентам информацию об уязвимостях как коммерческую услугу.

Сегодня главные игроки на коммерческом рынке уязвимостей - компания iDefense [56], которая начала свою программу спонсирования обнаружения уязвимостей (VCP) в 2003, и TippingPoint [57], создавшая в 2005 году проект “нулевой день” (ZDI). ZDI TippingPoint получает в среднем запрос на продажу информации о приблизительно 40 новых уязвимостях в месяц и покупает информацию приблизительно об 1 из 10. Цены уязвимостей не раскрываются, но известно, что ZDI может выплатить главным исследователям в виде премии до 20,000$. Согласно опросу, проводимому TippingPoint, приблизительно 40 процентов главных исследователей ZDI (более 600 человек) работают в сфере

обеспечения надежности информационных технологий. Кроме того, согласно этому же опросу, 10 процентов исследователей признали, что рассмотрели бы продажу их открытий на черном рынке, если бы там им предложили больше денег [58].

Провайдеры информации о безопасности. В связи с быстрым развитием киберкриминала, фирмы и частные пользователи постоянно нуждаются в точной и проверенной информации об уязвимостях для оценки риска и защиты своих ИС. Информацию об уязвимостях можно получить на сайтах вендоров, на порталах безопасности, из специальных электронных рассылок, на конференциях по надежности и безопасности, из блогов экспертов и многих других источников. Однако, для большинства фирм и пользователей мониторинг всех этих источников для извлечения важной для них информации о безопасности неудобен с экономической и других точек зрения. Поэтому с первых лет существования сети Интернет некоторые частные и правительственные организации специализируются на сборе и публикации информации об уязвимостях. Некоторые из этих организаций создают научно-исследовательские лаборатории безопасности, продают средства обеспечения надежности (например, системы обнаружения проникновения, антивирусное программное обеспечение), или предоставляют платные услуги по обеспечению надежности и консалтинговые услуги. Эти организации проводят эффективный мониторинг различных источников информации об уязвимостях, проверяют найденный контент и публикуют их открытия как консультации по безопасности, которые описывают уязвимости в стандартизированном формате. Эти организации играют очень серьезную роль в обеспечении надежности работы ИС, и далее по тексту они будут обозначаться как провайдеры информации о безопасности (ПИБ). Ниже приведена таблица источников информации об уязвимостях, предоставляемых основными ПИБ [45].

Таблица 1.2 – Источники информации об уязвимостях и тип организации ПИБ, который их предоставляет: источник, спонсируемый государством (гос.), публичный источник (пуб.), коммерческий источник, предоставляемый за определенную плату (ком.)

Источник	Аббревиатура	Страна	Тип	Год начала сбора базы данных
US-CERT	CERT	США	гос.	1988
National Vulnerability Database	NVD	США	гос.	1997
SecurityFocus	SF	США	пуб.	1996
IBM ISS X-Force	IBM-XF	США	пуб.	1996
The Open Source Vulnerability Database	OSVDB	Международная	пуб.	2003
Secunia	Secunia	Дания	ком.	2002
FrSIRT	FrSIRT	Франция	ком.	2005
Security Tracker	SecTrack	США	ком.	2001
SecurityWatch	SecWatch	США	ком.	2004

Через службы ПИБ у общественности есть систематический доступ к своевременной, проверенной и понятной информации об уязвимостях. Доступность информации об уязвимостях от независимых организаций ПИБ оказывает важное влияние на поведение и стимулы всех участников ИТ сообщества, влияющих на надежность работы ИС [45]:

- фирмы и частные пользователи получают проверенную информацию об уязвимостях в стандартном, понятном формате, позволяющем им оценить их рискозависимость;
- информация об уязвимости, опубликованная как консультация безопасности установленным провайдером информации о безопасности, практически не может быть проигнорирована вендором, в ПО которого была обнаружена уязвимость;
- ПИБ играют крайне важную роль в процессе согласованного раскрытия уязвимостей.

Использование уязвимостей для организации преднамеренных негативных воздействий на ИС. В общем случае реализация преднамеренного НВ (ПНВ) на ИС включает несколько фаз [3]. Источником негативного воздействия может быть злоумышленник или независимый тестировщик системы, а также пользователь, совершающий ошибки в процессе работы системы и действующий в нештатном режиме. В наиболее общем случае таких фаз может быть 5:

1. Разведка: ИНВ собирает информацию об ИС, используя активные или пассивные средства.
2. Сканирование: ИНВ начинает активно зондировать ИС для поиска уязвимостей, которые могут быть использованы для ПНВ.
3. Получение доступа: Если уязвимость обнаружена, ИНВ использует ее, чтобы получить доступ к ИС.
4. Поддержание доступа: Как только доступ к ИС получен, ИНВ обычно занимается поддержкой доступа, чтобы реализовать цель негативного воздействия.
5. Уничтожение следов: ИНВ пытается уничтожить все доказательства осуществления ПНВ.

Не все из 5 приведенных этапов ПНВ обязательны, а с точки зрения анализа надежности оцениваемой ИС (т.к. в 4-м и 5-м случаях ИС уже взломана) представляется уместным разделить ПНВ на 3 этапа [4,5]:

- определение (инвентаризация) ПО, установленного в ИС;
- определение уязвимостей в ПО (хотя бы одной);
- определение способа использования уязвимости для негативного воздействия на ИС.

Такое представление ПНВ позволяет охарактеризовать ИНВ через среднее время, которое ему необходимо на каждый из этих этапов, при этом эти времена будут зависеть, с одной стороны, от квалификации ИНВ, а с другой стороны, от его уровня осведомленности об ИС. На практике преднамеренно негативно воздействовать на ИС может не один ИНВ, а команда ИНВ, которая может использовать разделение труда, что также должно учитываться при анализе надежности ИС.

Во время проведения со стороны ИНВ негативного воздействия уязвимость, которую он хочет использовать, может быть закрыта администратором ИС, вследствие чего ИНВ не сможет завершить его. То есть от того, с какой скоростью будут закрываться уязвимости в ПО, установленном в ИС, будет напрямую зависеть надежность этой ИС.

Также возможно, что ИС будет защищена при помощи специальных средств защиты информации (СЗИ), типа сетевых экранов. ИНВ по отношению к этим средствам может быть внешним или внутренним. Под внешним ИНВ в этом случае понимается ИНВ, которому для успешного негативного воздействия на ИС сначала нужно негативно воздействовать на СЗИ, тем самым преодолев защиту, а потом уже на саму ИС, используя уязвимости в ее ПО. Под внутренним ИНВ понимается ИНВ, который сразу непосредственно может негативно воздействовать на ИС, используя уязвимости в ее ПО. Возможен и вариант, когда используется обман ИНВ, и вместо реальной ИС подставляется подложная, и ИНВ исследуют ее до тех пор, пока не раскроют обман. Все эти возможности должны быть учтены при создании моделей и алгоритмов анализа надежности использования ПО в ИС в условиях ПНВ.

Устранение уязвимостей из ИС. Для того чтобы устранить уязвимость из информационной системы, администратору ИС требуется либо деинсталлировать

ПО, содержащее уязвимость, либо установить патч, созданный разработчиками ПО, закрывающий данную уязвимость, либо применить какое-либо временное решение, устраняющее возможность использования уязвимости. Такие временные решения публикуют как разработчики ПО, так и другие участники ИТ сообщества.

Поиск патчей, закрывающих уязвимости, и их установка могут проводиться системными администраторами как самостоятельно, так и при помощи специальных программ, которые автоматически находят обновления ПО на сайтах вендоров и устанавливают их.

Системные администраторы могут также использовать сканеры уязвимостей (такие же, как и ИНВ), для поиска уязвимостей в ИС. Кроме того администраторы ИС могут устанавливать специальное ПО для защиты ИС. В этом случае ИНВ перед взломом ИС сначала необходимо взломать это специальное ПО, используя уязвимости, которые есть в нем.

Организации также могут нанимать так называемых “этических” или “белых” ИНВ для определения слабых мест в их ИС (в том числе для нахождения уязвимостей, которые еще никем до этого не обнаруживались) и их устранения [3].

Скорость устранения уязвимостей, с одной стороны, зависит от уровня технической поддержки ПО, то есть от того, как быстро вендор создает патчи и выпускает временные решения, которые закрывают уязвимости, с другой стороны, от уровня (подготовки или выполнения своих обязанностей) системного администратора, то есть от того, как быстро он устанавливает патчи и применяет временные решения, закрывающие уязвимости, и может ли он самостоятельно (в том числе и с помощью специального ПО) находить уязвимости и придумывать временные решения для их закрытия.

Обобщённая структурная схема субъектов и основных процессов, влияющих на надёжность ИС и реализуемых в них информационных технологий (ИТ), представлена на рисунке 1.2.

Рисунок 1.2 – Обобщённая структурная схема субъектов и основных процессов, влияющих на надёжность ИС

Анализ условий функционирования современных информационных систем, схематично представленных на рисунке 1.2, показывает, что на надежность работы ИС влияют очень многие факторы, причем в граничных случаях любой из них может оказаться определяющим.

Большинство из этих факторов носят случайный характер, что должно быть обязательно учтено в моделях, описывающих поведение ИС при наличии преднамеренных негативных воздействий (ПНВ).

Также ясно, что условия функционирования современных информационных систем при наличии ПНВ не являются статическими, а изменяются во времени. Изменяется скорость обнаружения уязвимостей, изменяется скорость их устранения, изменяются возможности ИНВ (и собственно сами ИНВ), которые негативно воздействуют на ИС. В связи с этим статические модели, описывающие состояние современных ИС, будут являться заведомо недостоверными.

Процесс закрытия уязвимостей напрямую влияет на процесс использования уязвимостей для ПНВ, следовательно, модели, описывающие поведение ИС при наличии ПНВ, должны учитывать динамику не только отдельных процессов, но и динамику конфликтного взаимодействия между различными субъектами, участвующими в этих процессах.

Так как некоторые из характеристик, описывающих работу ИС, зависят от времени, то для определения того, насколько надежно будет функционировать информационная система при наличии НВ, нужно определять не текущие значения этих характеристик, а их будущие значения, за период, для которого анализируется надежность работы ИС, то есть делать прогноз в отношении этих характеристик.

Важным требованием к создаваемым алгоритмам и моделям будет наличие возможности простого усовершенствования этих алгоритмов и моделей, не требующего серьезных изменений в их концепции, так как в конкретной ситуации противостояния ИС и ИНВ могут иметь место дополнительные ограничения и возможные условия реализации конфликтного взаимодействия. Кроме того,

разрабатываемые алгоритмы и модели должны использовать параметры, для оценки которых существуют доступные источники данных.

1.2. Анализ современных подходов к оценке надежности информационных систем и технологий в условиях негативных воздействий

К настоящему времени существует большое число подходов к анализу надежности в условиях негативных воздействий как в целом информационных систем, так и отдельных информационных технологий [16-31]. В отличие от известных работ [16-18], посвященных оценке влияния на надежность любых дефектов ПО, в настоящей работе основное внимание уделено анализу подходов [19-31], так или иначе затрагивающих вопросы возможности использования уязвимостей ПО для внешних негативных воздействий, нарушающих работоспособность ИС. Эти подходы можно разделить на 3 категории.

1. Подходы, официально закрепленные нормативными документами, имеющими государственный или международный статус.
2. Подходы, используемые на рынке услуг компьютерной безопасности (в бизнесе).
3. Подходы, имеющее на данный момент только научное приложение.

При сравнении различных подходов имеет смысл принимать во внимание то, насколько полно они учитывают реальные условия функционирования ИС при наличии ПНВ: какое количество факторов они учитывают, какие это факторы и каким образом они учитываются.

В соответствии с выводами, сделанными в 1.1, подходы к анализу надежности информационных систем при преднамеренных негативных воздействиях, следует сравнивать по следующим критериям:

1. Учитывается ли динамика надежности информационной системы (то есть фактически, учитываются ли процессы или учитываются конкретные состояния ИС).
2. Какие процессы учитываются.

3. Учитывается ли недетерминированный характер процессов.

4. Какие параметры, от которых зависят процессы, учитываются .

5. Как оцениваются учитываемые параметры (оценка на основе имеющейся статистики, оценка на основе прогноза).

Сравнивая подходы к анализу надежности информационных систем при целенаправленных негативных воздействиях по 1 критерию, их можно разделить на 2 категории:

- статические подходы;
- динамические подходы.

Статические подходы. К 1 категории относятся такие подходы, которые учитывают только конкретное состояние ИС, в основном это текущее состояние [19-22]. То есть анализируются текущие условия функционирования ИС, и на основе этого анализа делается оценка о надежности ИС, при этом предполагается, что текущие условия функционирования ИС меняться не будут, а если они все-таки будут изменяться, то эти изменения будут санкционированы администраторами ИС, вследствие чего они смогут при таких изменениях оперативно оценить надежность ИС в новых условиях. Главная проблема такого подхода заключается в том, что далеко не все изменения в условиях функционировании ИС зависят от ее администраторов. Как было показано в 1.1, надежность ИС зависит от 3 процессов: от обнаружения уязвимостей в ПО, от использования этих уязвимостей ИНВ для ПНВ на ИС и от закрытия уязвимостей, а также от динамики конфликтного взаимодействия между процессами закрытия уязвимостей и процессом ПНВ на ИС. Администраторы ИС могут влиять только на процесс закрытия уязвимостей, и то для установки патча или применения временного решения, закрывающего уязвимость, администраторам необходимо иметь в наличии этот патч или временное решение, а их наличие почти целиком и полностью зависит от вендора, выпускающего ПО, в котором была найдена уязвимость (хотя, конечно, если администратор ИС обладает очень высокой квалификацией, он и сам может разработать временное решение, устраняющее уязвимость).

То есть, по сути, статические подходы обладают следующими недостатками:

- не учитывают динамику процессов обнаружения и закрытия уязвимостей в ИС;
- не учитывают динамику ПНВ на ИС;
- не учитывают динамику конфликта между системным администратором, закрывающим уязвимости, и ИНВ, пытающимся осуществить ПНВ на ИС.

В качестве примера можно привести следующую ситуацию. Допустим, на данный момент в ИС отсутствуют какие-либо известные уязвимости, на основании чего делается вывод, что ИС полностью надежна и защищена, но через некоторое время в ПО, установленном на ИС, обнаруживают новые уязвимости. Безусловно, если эти уязвимости будут опубликованы, то администраторы смогут узнать об их существовании и сделать переоценку состояния ИС, но, во-первых, далеко не все уязвимости публикуются сразу после своего открытия, а во-вторых, новые уязвимости в некотором ПО открываются настолько часто, что оценка конкретного состояния ИС в таких условиях будет попросту бессмысленна.

Несмотря на этот очевидный недостаток, такие подходы часто используются для государственного регулирования вопросов надежности ИС. В качестве примера можно привести методику определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных, утвержденную ФСТЭК России [19], приложения приказа Федеральной службы безопасности Российской Федерации от 27 декабря 2011 г. N 796 "Об утверждении Требований к средствам электронной подписи и Требований к средствам удостоверяющего центра" [20] и международный стандарт ISO 15408, принятый на данный момент в России как ГОСТ Р ИСО/МЭК 15408-2008 [21,22].

Кроме нормативных документов, используемых в государственном регулировании сферы надежности ИТ, статические подходы также реализуют

многие методики и программные продукты для оценки рисков, используемые в бизнесе. К таким методикам, например, относятся [22-28]:

- методика CRAMM;
- методика FRAP;
- методика OCTAVE.

Динамические подходы. Динамические подходы [29-31], в отличие от статических, рассматривают характеристики, описывающие процессы, влияющие на надежность ИС, а не характеристики, описывающие конкретные состояния ИС. В качестве примера можно привести модель, описывающую динамику появления уязвимостей в ИС, основанную на теории массового обслуживания [29], модель конфликта ИНВ и ИС [30] и модель оценки надежности системы защиты информации от несанкционированного доступа [31]**.** Рассмотрим их подробнее.

Модель, описывающая динамику появления уязвимостей в ИС, основанная на теории массового обслуживания. В [29] предлагается представить процесс появления новых уязвимостей и их устранения в виде работы системы массового обслуживания (СМО), на вход которой поступает пуассоновский поток заявок (уязвимостей) с интенсивностью λ, и далее СМО обслуживает эти заявки (устраняет уязвимости) с интенсивностью μ. Кроме того, предполагается, что работа над устранением каждой уязвимости начинается сразу же после ее обнаружения, соответственно, данная СМО имеет бесконечное число каналов обслуживания. В данных предположениях вероятность того, что в системе отсутствуют уязвимости, получилась равной [29]:

$$P(0) = \frac{1}{1 + \sum_{n=1}^{\infty} \frac{1}{n!} \left(\frac{\lambda}{\mu} \right)^n} \tag{1.1}$$

Можно показать, что с учетом формулы [59]:

$$e^x = 1 + \sum_{n=1}^{\infty} \frac{x^n}{n!}, \tag{1.2}$$

выражение (1.1) принимает вид:

$$P(0) = e^{-\frac{\lambda}{\mu}}. \qquad (1.3)$$

Данные для оценки параметров модели [29] интенсивности открытия уязвимостей λ и интенсивности закрытия уязвимостей μ предлагается брать из текущей статистики [29], следовательно, подход, предложенный в [29], не учитывает, что параметры процессов обнаружения и закрытия уязвимостей со временем изменяются, и для наиболее точного анализа надежности ИС необходимо осуществлять прогноз для этих параметров на период оценки. Кроме того, данный подход не учитывает зависимость надежности ИС от характеристик ИНВ, которые могут осуществлять негативные воздействия на эту ИС (в том числе количество ИНВ и разделение труда между ними).

Модель конфликта ИНВ и ИС, предложенная в [30], представляет собой случайный полумарковский процесс (рис. 1.4), построенный на основе концептуальной модели конфликта ИС – ИНВ (рис. 1.3)

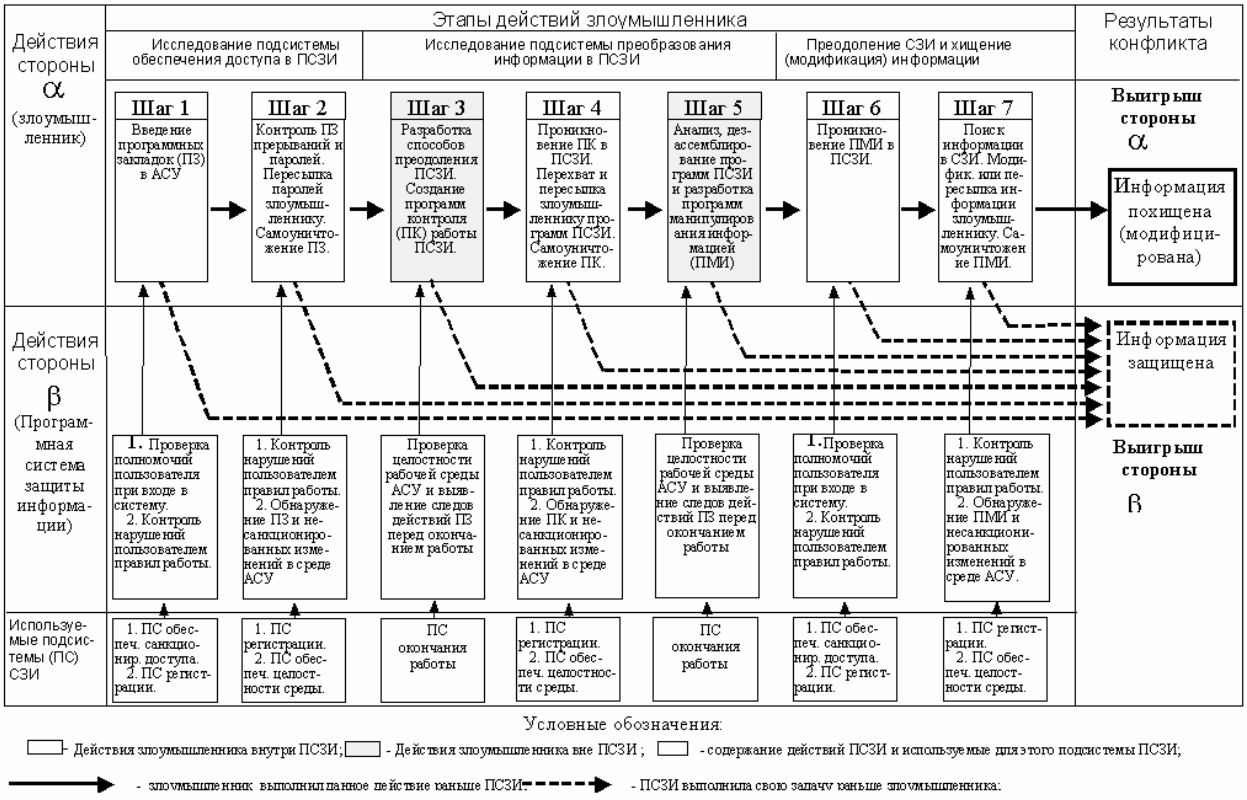

Рисунок 1.3 – Концептуальная модель конфликта ИС – ИНВ

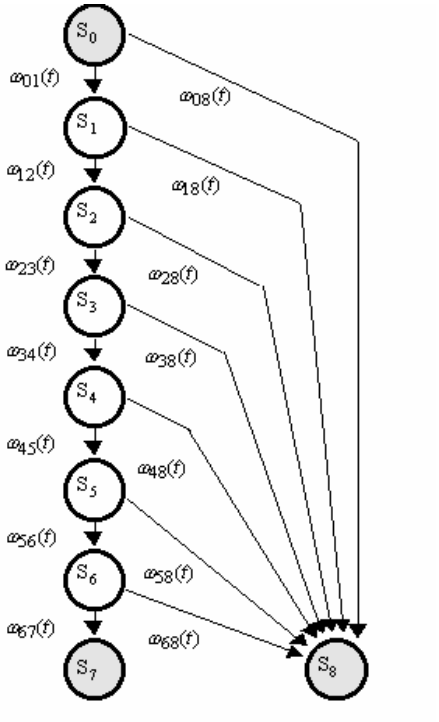

Рисунок 1.4 – Полумарковский процесс, описывающий конфликт ИНВ и ИС

Состояния данного процесса отражают этапы целенаправленного негативного воздействия ИНВ на ИС [30]:

S0 - начальное состояние процесса;

S7 - конечное состояние процесса, соответствующее выигрышу стороны α (информация модифицирована ИНВ);

S8 - конечное состояние процесса, соответствующее выигрышу стороны β (информация защищена);

S1 , S2 .. S6 - промежуточные состояния процесса, соответствующие успешному выполнению ИНВ соответствующих шагов по доступу к информации.

Переходы между состояниями характеризуются плотностями вероятности $\omega_{01}(t)$, $\omega_{12}(t)$, $\omega_{23}(t)$, $\omega_{34}(t)$, $\omega_{45}(t)$, $\omega_{56}(t)$, $\omega_{67}(t)$, $\omega_{08}(t)$, $\omega_{18}(t)$, $\omega_{28}(t)$, $\omega_{38}(t)$, $\omega_{48}(t)$, $\omega_{58}(t)$, $\omega_{68}(t)$ [30].

Далее в [30] показано, что решая систему уравнений для случайного полумарковского процесса, можно определить вероятности выигрыша сторон α и

β, которые будут соответствовать вероятностям того, что ИНВ модифицирует информацию, или же не сумеет этого сделать.

В отличие от предыдущего описанного динамического подхода к анализу надежности информационных систем в условиях внутренних уязвимостей и преднамеренных негативных воздействий, данный подход позволяет учесть характеристики ИНВ, который может негативно воздействовать на ИС, но при этом не учитывает зависимость возможности преднамеренного негативного воздействия на ИС от наличия уязвимостей и соответственно динамику уязвимостей в ИС. Кроме того, представляется нецелесообразным выбор данных этапов преднамеренного негативного воздействия. Эксперты в области компьютерной безопасности и сами ИНВ разделяют процесс преднамеренного негативного воздействия иным образом, описанным в [4,5], поэтому статистику, с помощью которой можно было бы оценить плотности вероятности переходов между состояниями процесса конфликта ИНВ и ИС, описанного в [30], невозможно где-либо найти или же получить самостоятельно. И, наконец, данный подход не учитывает ни атаки на отказ в обслуживании, ни возможности восстановления ИС после того, как ИНВ модифицирует информацию (например, информацию можно восстановить из резервной копии, хранящейся в другой ИС).

Модель надежности системы защиты информации (СЗИ) от несанкционированного доступа (НСД), описанная в [31], представляет из себя систему массового обслуживания с состояниями, полученными изменением трех признаков: $u \in \{0,1\}$ - система не уязвима (уязвимости не известны) или система с известной уязвимостью; $v \in \{0,1\}$ - СЗИ работоспособна или отказ; $k \in \{0,1\}$ - есть попытка НСД или нет. СЗИ может быть представлена следующим образом (рис. 1.5).

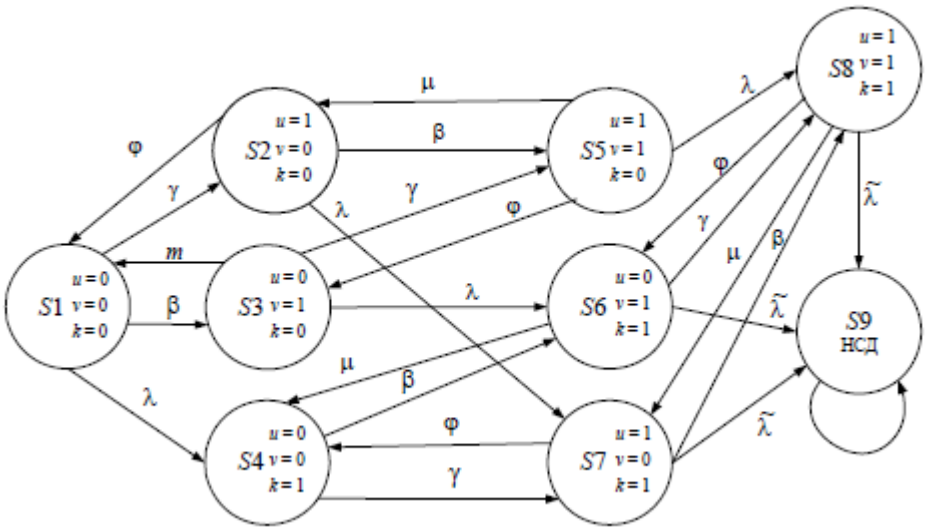

Рисунок 1.5 – Модель надежности системы защиты информации от НСД

На рис. 1.5 $S_i (i \in 1,...,9)$ состояния системы получены изменением признаков u, v и k, причем за один переход может измениться только один признак. Например, состояние $S5$ $(u = 1, v = 1, k = 0)$ - уязвимая система защиты информации отказала, но попытки НСД нет. λ - интенсивность попыток НСД; β - интенсивность отказов СЗИ; γ - интенсивность нахождения уязвимостей в СЗИ; μ - интенсивность проверок функционирования СЗИ с приведением ее в рабочее состояние; φ - интенсивность устранения уязвимостей. После попадания в поглощающее состояние $S9$, т.е. в состояние успешной реализации НСД, система больше не возвращается в другие состояния, следовательно, надежность СЗИ от НСД в рамках этой модели характеризуется временем пребывания системы в состояниях $S1 - S8$.

Преимуществом данной модели перед двумя предыдущими является то, что в ней одновременно учитывается и динамики уязвимостей в ИС (в данном случае в СЗИ) и зависимость надежности ИС от процесса негативного воздействия на ИС – в данном случае от интенсивности попыток НСД. При этом в качестве недостатка можно отметить, что данная модель предполагает только 2 состояния, характеризующих наличие уязвимостей в системе: есть известная уязвимость и

нет известной уязвимости, на самом же деле в ИС может быть разное количество известных уязвимостей, и от его числа напрямую зависит успешность попыток преднамеренного негативного воздействия на ИС. Помимо этого подход, предложенный в [31], так же, как и подход [30], не учитывает, что параметры процессов обнаружения и закрытия уязвимостей со временем изменяются, и для наиболее точного анализа надежности ИС необходимо осуществлять прогноз для этих параметров на период, за который мы будем проводить этот анализ. Также недостатком является и то, что в данном подходе процесс негативного воздействия на ИС не разделен на этапы (в отличие от подхода [30]), следовательно, этот подход не позволяет учесть все характеристики ИНВ, а предлагает определить единственную обобщающую - интенсивность попыток НСД.

Все 3 описанных выше динамических подхода к анализу надежности ИС предполагают, что процессы, влияющие на надежность ИС – случайны, что, безусловно, наряду с введением динамических характеристик условий функционирования ИС, является их преимуществом перед статическими подходами. Тем не менее, несмотря на очевидные преимущества, динамические подходы на данный момент имеют в основном только исследовательское приложение. Ниже приведена таблица сравнения описанных статических и динамических подходов.

Таблица 1.3 – Сравнение статических и динамических подходов анализа надежности ИС по учитываемым факторам

Факторы \ Подходы		Методика ФСТЭК	Приложения приказа ФСБ от 27 декабря 2011 г. N 796	ГОСТ Р ИСО/МЭК 15408-2008	CRAMM	FRAP	OCTAVE	Модель динамики уязвимостей в ИС	Модель конфликта ИНВ и ИС	Модель надежности СЗИ от НСД
ИС	ПО	+	+	+	+	+	+	+	+	+
	Наличие СЗИ	+	+	+	+	+	+	+	+	+
	Наличие средств обмана	-	-	-	-	-	-	-	-	-
Уязвимости	Зависимость угроз от наличия уязвимостей	+	-	+	-	-	+	+	-	+
	Динамика обнаружения и закрытия уязвимостей	-	-	-	-	-	-	+	-	+/-
	Прогноз скорости обнаружения уязвимостей	-	-	-	-	-	-	-	-	-
Обслуживание ИС	Уровень вендора	-	-	-	-	-	-	+	-	-
	Уровень системного администратора	-	-	-	-	-	-	-	+	+
ИНВ	Квалификация	-	+	+	-	-	-	-	+	+
	Осведомленность об ИС	+/-	+	+	-	-	-	-	+	+
	Количество ИНВ	-	-	-	-	-	-	-	-	-
	Разделение труда	-	-	-	-	-	-	-	-	-
	Этапы ПНВ	-	-	+	-	-	-	-	+	-
	Динамика ПНВ	-	-	-	-	-	-	-	+	+
Динамика конфликта ИНВ и ИС		-	-	-	-	-	-	-	+	-

1.3. Технологическая схема построения моделей и алгоритмов анализа и прогнозирования надежности информационных систем и технологий

Решение задачи создания моделей и алгоритмов анализа надежности информационных систем в условиях внутренних уязвимостей и преднамеренных негативных воздействий включает 3 последовательных этапа (см. рис. 1.6).

1. Разработка описательных моделей надежности ИС и ИТ в условиях внутренних уязвимостей и НВ. В ходе этого этапа исследуются процессы и субъекты, влияющие на надежность ИС в условиях внутренних уязвимостей и НВ, и их основные параметры, характеристики и возможности. Ключевым элементом здесь является описательная модель жизненного цикла уязвимостей информационных технологий, реализуемых в ИС, которая определяет все основные взаимосвязи и характеристики субъектов, а также динамику исследуемого процесса. Результаты этого этапа позволяют определить основные требования к разрабатываемым моделям.

2. Разработка математических моделей функционирования ИС в условиях внутренних уязвимостей и НВ. В ходе этого этапа на основе описательных моделей создаются модель динамики уязвимостей в ПО и модели функционирования ИС в условиях внутренних уязвимостей (без СЗИ и с СЗИ), использующие аппарат теории массового обслуживания. Далее определяются статистические и динамических параметры учитываемых исходных данных и разрабатываются методы (методики) и алгоритмы оценки этих параметров, в том числе: оценки интенсивности обнаружения уязвимостей, оценки интенсивности создания вендором патчей и оценки квалификации системного администратора. В итоге на основе созданных моделей и алгоритмов разрабатывается алгоритм анализа вероятностных характеристик надежности использования ПО в ИС без учета характера НВ (преднамеренное или непреднамеренное).

3. Разработка математических и имитационных моделей конфликта ИНВ и ИС. В ходе 3 этапа рассматриваются 4 варианта конфликта ИС и ИНВ (конфликт ИС без СЗИ с одним ИНВ, конфликт ИС с СЗИ с одним ИНВ,

конфликт ИС без СЗИ с коалицией ИНВ без инсайдера, конфликт ИС без СЗИ с коалицией ИНВ с инсайдером) и для каждого из этих вариантов создается концептуальная объектно-ориентированная модель конфликта, описывающая основные состояния, в которых могут находится субъекты конфликта, и переходы между ними, а на ее основе - математическая модель, использующая те ли иные вероятностные описания динамики конфликта, и компьютерная имитационная модель, реализованная в интегрированной среде Matlab+Simulink+Stateflow, обеспечивающая адекватный учет исходных концептуальных и функциональных объектных представлений. Далее на основе моделей и алгоритмов, созданных на 2 и 3 этапе, разрабатывается алгоритм анализа вероятностных характеристик надежности использования ПО в ИС в условиях ПНВ.

4. **Оценка достоверности разработанных моделей.** На четвертом этапе, исходя из результатов моделирования с использованием математических и имитационных моделей, производится оценка достоверности разработанных моделей.

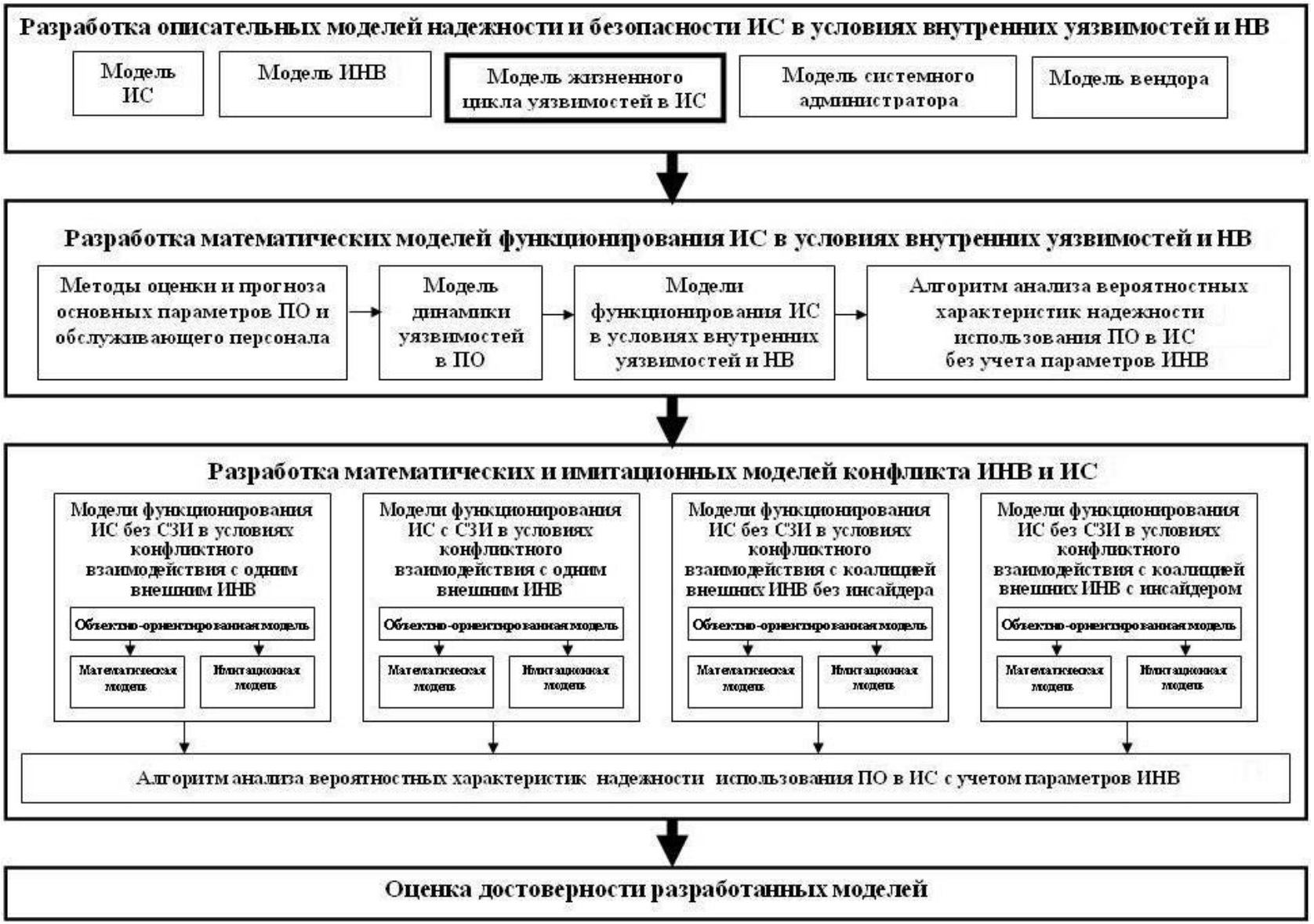

Рисунок 1.6 – Технологическая схема решения задачи создания моделей и алгоритмов анализа надежности информационных систем в условиях внутренних уязвимостей и преднамеренных негативных воздействий

Выводы по главе

1. Надежность ИС и реализуемых в них ИТ зависит от большого количества факторов, причем, в пограничных случаях любой из этих факторов может оказаться доминантным.

2. Существующие на данный момент подходы к анализу надежности ИС и ИТ в условиях наличия внутренних уязвимостей и НВ можно разделить на статические и динамические. Динамические подходы являются более предпочтительными, но и эти подходы имеют ряд недостатков, не позволяющих учитывать все наиболее значимые факторы, влияющие на надежность ИС.

3. Основными требованиями к разрабатываемым алгоритмам и моделям анализа надежности ИС и ИТ в условиях внутренних уязвимостей и НВ является: учет случайного характера факторов, влияющих на надежность ИС и ИТ; учет динамики отдельных процессов, влияющих на надежность ИС и ИТ, и динамики конфликта между различными субъектами, участвующими в этих процессах; доступность источников данных для оценки параметров, влияющих на надежность ИС и ИТ, и реализация моделей и алгоритмов прогнозирования этих данных; возможность усовершенствования используемых моделей и алгоритмов, не требующего серьезных изменений в общей схеме исследования надежности на основе применения принципов объектно-ориентированного подхода.

Глава 2. Математические модели функционирования информационных систем при наличии внутренних уязвимостей

2.1. Модели и алгоритмы статистического анализа и прогнозирования уязвимостей программного обеспечения

Одним из основных факторов, влияющих на надежность ИС, является наличие уязвимостей ПО, установленного в ИС. При этом для анализа ближайшего состояния ИС необходим прогноз в отношении динамики обнаружения уязвимостей, которые можно использовать для нарушения надежности ИС. Для нарушения надежности ИС используются только те уязвимости, с помощью которых можно нарушить целостность и доступность информации в ИС. На данный момент существует ряд аналитических моделей обнаружения уязвимостей (АМОУ), позволяющих прогнозировать их динамику обнаружения [6]:

- Термодинамическая модель Андерсона;
- Линейная модель Рескорлы;
- Экспоненциальная модель Рескорлы;
- Логарифмическая пуассоновская модель;
- Логистическая модель Алхазми-Малаийа.

Хотя исследования [6,60-68] и показали, что в большинстве случаев наилучшими прогностическими способностями обладает логистическая модель Алхазми-Малаийа [6], те же исследования показали, что в ряде случаев более эффективными оказываются другие аналитические модели. В [69,70] было показано, что число обнаруженных в ПО за месяц уязвимостей зависит не только от времени существования ПО, но и от того, какой это конкретный месяц года, что аналитические модели не учитывают. И более того, динамика обнаружения уязвимостей имеет случайную составляющую, что также не учитывается аналитическими моделями. В итоге данные модели способны прогнозировать только усредненные тенденции в изменении интенсивности обнаружения

уязвимостей, когда по факту их значительно больше [51]. Для учета вышеназванных факторов предлагается использовать подход для прогнозирования обнаружения уязвимостей на основе искусственных нейронных сетей [71,72].

В качестве исходных данных для решения задачи прогнозирования рассматривались реальные данные для операционных систем Windows XP, Windows Vista и Windows Server 2003. Прогноз производился на 6 месяцев вперед. Данные сроки прогноза актуальны как для разработчиков, так и для пользователей ПО, так как позволяют в среднесрочной перспективе первым – грамотно распределить ресурсы между разработкой нового ПО и сопровождением различного старого ПО, а вторым - идентифицировать слабые места в ИС (ПО, в котором будет найдено большее число уязвимостей), позволяющие нарушить ее работу, и принять соответствующие меры. Прогноз для Windows XP осуществлялся для случаев, когда известны данные за 96, 102, 108, 114, 120 и 126 месяцев, для Windows Vista – когда известны данные за 66, 60, 54, 48, 42 и 36 месяцев, и для Windows Server 2003 – когда известны данные за 102, 96, 90, 84, 78 и 72 месяца. Для прогноза использовались данные по уязвимостям из National Vulnerability Database [51]. Анализ данных проводился в среде Matlab с использованием компонентов входящей в нее подсистемы Neural Network Toolbox.

Процесс обработки в соответствие с предлагаемым подходом предусматривает выполнение двух этапов с использованием отдельных алгоритмов.

На первом этапе осуществлялась предварительная обработка данных о ранее обнаруженных уязвимостях, полученных, как правило, в моменты времени $t^{(1)},...,t^{(P)}$ с различными интервалами между соседними моментами. Соответствующий алгоритм обработки должен обеспечить их сглаживание и интерполяцию для представления в виде непрерывной функциональной зависимости от времени. При проведении предварительной обработки

предложено осуществлять восстановление зависимости в виде взвешенной суммы радиально-базисных функций

$$F(t)=\sum_{i=1}^{K} w_i\varphi_i(t)=w^T\varphi(t),\ \varphi_i(t)=\varphi\left(\left\|t-u_i\right\|\right)=\exp\left[-\frac{\left(\left\|t-u_i\right\|\right)^2}{2{\sigma_i}^2}\right], \tag{2.1}$$

где $\varphi_i(t)$ – i-я радиально-базисная функция; u_i – центр i-ой радиально-базисной функции; σ_i – параметр влияния i-ой радиально-базисной функции; w_i – соответствующий весовой коэффициент этой функции; K – количество используемых функций.

Параметр влияния i-ой радиально-базисной функции σ_i ($i=\overline{1,K}$) выбирается исходя из правила

$$\sigma_i = C_1 du,\ C_1>0 \tag{2.2}$$

где du - минимальное расстояние между центрами радиально-базисных функций, а C_1 некоторая константа.

Количество радиально-базисных функций берется равным

$$K=\left[C_2(P-1)\right],\quad 0<C_2\le 1 \tag{2.3}$$

где P - число моментов времени, для которых рассчитываются значения радиально-базисных функций (в рассматриваемом случае - число месяцев, по которым есть данные по количеству новых обнаруженных уязвимостей), а C_2 - некоторая константа.

При вычислении коэффициентов ряда проводилось решение переопределенной системы линейных уравнений

$$Gw=d, \tag{2.4}$$

$$G=\left\|g_{p,i}\right\|,\ g_{p,i}=\left\|\varphi_i\left(t^{(p)}\right)\right\|,\ p=\overline{1,P},\ i=\overline{1,K},\ K<P$$

где G – матрица Грина, являющаяся в данном случае прямоугольной; $d=\left(d^{(1)},...,d^{(P)}\right)^T$ – целевой вектор, определяемый из исходного множества аппроксимируемых данных.

Решение данной системы линейных уравнений может быть найдено несколькими методами [73], в том числе методом наименьших квадратов (МНК) Гаусса, методом псевдообратной матрицы Мура-Пенроуза и методом регуляризации А.Н. Тихонова [73]. Последний метод имеет преимущество перед остальными, поскольку позволяет учитывать априорное решение, в качестве которого может быть использована любая АМОУ, что дает возможность объединить преимущества нейросетевого алгоритма прогнозирования и АМОУ.

Решение системы уравнений (2.2) методом регуляризации А.Н. Тихонова выглядит следующим образом.

$$w = w^{(a)} + \left(G^T G + \alpha I\right)^{-1} G^T (d - Gw^{(a)}), \qquad (2.5)$$

где $w^{(a)}$ – априорное решение (в качестве которого использовались логистическая модель Алхазми-Малаийа и линейная модель Рескорлы [10]); α – параметр регуляризации, который может быть выбран одним из стандартных методов [73-75]; I – единичная матрица размера $K \times K$.

В данном случае параметр α предлагается определять исходя из следующего условия (метод подбора): все значения восстановленной зависимости $F(t) = Gw$ должны попадать в интервал от $\left(Gw^{(a)} - \sigma^{(a)} : Gw^{(a)} + \sigma^{(a)}\right)$, где $\sigma^{(a)}$ - среднеквадратичное отклонение восстановленной зависимости $F(t) = Gw$ при $w = w^{(a)}$ от реальных данных d, при этом невязка $\|Gw - d\|$ должна быть минимальной. Выбор данного правила определения параметра α обусловлен тем, что с одной стороны предлагается доверять априорному решению (АМОУ), с другой стороны предлагается ограничить степень доверия таким образом, чтобы при небольшом отклонении от него ($< \sigma^{(a)}$) восстановленная зависимость $F(t) = Gw$ минимально отличалась от реальных данных d.

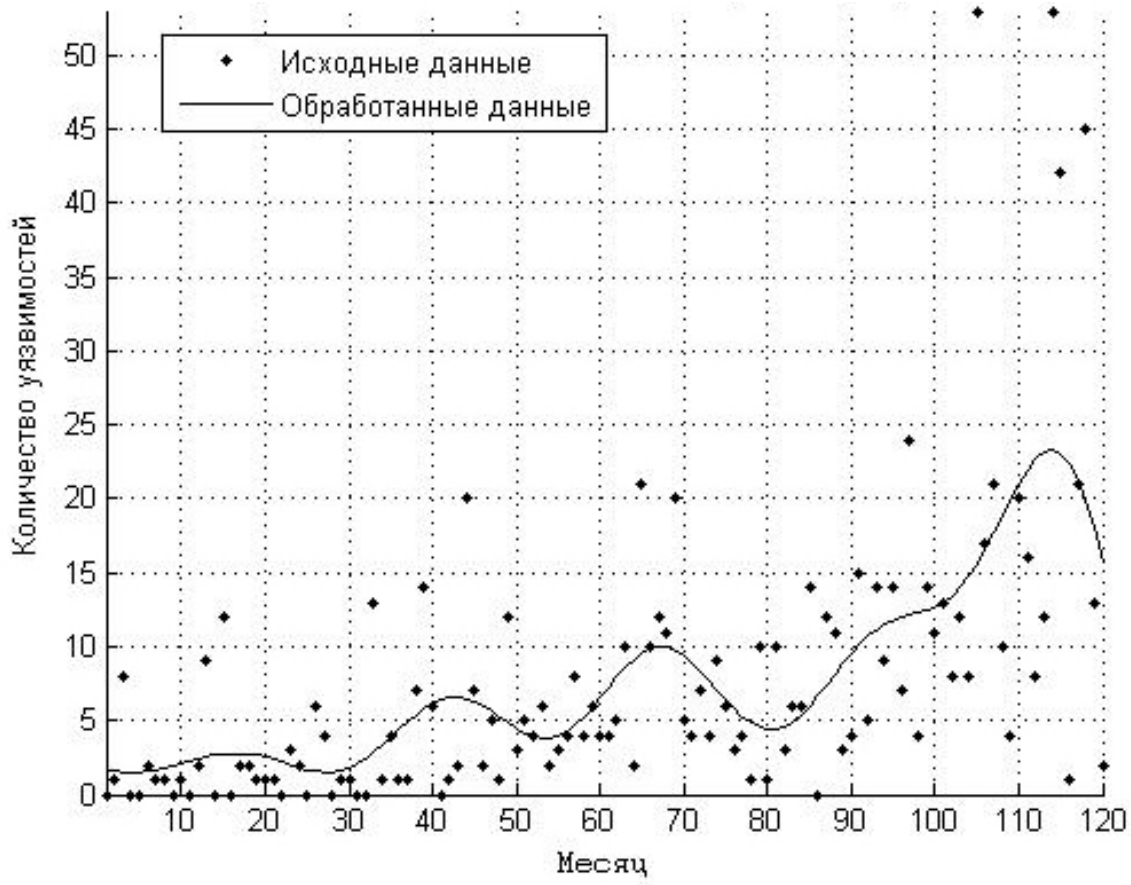

Рисунок 2.1 – Результаты сглаживания и интерполяции данных по уязвимостям (влияющим на надежность работы ИС), обнаруженным в Windows XP за 120 месяцев

На рисунке 2.1 показаны результаты предварительной обработки, реализованной в среде Matlab на основе представления функциональной зависимости в виде взвешенной суммы $K = 119$ $(C_2 = 1)$ радиально-базисных функций с параметром влияния $\sigma_i = 10{,}085$ $(i = \overline{1, K})$ $(du = 1{,}0085,\ C_1 = 10)$. Весовые коэффициенты разложения получены методом регуляризации по Тихонову с параметром регуляризации $\alpha = 3{,}61 \times 10^{-4}$.

На втором этапе обработки осуществлялось прогнозирование сглаженных и интерполированных данных с использованием комитета из 10 искусственных двухслойных нейронных сетей прямого распространения (рис 2.2) с сигмоидной функцией активации в виде гиперболического тангенса

$\text{tansig}(x) = \dfrac{2}{(1+\exp(-2x)} - 1$ для 1-го слоя и линейной функций активации $\text{purelin}(x) = x$ для 2-го слоя. Для построения прогнозирующего алгоритма проводилось обучение каждой нейронной сети при помощи функции, которая модифицирует веса и смещения в соответствии с методом шкалированных связанных градиентов (в Matlab функция trainscg), обеспечивающее восстановление нелинейной авторегрессионной зависимости разницы между очередным (прогнозируемым) значением и предыдущим значением анализируемого процесса от Nu предшествующих значений (Nu предлагается выбирать равным $Nu = \left[\dfrac{P}{3}\right]$). В качестве итогового результата прогноза за каждый месяц бралось среднее значение между прогнозами 10 нейронных сетей.

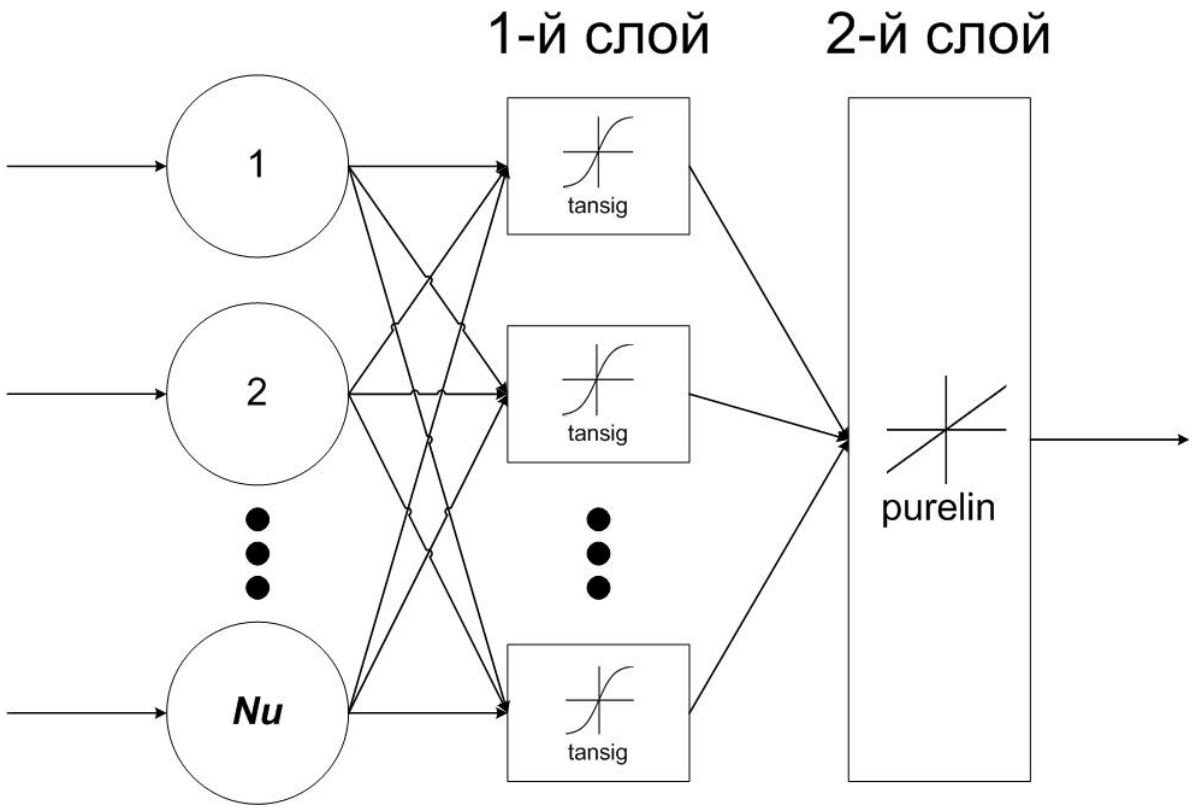

Рисунок 2.2 – Двухслойная нейронная сеть прямого распространения с сигмоидной функцией активации в виде гиперболического тангенса для 1-го слоя и линейной функций активации для 2-го слоя

На рисунке 2.3 приведены результаты прогноза для Windows XP по данным, полученным в ходе предварительной обработки, на период с 121 по 126 месяц для значений параметра $Nu = 40$.

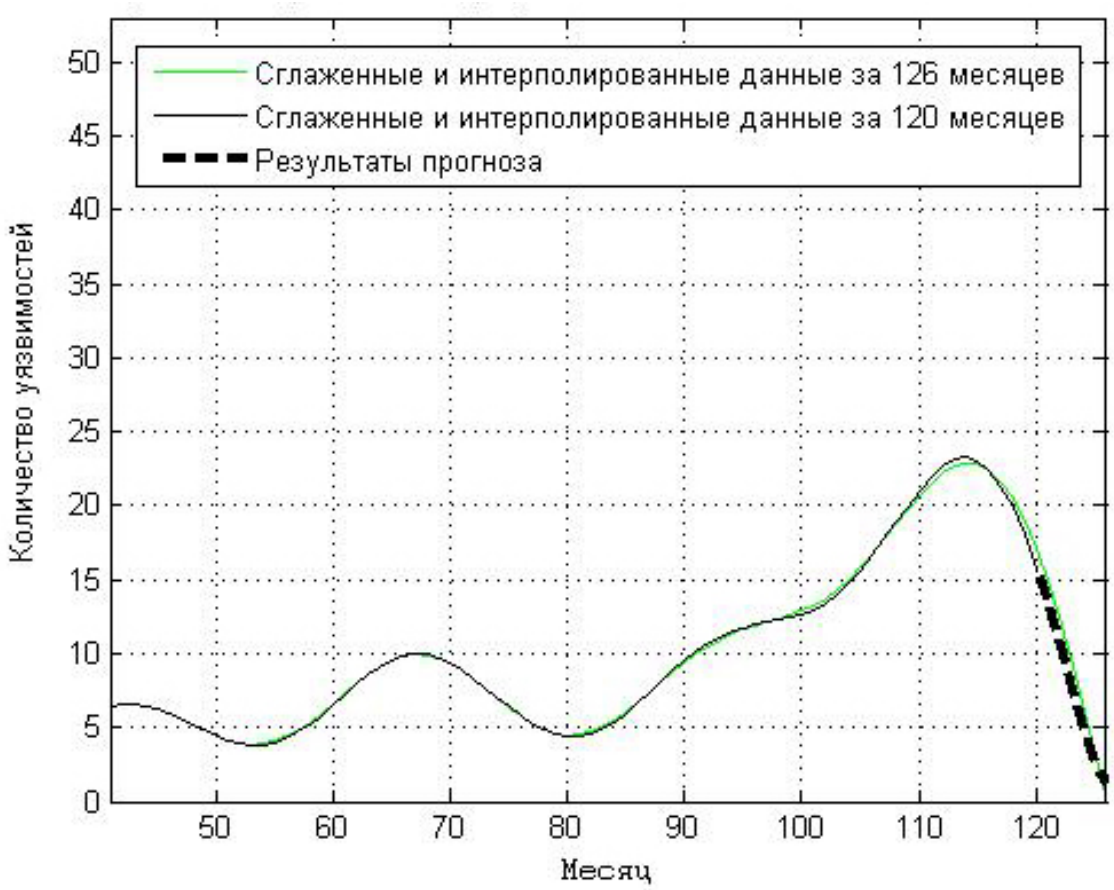

Рисунок 2.3 – Результаты прогноза обнаружения уязвимостей (влияющих на надежность работы ИС) в Windows XP на 6 месяцев при известных данных за 120 месяцев

Предложенный способ прогнозирования обнаружения уязвимостей сравнивался с прогнозом, получаемым при помощи линейной модели Рескорлы и логистической модели Алхазми-Малаийа [6]. Для этого вычислялось среднее абсолютное отклонение прогноза от реальных данных. Результаты сравнения приведены в таблицах 2.1-2.4.

Таблица 2.1 – Среднее абсолютное отклонение полугодового прогноза обнаружения уязвимостей (влияющих на надежность работы ИС) в Windows XP от реальных данных

Известные данные (время жизни ПО), месяцы	Среднее абсолютное отклонение прогноза от реальных данных, уязвимости				
	Линейная модель Рескорлы	Логистическая модель Алхазми-Малаийа	Нейронная сеть (без априорного решения)	Нейронная сеть (априорное решение – линейная модель Рескорлы)	Нейронная сеть (априорное решение – логистическая модель Алхазми-Малаийа)
126	12,15	11,99	12,34	12,58	12,76
120	8,15	7,98	5,94	9,22	3,62
114	15,58	15,33	16,07	15,34	15,27
108	11,18	10,82	12,33	10,88	10,83
102	10,82	10,20	14,00	10,70	10,20
96	8,30	6,77	8,45	7,76	5,26
Среднее значение	11,03	10,52	11,52	11,08	9,66

Таблица 2.2 – Среднее абсолютное отклонение полугодового прогноза обнаружения уязвимостей (влияющих на надежность работы ИС) в Windows Vista от реальных данных

Известные данные (время жизни ПО), месяцы	Среднее абсолютное отклонение прогноза от реальных данных, уязвимости				
	Линейная модель Рескорлы	Логистическая модель Алхазми-Малаийа	Нейронная сеть (без априорного решения)	Нейронная сеть (априорное решение – линейная модель Рескорлы)	Нейронная сеть (априорное решение – логистическая модель Алхазми-Малаийа)
66	9,70	8,14	4,18	2,52	3,07
60	15,17	14,52	2,36	3,34	8,33
54	12,38	12,62	12,76	12,62	12,85
48	14,9	14,06	16,08	16,74	14,44
42	12,29	12,02	15,76	12,11	12,05
36	3,27	3,25	5,07	4,34	3,47
Среднее значение	11,29	10,77	9,37	8,61	9,04

Таблица 2.3 – Среднее абсолютное отклонение полугодового прогноза обнаружения уязвимостей (влияющих на надежность работы ИС) в Windows Server 2003 от реальных данных

Известные данные (время жизни ПО), месяцы	Среднее абсолютное отклонение прогноза от реальных данных, уязвимости				
	Линейная модель Рескорлы	Логистическая модель Алхазми-Малаийа	Нейронная сеть (без априорного решения)	Нейронная сеть (априорное решение – линейная модель Рескорлы)	Нейронная сеть (априорное решение – логистическая модель Алхазми-Малаийа)
102	5,59	5,70	5,24	3,62	2,75
96	11,39	11,31	11,22	10,88	11,46
90	5,32	5,45	5,34	5,13	4,86
84	5,76	5,88	5,44	7,06	7,24
78	3,04	2,84	2,70	2,85	4,83
72	2,60	2,34	2,26	1,79	1,73
Среднее значение	5,62	5,59	5,37	5,22	5,48

Таблица 2.4 – Среднее абсолютное отклонение полугодового прогноза обнаружения уязвимостей (влияющих на надежность работы ИС) в Windows XP, Windows Vista и Windows Server 2003 от реальных данных

Программное обеспечение	Среднее абсолютное отклонение прогноза от реальных данных, уязвимости				
	Линейная модель Рескорлы	Логистическая модель Алхазми-Малаийа	Нейронная сеть (без априорного решения)	Нейронная сеть (априорное решение – линейная модель Рескорлы)	Нейронная сеть (априорное решение – логистическая модель Алхазми-Малаийа)
Windows XP	11,03	10,52	11,52	11,08	9,66
Windows Vista	11,29	10,77	9,37	8,61	9,04
Windows Server 2003	5,62	5,59	5,37	5,22	5,48
Среднее значение	9,31	8,96	8,75	8,30	8,06

Результаты сравнения показывают, что прогноз динамики обнаружения уязвимостей для операционных систем семейства Windows с использованием нейронной сети без учета априорного решения в среднем на 2 % точнее, чем прогноз при помощи линейной модели Рескорлы и логистической модели Алхазми-Малаийа, а прогноз с использованием нейронной сети с учетом априорного решения в среднем на 10% точнее, чем прогноз при помощи этих моделей. При этом стоит отметить, что в отдельных точках (при попадании на осцилляцию) разница в прогнозе гораздо существенней и может составлять порядка 70%. Разница в 10% между нейросетевым алгоритмом и аналитическими моделями прогнозирования обнаружения уязвимостей при средней скорости открытия уязвимостей (приблизительно 8-9 уязвимостей в месяц для Windows XP) в среднем означает неточность в 1 уязвимость в месяц, что в ряде случаев недопустимо (особенно с учетом того, что срок закрытия уязвимости может составлять месяц и более), так как наличие даже одной уязвимости в ИС позволяет нарушить ее работу, а каждая новая уязвимость в ИС предоставляет дополнительные возможности для этого. Следовательно, при прогнозе интенсивности обнаружения уязвимостей предпочтительней использовать нейронную сеть с учетом априорного решения. При этом стоит отметить, что в приведенном выше примере прогноз осуществлялся для уязвимостей с любой степенью серьезности (низкая, средняя, высокая [9]), но такой прогноз может быть произведен и отдельно для каждой степени серьезности.

2.2 Модель динамики обнаружения и устранения уязвимостей программного обеспечения

ИНВ может негативно воздействовать (как преднамеренно, так и непреднамеренно) только на ИС, в ПО которых есть уязвимости, более того, чем больше уязвимостей в ПО ИС, тем больше вероятность того, что негативное воздействие будет успешным, следовательно, для моделирования конфликтных взаимодействий между ИНВ и ИС необходимо оценить динамику изменения

количества уязвимостей в ПО ИС. Необходим не только прогноз в отношении динамики обнаружения уязвимостей, которые можно использовать для нарушения надежности ИС, но и оценка интенсивности устранения этих уязвимостей из ИС. Как было показано в главе 1, на процесс устранения уязвимостей из ИС влияют вендоры и системные администраторы. При этом от вендора зависит, насколько быстро после обнаружения уязвимости будет выпущен патч (временное решение), устраняющий ее, а от системного администратора - насколько быстро этот патч (временное решение) будет установлен или насколько быстро сам системный администратор без помощи вендора разработает свое временное решение, устраняющее уязвимость. Поэтому среднюю скорость устранения уязвимостей из ИС предлагается описывать следующим образом:

$$\mu = k\mu_в, \tag{2.6}$$

где $\mu_в$ - средняя скорость создания вендором патча, устраняющего уязвимость в данной программе, а k - коэффициент, характеризующий работу системного администратора (для данной программы).

Оценку $\mu_в$ предлагается производить следующим образом:

$$\mu_в = \frac{1}{T_в}, \tag{2.7}$$

где $T_в$ – среднее время создания вендором патча, закрывающего уязвимость, после ее обнаружения. Для каждой программы оценка $\mu_в$ должна производиться отдельно, так как даже один и тот же вендор зачастую создает патчи для разных программ с разной скоростью.

Коэффициент k предлагается оценивать экспертным путем:

- $k = 0$, если системный администратор вообще не устанавливает патчи, выпущенные вендором;
- $k < 1$, если системный администратор не своевременно устанавливает патчи, выпущенные вендором;

- $k = 1$, если системный администратор устанавливает патчи, выпущенные вендором, сразу же после их выпуска;

- $k > 1$, если системный администратор устанавливает патчи, выпущенные вендором, сразу же после их выпуска и при этом сам предлагает временные решения для устранения некоторых уязвимостей.

Следует отметить, что можно считать $k = 1$ в случае, когда настроено автоматическое обновление ПО, в случае же, если системный администратор обновляет ПО самостоятельно, преимущественно можно считать что $k < 1$.

Для каждой программы оценка k должна производиться отдельно, так как системные администраторы зачастую по-разному реализуют политику обновления для разных программ.

Для оценки числа уязвимостей ПО ранее предложена модель [29] (далее модель Щеглова А.Ю.), но она имеет существенный недостаток, а именно не учитывает изменение интенсивности обнаружения уязвимостей в ПО во времени, т.е. нестационарный характер процесса обнаружения уязвимостей. Следовательно, необходимо разработать модель (далее разработанная модель), устраняющую данное ограничение.

Аналогично [29] предлагается представить процесс появления новых уязвимостей и их устранения как процесс функционирования системы массового обслуживания (СМО) [76]. Но в отличие от [29] предполагается, что на вход СМО поступает нестационарный пуассоновский поток заявок (уязвимостей) с интенсивностью $\lambda(t)$, зависящей от времени t (алгоритмы оценки и прогноза которой описаны в параграфе 2.1). Поток уязвимостей является нестационарным пуассоновским, так как фактически он представляет собой сумму порядка 100÷1000 независимых нестационарных потоков с приблизительно одинаковой интенсивностью [77], порождаемых ИНВ и специалистами в области компьютерной безопасности, занимающимися поиском новых уязвимостей [78]. Далее СМО обслуживает эти заявки (устраняет уязвимости) с интенсивностью μ, рассчитываемой по формуле (2.6). Так же, как в [29], предполагается, что работа над устранением каждой уязвимости начинается сразу же после ее обнаружения,

соответственно, данная СМО имеет бесконечное число каналов обслуживания. При данных предположениях среднестатистическое число уязвимостей в программе на данный момент времени t рассчитывается по формуле [76]

$$N_{cp}(t) = \frac{e^{-t}}{\mu}\left(\lambda(t) + \int_0^t \lambda(\tau)e^{\tau}d\tau\right). \quad (2.8)$$

С учетом (2.6) и (2.7), формула (2.8) примет вид

$$N_{cp}(t) = \frac{T_в e^{-t}}{k}\left(\lambda(t) + \int_0^t \lambda(\tau)e^{\tau}d\tau\right). \quad (2.9)$$

где $\lambda(t)$ - интенсивность обнаружения уязвимостей, $T_в$ - среднее время создания вендором патча, закрывающего уязвимость, после ее обнаружения, а k - коэффициент работы системного администратора. В граничном случае, если коэффициент работы системного администратора $k = 0$, уязвимости, найденные в программе, не закрываются, а только накапливаются. Следовательно, в этом случае среднестатистическое число уязвимостей в системе равно среднестатистическому числу уязвимостей, обнаруженных в программе за время ее жизни

$$N_{cp}(t) = \int_0^t \lambda(\tau)d\tau \quad (2.10)$$

При данных предположениях вероятность того, что в конкретной программе находится n уязвимостей, равна [76]:

$$P_n(t) = \frac{\left[N_{cp}(t)\right]^n}{n!}e^{-N_{cp}(t)}, \quad (2.11)$$

Таким образом, вероятность отсутствия в программе уязвимостей равна

$$P_0(t) = e^{-N_{cp}(t)}. \quad (2.12)$$

Далее предлагается оценка среднестатистического числа уязвимостей и вероятности отсутствия уязвимостей в операционных системах семейства Windows: Windows XP, Windows Vista и Windows Server 2003 за 132, 72 и 108

месяцев соответственно для случаев, когда коэффициент работы системного администратора $k = 0{,}5$; $k = 1$; $k = 1{,}5$ и $k = 3$ (граничный случай, когда в качестве системного администратора выступает целая служба безопасности, в которую входит большое число высококвалифицированных сотрудников, и применяются серьёзные организационные и технические меры по устранению уязвимостей из ПО), при помощи разработанной модели и модели Щеглова А.Ю [29]. Данные для расчета (для определения значений интенсивности обнаружения уязвимостей в ПО $\lambda(t)$ и среднего времени выпуска вендором патча, закрывающего уязвимость $T_в$) брались из [51,77,79]. Ниже приведены графики для операционной системы Windows XP (рис 2.4) .

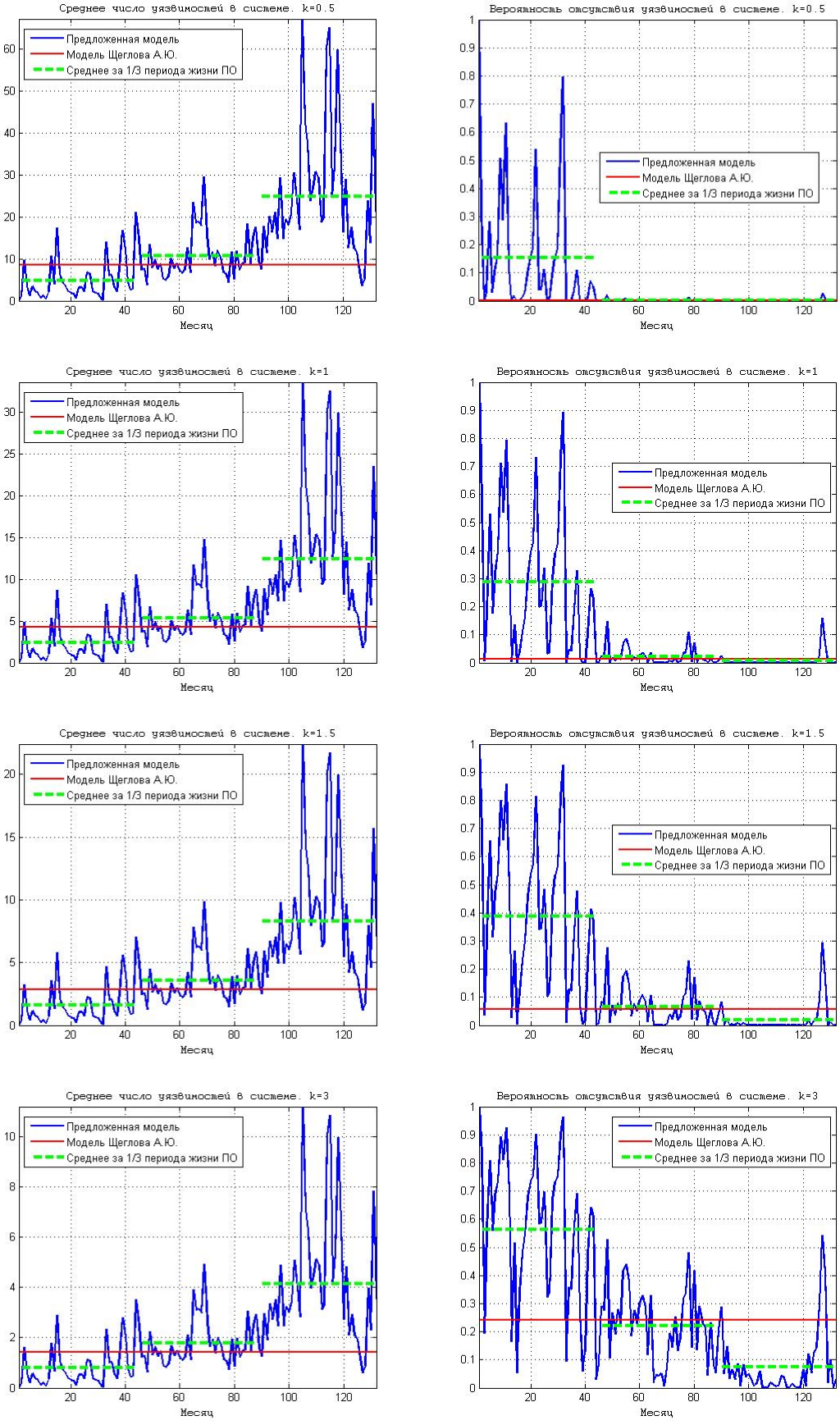

Рисунок 2.4 – Среднестатистическое число уязвимостей и вероятность отсутствия уязвимостей в Windows XP

Значения среднестатистического числа уязвимостей, рассчитанные с использованием модели Щеглова А.Ю. [29], сравнивались со средними значениями среднестатистического числа уязвимостей за каждую 1/3 периода существования ПО, рассчитанными с использованием разработанной модели (Таблицы 2.5-2.10).

Таблица 2.5 – Среднестатистическое число уязвимостей в операционной системе Windows XP.

Коэффициент k	Разработанная модель			Модель Щеглова А. Ю.
	Среднее за период с 1 по 44 месяц	Среднее за период с 45 по 88 месяц	Среднее за период с 89 по 132 месяц	
0,5	4,77	10,76	24,87	8,57
1	2,38	5,38	12,43	4,28
1,5	1,59	3,59	8,29	2,86
3	0,79	1,79	4,14	1,43

Таблица 2.6 – Среднестатистическое число уязвимостей в операционной системе Windows Vista.

Коэффициент k	Разработанная модель			Модель Щеглова А. Ю.
	Среднее за период с 1 по 44 месяц	Среднее за период с 45 по 88 месяц	Среднее за период с 89 по 132 месяц	
0,5	7,48	19,59	22,83	10,53
1	3,74	9,8	11,42	5,26
1,5	2,49	6,53	7,61	3,51
3	1,25	3,27	3,81	1,75

Таблица 2.7 – Среднестатистическое число уязвимостей в операционной системе Windows Server 2003.

Коэффициент k	Разработанная модель			Модель Щеглова А. Ю.
	Среднее за период с 1 по 44 месяц	Среднее за период с 45 по 88 месяц	Среднее за период с 89 по 132 месяц	
0,5	4,95	6,39	11,94	4,91
1	2,47	3,19	5,97	2,45
1,5	1,65	2,13	3,98	1,64
3	0,82	1,06	1,99	0,82

Таблица 2.8 – Вероятность отсутствия уязвимостей в операционной системе Windows XP.

Коэффициент k	Разработанная модель			Модель Щеглова А. Ю.
	Среднее за период с 1 по 44 месяц	Среднее за период с 45 по 88 месяц	Среднее за период с 89 по 132 месяц	
0,5	0,153	0,001	0,001	0
1	0,288	0,022	0,007	0,014
1,5	0,387	0,065	0,018	0,057
3	0,562	0,22	0,074	0,24

Таблица 2.9 – Вероятность отсутствия уязвимостей в операционной системе Windows Vista.

Коэффициент k	Разработанная модель			Модель Щеглова А. Ю.
	Среднее за период с 1 по 44 месяц	Среднее за период с 45 по 88 месяц	Среднее за период с 89 по 132 месяц	
0,5	0,086	0	0,002	0
1	0,132	0,004	0,015	0,005
1,5	0,191	0,018	0,035	0,03
3	0,361	0,097	0,116	0,173

Таблица 2.10 – Вероятность отсутствия уязвимостей в операционной системе Windows Server 2003.

Коэффициент k	Разработанная модель			Модель Щеглова А. Ю.
	Среднее за период с 1 по 44 месяц	Среднее за период с 45 по 88 месяц	Среднее за период с 89 по 132 месяц	
0,5	0,194	0,073	0,03	0,007
1	0,291	0,152	0,075	0,086
1,5	0,371	0,235	0,122	0,195
3	0,536	0,425	0,255	0,441

Из таблиц 2.5-2.10 (и рисунка 2.4) видно, что разработанная модель, в отличие от модели Щеглова А.Ю. [29], позволяет учесть наличие на практике [51,77] периодов увеличения и уменьшения числа уязвимостей в ПО (увеличения и уменьшения вероятности отсутствия уязвимостей в ПО), а, следовательно, более приемлема для дальнейшего использования при моделировании динамики уязвимостей в ПО.

2.3 Математические модели функционирования информационной системы

Поскольку в ИС может быть установлено несколько разных программ, то простейшая математическая модель функционирования ИС в условиях внутренних уязвимостей и конфликтных взаимодействий может быть представлена как совокупность систем массового обслуживания, каждая из которых моделирует динамику уязвимостей в каждой отдельной программе. Данная модель отображена на рисунке 2.5, здесь $\lambda^{(m)}(t)$ - скорость обнаружения уязвимостей в m-й программе, $k^{(m)}$- коэффициент, характеризующий обслуживание системным администратором m-й программы, $T_в^{(m)}$- среднее время

создания вендором патча, закрывающего уязвимость, после ее обнаружения в m-й программе, а M - общее число программ.

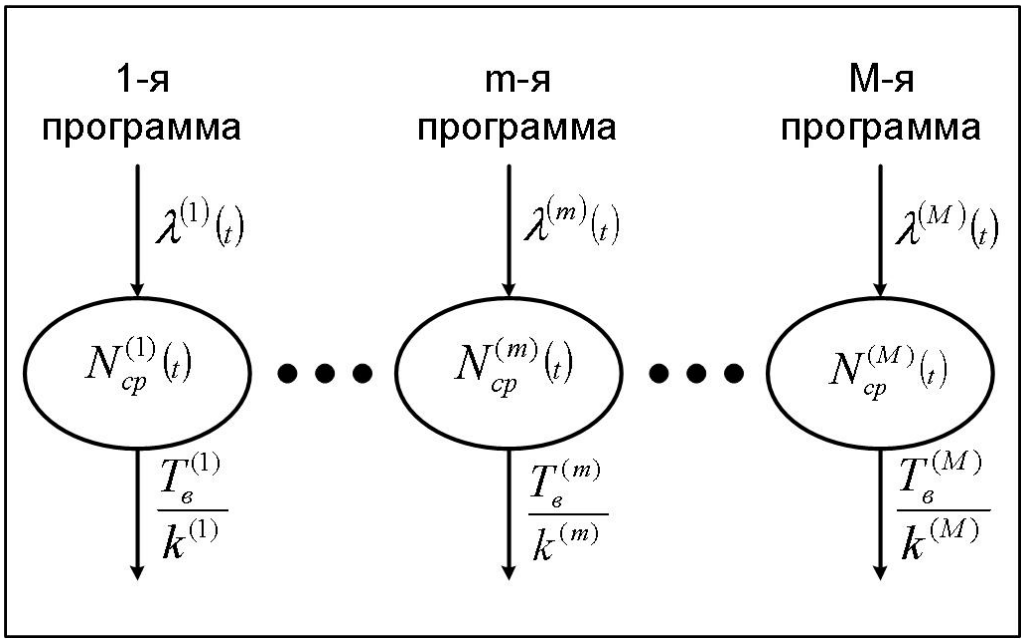

Рисунок 2.5 – Простейшая математическая модель функционирования ИС (без СЗИ) в условиях внутренних уязвимостей и конфликтных взаимодействий

В этом случае среднестатистическое число уязвимостей в ИС будет суммой среднестатистического числа уязвимостей в каждой программе, установленной в ИС

$$N_{cp}(t)=\sum_{m=1}^{M}N_{cp}^{(m)}\ ,\ N_{cp}^{(m)}(t)=\frac{T_в^{(m)}e^{-t}}{k^{(m)}}\left(\lambda^{(m)}(t)+\int_0^t\lambda^{(m)}(\tau)e^{\tau}d\tau\right), \qquad (2.13)$$

Вероятность отсутствия в ИС уязвимостей может быть рассчитана по формуле (2.12), если вместо среднего числа уязвимостей в конкретной программе (2.9) в нее подставить среднее число уязвимостей в ИС (2.13).

В простейшем случае, когда ИНВ имеет доступ ко всем программам, установленным в ИС, и уязвимости каждой программы могут быть использованы непосредственно для негативного воздействия на ИС, потенциальная вероятность того, что надежность данной ИС в момент времени t не может быть нарушена ИНВ (далее вероятность надежности ИС), совпадает с вероятностью отсутствия в

ИС уязвимостей (в данном случае не имеет значения, является ли ИНВ внешним или внутренним),

$$P_{над}(t) = P_0(t). \qquad (2.14)$$

Данная вероятность имеет потенциальный характер, так как при ее расчете не учитываются характеристики ИНВ, которые могут негативно воздействовать на ИС, а рассматривается только потенциальная возможность такого воздействия. Стоит отметить, что данная вероятность определяет не только возможность преднамеренного негативного воздействия, но и возможность непреднамеренного негативного воздействия, например, со стороны вредоносного ПО.

Возможны другие варианты организации ИС, когда в ней установлено специальное ПО, СЗИ, защищающее ее от непосредственного негативного воздействия внешнего источника, как преднамеренного, так и непреднамеренного. В этом случае внешний ИНВ, перед тем как непосредственно негативно воздействовать на ИС, должен сначала преодолеть СЗИ, негативно воздействовав на него, используя внутренние уязвимости СЗИ (внутренний ИНВ в этом случае, как и в предыдущем, может сразу непосредственно негативно воздействовать на ИС, используя уязвимости в ее ПО). Тогда надежность ИС не может быть нарушена внешним ИНВ в 2-х случаях:

- в СЗИ нет уязвимостей;
- в СЗИ есть уязвимости, но их нет в остальном ПО, установленном в ИС.

Таким образом, вероятность надежности данной ИС для данного момента времени по отношению к внешнему ИНВ будет рассчитываться по следующей формуле:

$$P_{над}(t) = P_0^{(СЗИ)}(t) + P_0^{(ПО)}(t)(1 - P_0^{(СЗИ)}(t)), \qquad (2.15)$$

где $P_0^{(СЗИ)}$ - вероятность отсутствия уязвимостей в СЗИ, а $P_0^{(ПО)}$ - вероятность отсутствия уязвимостей в остальном ПО, установленном в ИС. Математическая модель функционирования ИС в данном случае представлена на рисунке 2.6.

Здесь $\lambda^{(СЗИ)}(t)$ - скорость обнаружения уязвимостей в СЗИ, $k^{(СЗИ)}$ - коэффициент, характеризующий обслуживание системным администратором СЗИ, $T_в^{(СЗИ)}$ - среднее время создания вендором патча, закрывающего уязвимость, после ее обнаружения в СЗИ, $\lambda^{(m)}(t)$ - скорость обнаружения уязвимостей в m-й программе, $k^{(m)}$ - коэффициент, характеризующий обслуживание системным администратором m-й программы, $T_в^{(m)}$ - среднее время создания вендором патча, закрывающего уязвимость, после ее обнаружения в m-й программе, M - общее число программ.

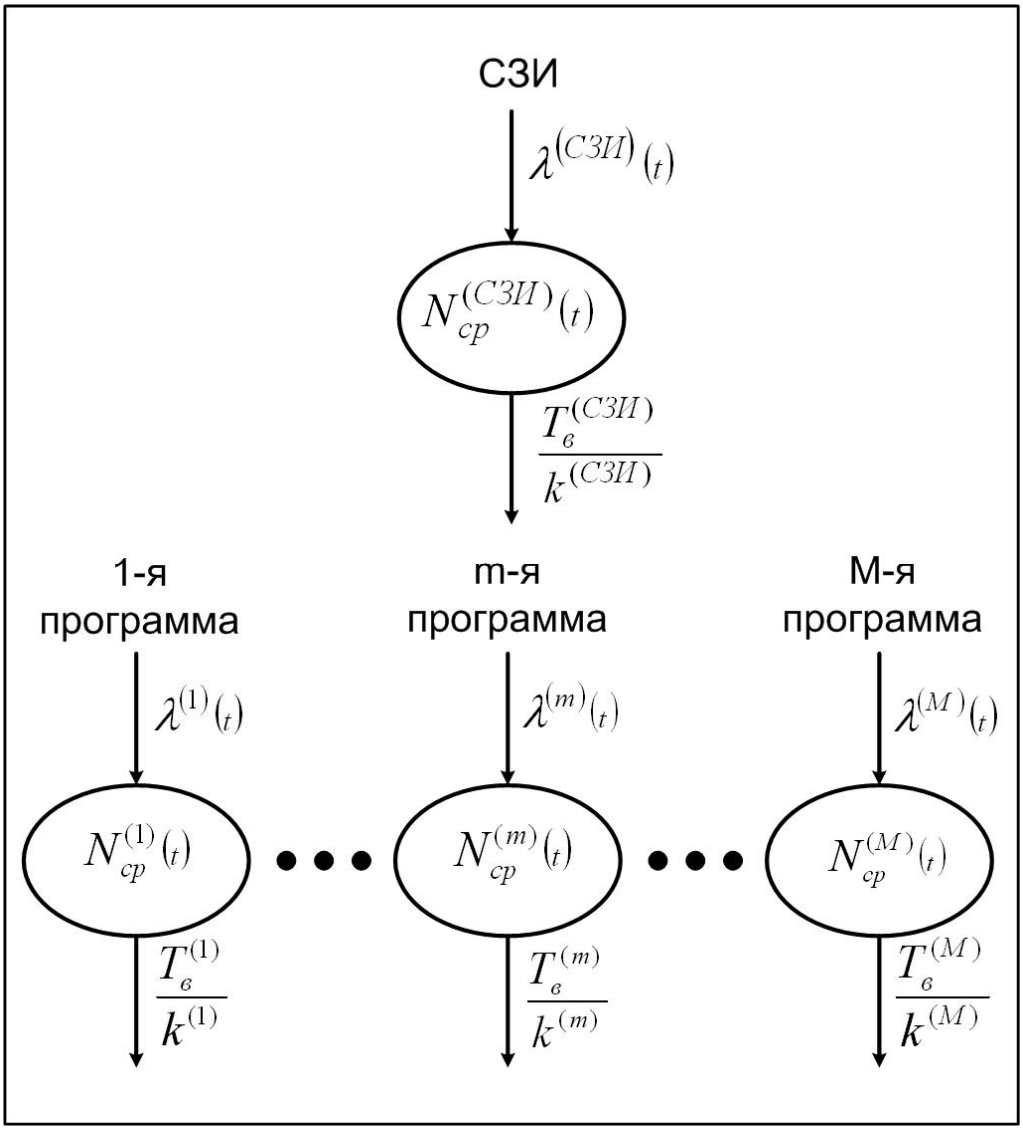

Рисунок 2.6 – Математическая модель функционирования ИС, имеющей СЗИ, в условиях внутренних уязвимостей и конфликтных взаимодействий

Возможны и другие случаи, например, когда в ИС установлено несколько видов СЗИ, которые внешний ИНВ должен преодолеть последовательно, или же, когда успешное негативное воздействие на СЗИ будет уже являться нарушением надежности ИС, или когда в ИС есть СЗИ, защищающие ее от негативного воздействия не только внешнего, но и внутреннего ИНВ, а также многие другие (в том числе и случаи, когда в течение периода оценки надежности ИС некоторое ПО деинсталлируется, а некоторое, наоборот, инсталлируется в ИС). Следовательно, математическая модель функционирования ИС должна разрабатываться в каждом случае отдельно.

Далее предлагается сравнение 2-х ИС. Для простоты предполагается, что в первой ИС установлена только операционная система Windows XP, а во второй помимо этой же операционной системы установлено СЗИ (сетевой экран семейства Cisco IOS 12.x). Статистика для ПО (для определения значений интенсивности обнаружения уязвимостей в ПО $\lambda(t)$ и среднего времени выпуска вендором патча, закрывающего уязвимость, $T_в$) бралась из [51,77,79]. Для сравнения вычислялась вероятность надежности, рассчитанная по формуле (2.14) для ИС без СЗИ и рассчитанная по формуле (2.15) для ИС с СЗИ для случаев, когда коэффициент работы администратора для операционной системы и СЗИ одинаков и равен соответственно $k = 0{,}5$; $k = 1$; $k = 1{,}5$ и $k = 3$ (Таблица 2.11). Ниже приведены графики сравнения ИС (рисунок 2.7).

Таблица 2.11 – Средняя вероятность надежности ИС (за 12 лет) с операционной системой Windows XP без СЗИ и с СЗИ типа сетевой экран Cisco IOS 12.x.

ИС	Коэффициент работы системного администратора k			
	0.5	1	1.5	3
Без СЗИ	0,052	0,106	0,157	0,285
С СЗИ	0,807	0,887	0,922	0,964

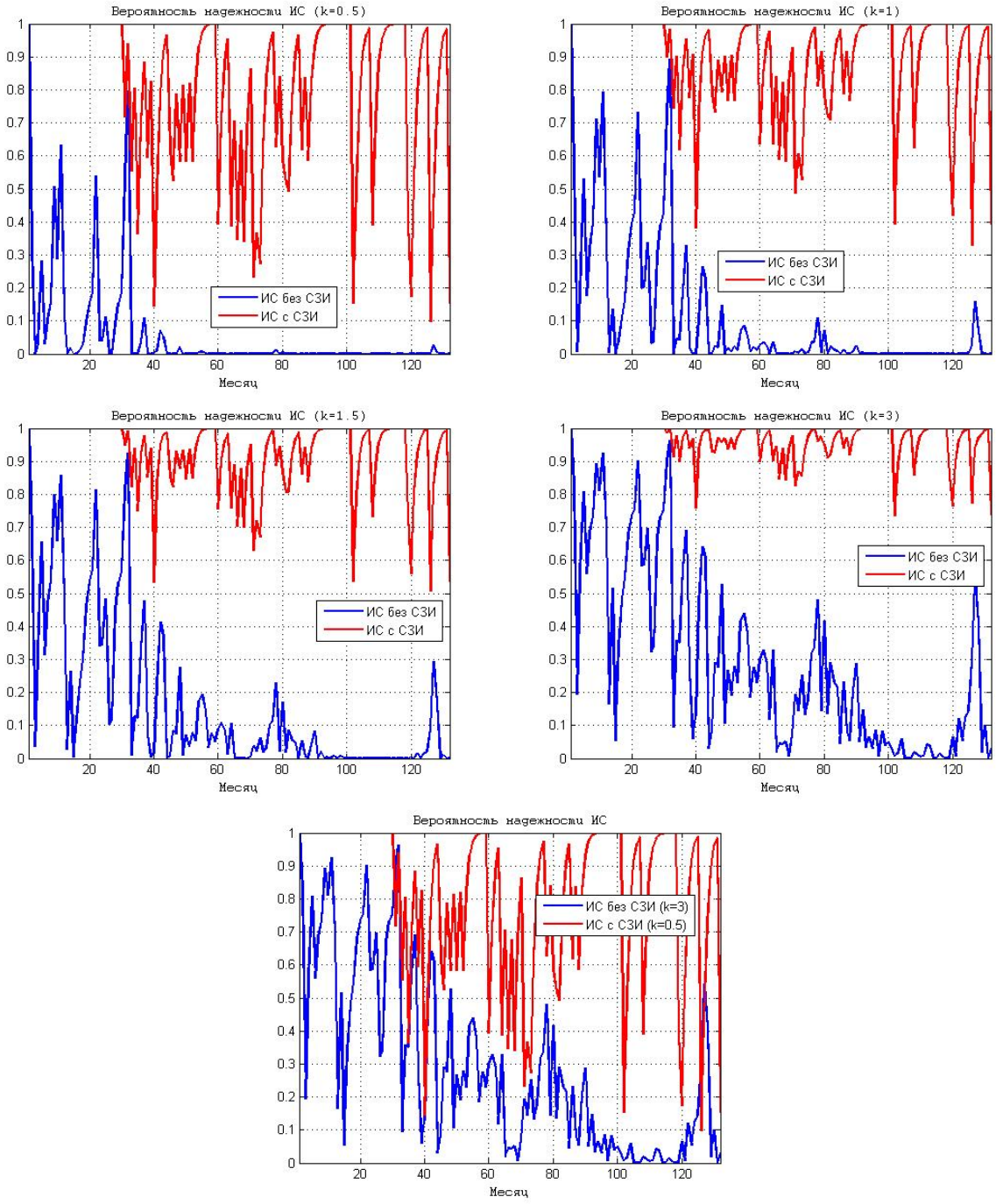

Рисунок 2.7 – Вероятность надежности ИС с операционной системой Windows XP без СЗИ и с СЗИ типа сетевой экран Cisco IOS 12.x

Вероятность надежности ИС с СЗИ даже в худшем случае, когда коэффициент работы системного администратора $k = 0{,}5$, приблизительно в 3 раза выше, чем у ИС без СЗИ в наилучшем случае, когда коэффициент работы системного администратора $k = 3$. В худшем же случае для обеих ИС ($k = 0{,}5$) вероятность надежности ИС с СЗИ приблизительно в 16 раз выше, чем без СЗИ.

Данные результаты показывают, что отсутствие учета наличия СЗИ при моделировании ИС может серьезно повлиять на оценку ее надежности, следовательно, при моделировании ИС учет фактора наличия СЗИ обязателен.

2.4 Общий алгоритм анализа вероятностных характеристик надежности использования программного обеспечения в информационной системе без учета характера негативных воздействий

Совокупность предложенных моделей и алгоритмов оценки интенсивности обнаружения уязвимостей в ПО, оценки интенсивности закрытия уязвимостей в ПО, математических моделей динамики уязвимостей в ПО и моделей функционирования ИС позволяет осуществить анализ вероятностных характеристик надежности использования программного обеспечения в ИС в условиях внутренних уязвимостей и НВ. При этом характер НВ (преднамеренное иди непреднамеренное) не учитывается. Для реализации этой задачи предлагается следующий алгоритм, общая схема которого приведена на рис. 2.8.

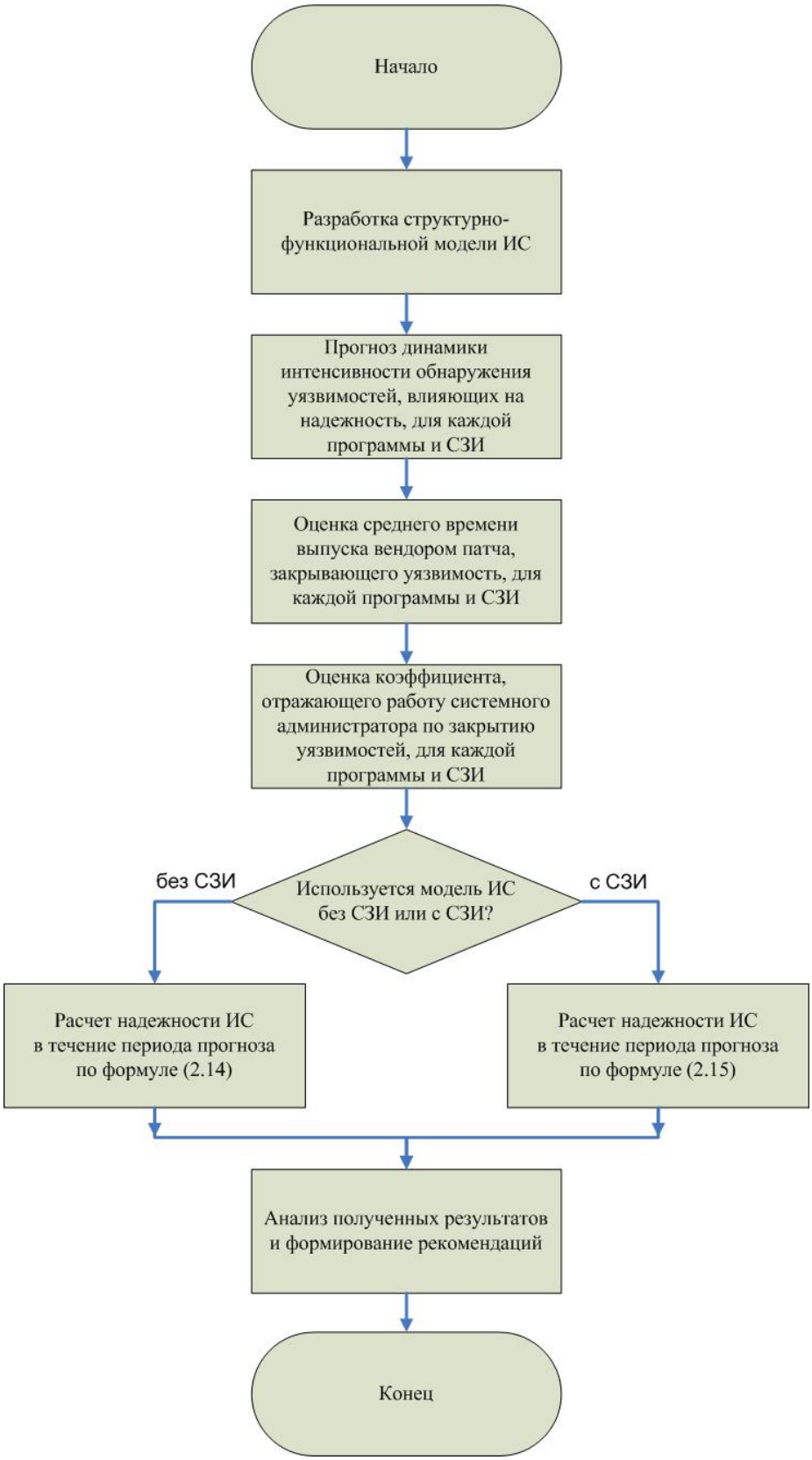

Рисунок 2.8 – Блок-схема общего алгоритма анализа вероятностных характеристик надежности использования ПО в ИС в условиях внутренних уязвимостей и НВ без учета характера НВ

При реализации алгоритма выполняются следующие этапы анализа системы:

1. Разработка структурно-функциональной модели ИС: определение перечня предустановленного ПО, наличия СЗИ и их конфигурации (защищают от внешнего и/или внутреннего ИНВ).

2. Прогноз динамики интенсивности обнаружения уязвимостей, влияющих на надежность, для каждой программы и СЗИ.

3. Оценка среднего времени выпуска вендором патча, закрывающего уязвимость, для каждой программы и СЗИ.

4. Оценка коэффициента, отражающего работу системного администратора по закрытию уязвимостей, для каждой программы и СЗИ.

5. Расчет вероятности надежности ИС без учета характера НВ в течение периода прогноза исходя из разработанной модели ИС.

6. Анализ полученных результатов и формирование рекомендаций.

Таким образом, полученный в рамках предложенного подхода общий алгоритм, в отличие от других подходов к оценке надежности ИС [29-31], позволяет одновременно использовать прогноз в отношении интенсивности обнаружения уязвимостей в ПО, учесть работу вендоров и системных администраторов и состав ИС (наличие различного ПО и СЗИ). Кроме того, в отличие от других динамических подходов [29,31], данный алгоритм при оценке и прогнозе ряда параметров использует общедоступную статистику, опубликованную в сети Интернет (например [51,77,79]), сбор которой может быть автоматизирован.

Выводы по главе

1. Разработан двухэтапный нейросетевой алгоритм статистического анализа и прогнозирования нестационарных временных последовательностей, основанный на применении на первом этапе специальной процедуры интерполяции экспериментальных данных в виде разложения по радиально-базисным функциям с нахождением коэффициентов разложения с использованием метода регуляризации, а также на втором этапе – процедуры прогнозирования на основе нейронной сети в виде многослойного персептрона, обученной по интерполированным данным. Указанный алгоритм использован для прогноза интенсивности уязвимостей программного обеспечения, что позволило обеспечить повышение точности прогноза до 70% по сравнению с известными [10].

2. Разработана модель динамики уязвимостей программного обеспечения, позволяющая, в отличие от модели Щеглова А. Ю. [29], учитывать зависимость интенсивности обнаружения уязвимостей в ПО от времени и зависимость интенсивности закрытия уязвимостей в ПО от работы администратора.

3. Разработаны структурно-функциональные модели ИС, учитывающие, в отличии от модели Щеглова А. Ю. [29], зависимость интенсивности обнаружения уязвимостей в ПО от времени, зависимость интенсивности закрытия уязвимостей в ПО от работы системного администратора и различное устройство ИС (наличие различного ПО, наличие СЗИ и их конфигурации, и т.п.).

4. Разработан общий алгоритм анализа вероятностных характеристик надежности использования ПО в ИС в условиях внутренних уязвимостей и НВ без учета характера НВ, позволяющий, в отличие от известных [29-31], использовать прогноз в отношении интенсивности обнаружения уязвимостей в ПО, учесть работу вендоров и системных администраторов и состав ИС (наличие различного ПО и СЗИ). Кроме того, в отличие от других динамических подходов [30,31], данный алгоритм при оценке и прогнозе ряда параметров использует

общедоступную статистику, опубликованную в сети Интернет (например [51,77,79]), сбор которой может быть автоматизирован.

Глава 3. Компьютерное моделирование информационных процессов и систем при наличии внутренних уязвимостей и конфликтных взаимодействий

Для оценки надежности ИС требуется не только рассмотрение всех ее существенных параметров и характеристик в динамике функционирования, но и оценка возможностей противоборствующей стороны – ИНВ, осуществляющих НВ на ИС. При этом в качестве универсальной синтетической методологии исследований в сфере надежности информационных систем и технологий целесообразно рассматривать методологию математического и компьютерного моделирования динамики конфликта, опирающуюся на концептуальные модели конфликтных взаимодействий [41]. Предлагается рассмотреть 4 типовые ситуации развертывания такого конфликта:

1. Информационная система без СЗИ находится в конфликтном взаимодействии с одним внешним ИНВ.
2. Информационная система с СЗИ находится в конфликтном взаимодействии с одним внешним ИНВ.
3. Информационная система без СЗИ находится в конфликтном взаимодействии с коалицией ИНВ в отсутствии инсайдера (человека, имеющего прямой доступ к ПО ИС и поставляющего информацию об этом ПО и уязвимостях в нем ИНВ), работающего в данной организации.
4. Информационная система без СЗИ находится в конфликтном взаимодействии с коалицией ИНВ при наличии инсайдера, работающего в организации, на ИС которой планируется НВ.

Данные ситуации не исчерпывают всех возможных вариантов конфликта ИС – ИНВ; тем не менее, на их основе, исходя из тех же принципов, могут быть рассмотрены и другие варианты. Кроме того, стоит отметить, что при расчетах, иллюстрирующих возможности разработанных моделей, учитывались уязвимости с любой степенью серьезности (нижняя, средняя, высокая [9]), при использовании же этих моделей на практике следует либо производить отдельный анализ по

уязвимостям разной степени серьезности (при этом уязвимости средней и высокой степени серьезности вполне возможно объединить в одну группу), либо присваивать уязвимостям разной степени серьезности коэффициенты, отражающие их влияние на надежность ИС.

При моделировании информационных процессов и систем в указанной постановке будут использоваться три типа моделей: объектно-ориентированные модели в нотациях UML [80], математические модели, основанные на использовании тех или иных вероятностных описаний динамики конфликта, и компьютерные модели, реализованные в интегрированной среде Matlab+Simulink+Stateflow, обеспечивающей адекватный учет исходных концептуальных и функциональных объектных представлений.

3.1. Модели функционирования информационной системы без средств защиты информации в условиях конфликтного взаимодействия с одним внешним источником негативных воздействий

Пусть имеется ИС с установленным ПО. ИНВ, преднамеренно воздействующий на ИС, как было показано в главе 1, как правило, проводит предварительный анализ (компьютерную разведку) программного обеспечения, установленного на ИС, исследует уязвимости в этом программном обеспечении и возможные способы использования этих уязвимостей [4,5], при этом ИНВ может обладать различной квалификацией и возможностями. Квалификацию и возможности ИНВ будем описывать средним временем, которое требуется ИНВ [81-85]: для получения информации о ПО ИС; для получения информации обо всех уязвимостях в ПО ИС; для получения информации об использовании уязвимости в ПО ИС для организации НВ на ИС.

При этом динамика уязвимостей ПО ИС описывается простейшей математической моделью ИС, рассмотренной в п. 2.3.

Объектно-ориентированная модель конфликтного взаимодействия. Для конструирования объектно-ориентированной модели конфликта предлагается

применить аппарат языка UML [80], используя для описания поведения сторон, участвующих в конфликтном взаимодействии, диаграмму состояний. Ниже на рисунках 3.1 – 3.3 приведены диаграммы состояний, описывающие поведение ИС, системного администратора и ИНВ соответственно.

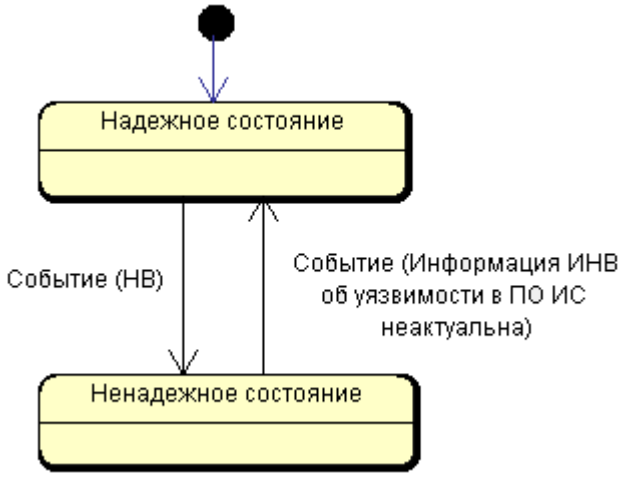

Рисунок 3.1 – Диаграмма состояний ИС без СЗИ в ходе конфликтного взаимодействия с ИНВ

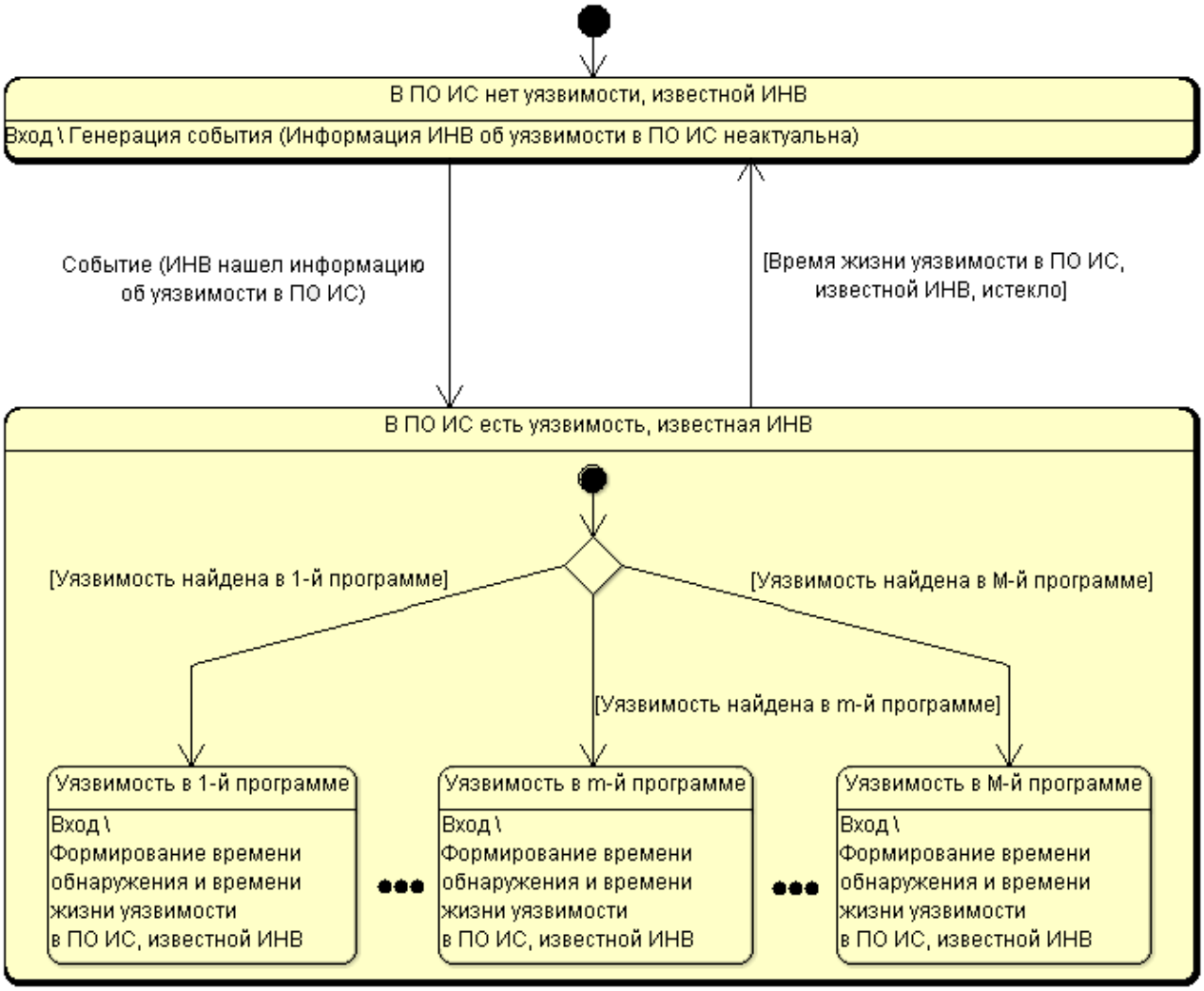

Рисунок 3.2 – Диаграмма состояний системного администратора в ходе конфликтного взаимодействия ИС без СЗИ с ИНВ

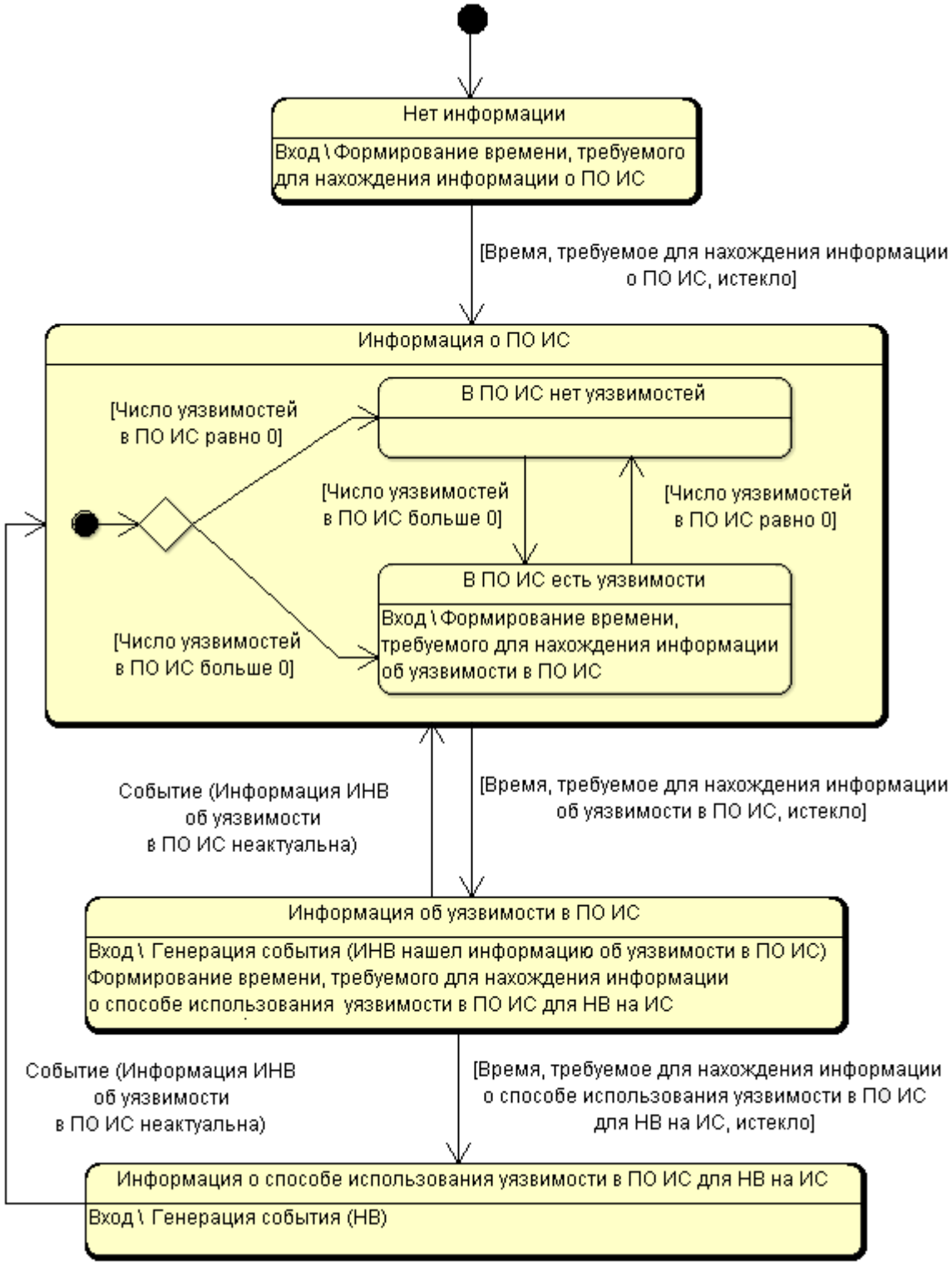

Рисунок 3.3 – Диаграмма состояний ИНВ в ходе конфликтного взаимодействия с ИС без СЗИ

Как показано на рисунке 3.1 информационная система может находиться в 2-х основных состояниях:

1. Состояние *«Надежное состояние»* – состояние, при котором ИНВ не может совершить успешное воздействие на ИС.

2. Состояние *«Ненадежное состояние»* – состояние, при котором ИНВ может совершить успешное воздействие на ИС.

Предполагается, что ИС первоначально находится в *«надежном состоянии»*. Переход из *«надежного состояния»* в *«ненадежное»* осуществляется при событии *«НВ»*. Обратный переход осуществляется при возникновении события *«Информация ИНВ об уязвимости в ПО ИС неактуальна»*.

Работа системного администратора представляется 2-мя состояниями, в которых может находиться уязвимость, известная ИНВ (рис. 3.2):

1. Состояние *«В ПО ИС нет уязвимости, известной ИНВ»* – состояние, при котором ИНВ не владеет информацией об уязвимости в ПО ИС;

2. Состояние *«В ПО ИС есть уязвимость, известная ИНВ»* – состояние, при котором у ИНВ есть информация об уязвимости в ПО ИС.

Предполагается, что первоначально системный администратор находится в первом состоянии (*«В ПО ИС нет уязвимости, известной ИНВ»*). При входе в это состояние генерируется событие *«Информация ИНВ об уязвимости в ПО ИС неактуальна»*. Переход во второе состояние (*«В ПО ИС есть уязвимость, известная ИНВ»*) осуществляется при возникновении событии *«ИНВ нашел информацию об уязвимости в ПО ИС»*. Во втором состоянии происходит выбор одного из подсостояний: *«Уязвимость в 1-й программе», …, «Уязвимость в 2-й программе», …, «Уязвимость в M-й программе»* (всего в ИС установлено M программ, $m \in \overline{1,M}$) в зависимости от того, в какой программе была обнаружена уязвимость, в выбранном подсостоянии формируется время обнаружения уязвимости, известной ИНВ, и время ее жизни (время до создания патча или временного решения, закрывающего ее). Обратный переход из состояния *«В ПО ИС есть уязвимость, известная ИНВ»*, в состояние *«В ПО ИС нет уязвимости, известной ИНВ»*, происходит по условию истечения времени жизни уязвимости.

Соответственно (рис.3.3), ИНВ может находится в 4-х состояниях:

1. Состояние *«Нет информации об ИС»* – начальное состояние, при котором у ИНВ отсутствует какая-либо информация об ИС;
2. Состояние *«Информация о ПО ИС»* – состояние, при котором у ИНВ есть информация о ПО ИС;
3. Состояние *«Информация об уязвимости в ПО ИС»* – состояние, при котором у ИНВ есть информация о ПО ИС и об одной уязвимости в этом ПО;
4. Состояние *«Информация о способе использования уязвимости в ПО ИС для НВ на ИС»* – состояние, при котором у ИНВ есть информация о ПО ИС, об одной уязвимости в этом ПО, а также информация о способе использования этой уязвимости для осуществления НВ на ИС.

Предполагается, что ИНВ первоначально находится в состоянии *«Нет информации об ИС»*. В данном состоянии формируется время, требуемое ИНВ для нахождения информации о ПО ИС. Переход в состояние *«Информация о ПО ИС»* происходит по условию истечения этого времени. Состояние *«Информация о ПО ИС»* содержит 2 подсостояния, *«В ПО ИС нет уязвимостей»* и *«В ПО ИС есть уязвимости»*, в одно из которых ИНВ попадает в зависимости от числа уязвимостей в ИС. Если оно больше 0 - то в состояния *«В ПО ИС есть уязвимости»*, если равно 0 – то в состояние *«В ПО ИС нет уязвимостей»*. Исходя из этого же условия, осуществляются переходы между этими подсостояниями. В подсостоянии *«В ПО ИС есть уязвимости»* формируется время, требуемое для нахождения одной уязвимости в ПО ИС. Переход в состояние *«Информация об уязвимости в ПО ИС»* происходит по истечению этого времени. При попадании в данное состояние генерируется событие *«ИНВ нашел информацию об уязвимости в ПО ИС»* и формируется время, требующееся для нахождения информации о способе использования уязвимости в ПО ИС для НВ на ИС. Переход в состояние *«Информация о способе использования уязвимости в ПО ИС для НВ на ИС»* происходит по условию истечения этого времени. При попадании в данное состояние генерируется событие *«НВ»* (переводящее ИС в состояние *«Ненадежное состояние»*). Также существуют переходы из состояний *«Информация об уязвимости»* и *«Информация о способе использования*

уязвимости для НВ на ИС» в состояние *«Информация о ПО ИС»* в случае возникновения события *«Информация ИНВ об уязвимости в ПО ИС неактуальна»*.

Таким образом, совокупность 3-х диаграмм состояний (рис 3.1-3.3) описывает конфликт между ИС без СЗИ и ИНВ, пытающимся осуществить НВ на ИС, и позволяет на своей основе создать математическую модель конфликта ИС без СЗИ и ИНВ и имитационную модель конфликта ИС без СЗИ и ИНВ, с помощью которых можно будет рассчитать вероятность надежности ИС в течение определенного времени, то есть вероятность непопадания ИС в *«ненадежное состояние»* в течение этого времени, и вероятность нахождения ИС в *«надежном состоянии»* в течение определенного времени.

Математическая модель. Математическая модель конфликта основывается на представлении процесса смены состояний объединенной системы ИС – ИНВ в виде цепи Маркова с конечным числом состояний, переходы между которыми осуществляются по экспоненциальному (пуассоновскому) закону распределения [37,40]. Данная модель является расширением простейшей математической модели ИС, предложенной п. 2.3, в плане учета действий ИНВ в зависимости от его осведомленности и квалификации. На рисунке 3.4 представлены состояния, в которых может находиться ИНВ при подготовке и проведении НВ на ИС, а также возможные переходы из одного состояния в другое.

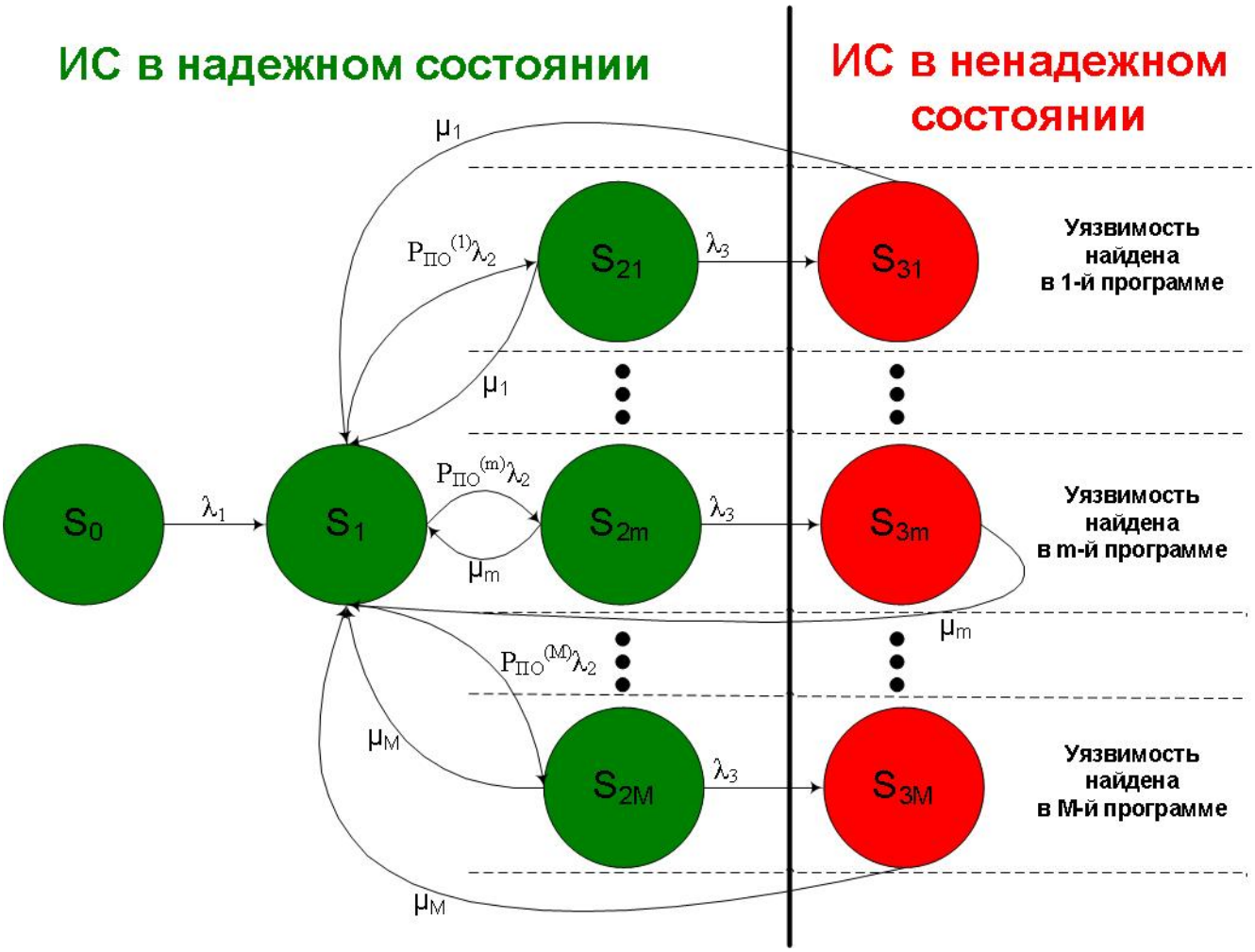

Рисунок 3.4 – Математическая модель конфликта ИС без СЗИ и ИНВ

Узлы цепи соответствуют следующим состояниям: S_0 – у ИНВ отсутствует какая-либо информация об ИС (состояние *«Нет информации об ИС»* в объектно-ориентированной модели); S_1 – у ИНВ есть информация о ПО ИС (состояние *«Информация о ПО ИС»* в объектно-ориентированной модели); S_{2m} – у ИНВ есть информация о ПО ИС и об одной уязвимости в этом ПО, где m - номер программы, в которой была найдена уязвимость ($m \in 1..M$), а M - количество программ в ИС (состояние *«Информация об уязвимости в ПО ИС»* в объектно-ориентированной модели); S_{3m} ($m \in 1..M$) – у ИНВ есть информация о ПО ИС, об одной уязвимости в этом ПО, а также о способе использования этой уязвимости для осуществления НВ на ИС (состояние *«Информация о способе использования уязвимости для НВ на ИС»* в объектно-ориентированной модели).

Вероятности нахождения в указанных состояниях обозначим соответственно $P_0, P_1, P_{21}, ..., P_{2m}, ..., P_{2M}, P_{31}, ..., P_{3m}, ..., P_{3M}$. При этом часть выделенных состояний ($S_0, S_1, S_{21}, ..., S_{2m}, ..., S_{2M}$) агрегируются в состояние *«ИС в надежном состоянии»* (состояние *«Надежное состояние»* в объектно-ориентированной модели), а состояния ($S_{31}, ..., S_{3m}, ..., S_{3M}$) - в состояние *«ИС в ненадежном состоянии»* (состояние *«Ненадежное состояние»* в объектно-ориентированной модели).

Переход из состояния S_0 в S_1 осуществляется с интенсивностью

$$\lambda_1 = \frac{1}{T_{по}}, \tag{3.1}$$

где $T_{по}$.- среднее время, требующееся ИНВ для нахождения информации о ПО ИС.

Переходы из состояния S_1 в состояния S_{2m} ($m \in 1..M$) осуществляются с интенсивностями $P_{ПО}^{(m)} \lambda_2$, где $P_{ПО}^{(m)}$- вероятность нахождения информации об уязвимости в m-й программе, которая равна

$$P_{ПО}^{(m)} = \frac{N_{ср\_конф}^{(m)}}{N_{ср\_конф}}, \tag{3.2}$$

где $N_{ср\_конф}^{(m)}$- среднеарифметическое среднестатистического числа уязвимостей, находящихся в m-й программе ИС $N_{ср}^{(m)}(t)$ (рассчитывается по формуле (2.10)) за время рассмотрения конфликта, а $N_{ср\_конф}$- среднеарифметическое среднестатистического общего числа уязвимостей, находящихся в ПО ИС $N_{ср}(t)$ (рассчитывается по формуле (2.10)) за время рассмотрения конфликта.

Интенсивность обнаружения уязвимостей в ПО ИС

$$\lambda_2 = \frac{N_{ср\_конф}}{T_{уязв}}, \tag{3.3}$$

где $T_{уязв}$ – среднее время, требующееся ИНВ для нахождения информации о всех уязвимостях в ИС. С учетом (3.2) и (3.3) интенсивности переходов из состояния S_1 в состояния $S_{2m}\,(m \in 1..M)$ будут равны

$$P_{ПО}^{(m)} \lambda_2 = \frac{N_{ср\_конф}^{(m)}}{T_{уязв}} \tag{3.4}$$

Переход из состояния $S_{2m}\,(m \in 1..M)$ в состояние $S_{3m}\,(m \in 1..M)$ осуществляется с интенсивностью

$$\lambda_3 = \frac{1}{T_{нв}}, \tag{3.5}$$

где $T_{нв}$ – среднее время, требующееся ИНВ для нахождения информации о способе использования уязвимости в ПО ИС для НВ на ИС.

Для расчета среднего времени, с момента нахождения ИНВ уязвимости до ее устранения из ИС, предлагается прибегнуть к следующим рассуждениям. Предполагается, что время обнаружения (появления) уязвимости в m-й программе $T_{обн\_уязв}^{(m)}$ является случайной величиной, принимающей с равной вероятностью значения из интервала от разности текущего времени $T_{тек}$ и среднего времени жизни уязвимости в m-й программе $T_{жизн\_уязв}^{(m)}$ до текущего времени $T_{тек}$, следовательно, ее математическое ожидание равно $T_{тек} - \frac{T_{жизн\_уязв}^{(m)}}{2}$, а время с момента нахождения ИНВ информации об уязвимости в m-й программе до закрытия $T_{закр}^{(m)}$ этой уязвимости соответственно равно

$$T_{закр}^{(m)} = \frac{T_{жизн\_уязв}^{(m)}}{2}. \tag{3.6}$$

Среднее время жизни уязвимости в m-й программе рассчитывается по формуле:

$$T_{жизн\_уязв}^{(m)} = \frac{T_{в}^{(m)}}{k^{(m)}}, \tag{3.7}$$

где $T_в^{(m)}$ - время, которое требуется вендору m-й программы для создания патча или временного решения, закрывающих уязвимость, с момента ее обнаружения, $k^{(m)}$ - коэффициент, отражающий работу системного администратора по устранению уязвимостей из m-й программы.

Переходы из состояний $S_{2m}\,(m \in 1..M)$ и $S_{3m}\,(m \in 1..M)$ в состояние S_1 осуществляются с интенсивностями

$$\mu_m = \frac{1}{T_{закр}^{(m)}}, \tag{3.8}$$

которые с учетом (3.6) и (3.7) равны

$$\mu_m = \frac{2k^{(m)}}{T_в^{(m)}}. \tag{3.9}$$

Согласно [86] полученная цепь Маркова описывается вектором начального распределения вероятностей нахождения в различных состояниях

$$P(0) = \begin{bmatrix} 1 & 0 & \ldots & 0 \end{bmatrix} \tag{3.10}$$

и переходной матрицей

$$P_{пер}(t) = \exp(Qt),$$

$$Q = \begin{bmatrix} 1-\lambda_1 & \lambda_1 & 0 & \ldots & 0 & 0 & .. & 0 \\ 0 & 1-\sum\limits_{m=1}^{M} P_{ПО}^{(m)}\lambda_2 & P_{ПО}^{(1)}\lambda_2 & \ldots & P_{ПО}^{(M)}\lambda_2 & 0 & \ldots & 0 \\ 0 & \mu_1 & 1-(\mu_1+\lambda_3) & \ldots & 0 & \lambda_3 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \mu_M & 0 & \ldots & 1-(\mu_M+\lambda_3) & 0 & \ldots & \lambda_3 \\ 0 & \mu_1 & 0 & \ldots & 0 & 1-\mu_1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \mu_M & 0 & \ldots & 0 & 0 & .. & 1-\mu_M \end{bmatrix}, \tag{3.11}$$

где Q - матрица интенсивностей переходов между состояниями цепи; t - текущее время, отсчитываемое от начала конфликта. С учетом (3.1), (3.4), (3.5) и (3.9) данная переходная матрица принимает вид

$$P_{пер}(t) = \exp(Qt),$$

$$Q = \begin{bmatrix} 1-\frac{1}{T_{по}} & \frac{1}{T_{по}} & 0 & \dots & 0 & 0 & .. & 0 \\ 0 & 1-\sum_{m=1}^{M}\frac{N_{ср\_конф}^{(m)}}{T_{уязв}} & \frac{N_{ср\_конф}^{(1)}}{T_{уязв}} & \dots & \frac{N_{ср\_конф}^{(M)}}{T_{уязв}} & 0 & \dots & 0 \\ 0 & \frac{2k^{(1)}}{T_{в}^{(1)}} & 1-\left(\frac{2k^{(1)}}{T_{в}^{(1)}}+\frac{1}{T_{нв}}\right) & \dots & 0 & \frac{1}{T_{нв}} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \frac{2k^{(M)}}{T_{в}^{(M)}} & 0 & \dots & 1-\left(\frac{2k^{(M)}}{T_{в}^{(M)}}+\frac{1}{T_{нв}}\right) & 0 & \dots & \frac{1}{T_{нв}} \\ 0 & \frac{2k^{(1)}}{T_{в}^{(1)}} & 0 & \dots & 0 & 1-\frac{2k^{(1)}}{T_{в}^{(1)}} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \frac{2k^{(M)}}{T_{в}^{(M)}} & 0 & \dots & 0 & 0 & .. & 1-\frac{2k^{(M)}}{T_{в}^{(M)}} \end{bmatrix}. \quad (3.12)$$

Распределение вероятностей в момент времени t с начала конфликта рассчитывается согласно [86] по следующей формуле

$$P(t) = P(0)P_{пер}(t). \quad (3.13)$$

Вероятность нахождения ИС в надежном состоянии на n-м шаге конфликта будет равна

$$P_{нах\_над}(t) = 1 - \sum_{m=1}^{M} P_{3m}(t) \quad (3.14)$$

Вероятность нахождения ИС в надежном состоянии за все время конфликта будет равна среднему арифметическому между вероятностями нахождения ИС в надежном состоянии на каждом шаге конфликта

$$P_{нах\_над\_конф} = \frac{\int_{0}^{T_{конф}} P_{нах\_над}(t)dt}{T_{конф}}, \quad (3.15)$$

где $T_{конф}$ - время длительности конфликта. С учетом (3.13-3.14) формула (3.15) принимает вид

$$P_{нах\_над\_конф} = \frac{\int_{0}^{T_{конф}} \left(1 - \sum_{m=1}^{M} \left(P(0)P_{пер}(t)\right)_{3m}\right)dt}{T_{конф}}. \quad (3.16)$$

Для нахождения вероятности надежности ИС следует упростить математическую модель, убрав переходы из состояний $S_{3m}\,(m \in 1..M)$ в состояние S_1, сделав таким образом состояния $S_{3m}\,(m \in 1..M)$ поглощающими. Полученная таким образом цепь Маркова представлена на рисунке 3.5.

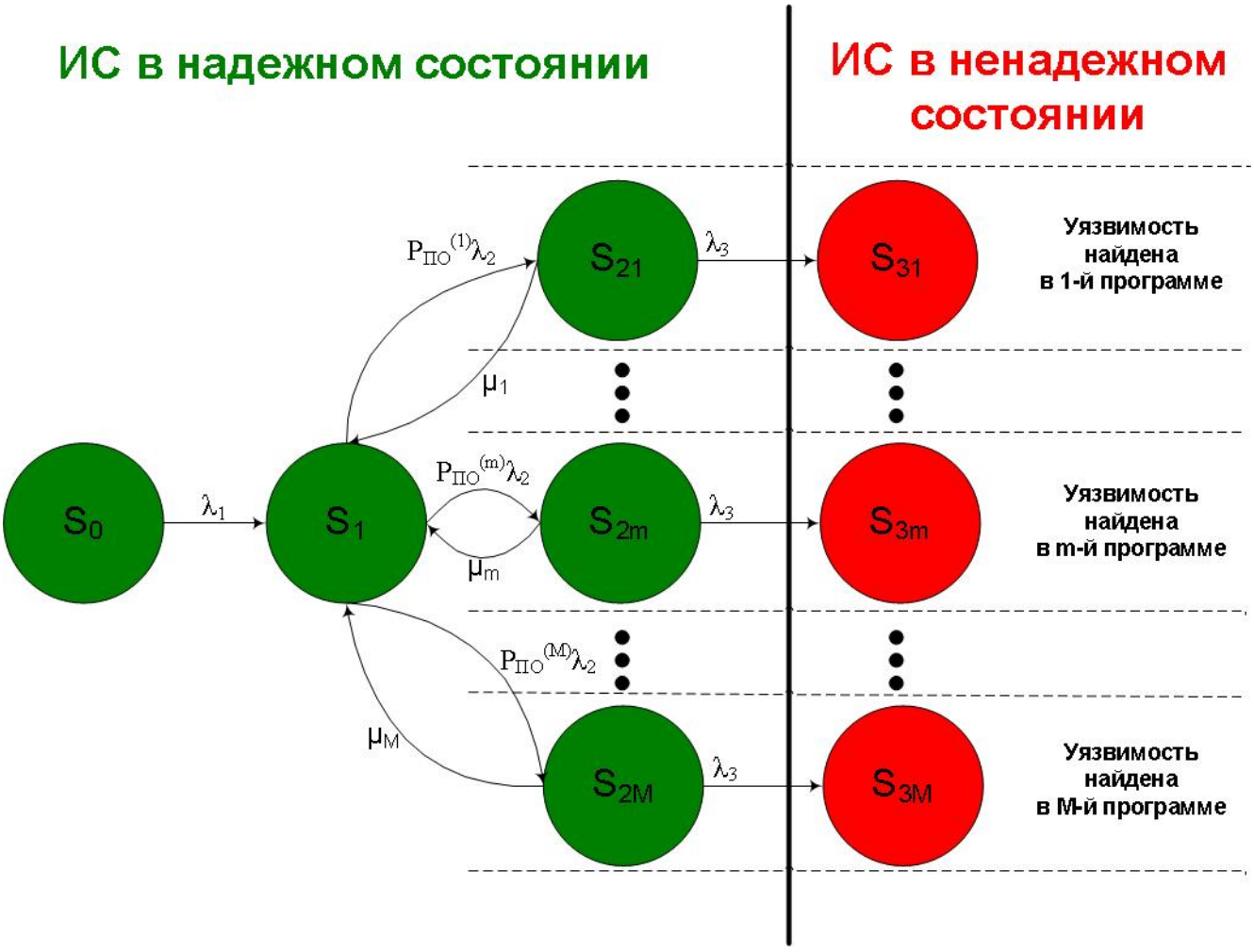

Рисунок 3.5 – Упрощенная математическая модель конфликта ИС – ИНВ

Вектор начального распределения вероятностей нахождения в различных состояниях остается прежним $P(0) = [1 \quad 0 \quad \ldots \quad 0]$, а переходная матрица имеет вид

$$P_{пер}(t) = \exp(Qt),$$

$$Q = \begin{bmatrix} 1-\frac{1}{T_{по}} & \frac{1}{T_{по}} & 0 & \ldots & 0 & 0 & .. & 0 \\ 0 & 1-\sum_{m=1}^{M}\frac{N_{ср\_конф}^{(m)}}{T_{уязв}} & \frac{N_{ср\_конф}^{(1)}}{T_{уязв}} & \ldots & \frac{N_{ср\_конф}^{(M)}}{T_{уязв}} & 0 & \ldots & 0 \\ 0 & \frac{2k^{(1)}}{T_{в}^{(1)}} & 1-\left(\frac{2k^{(1)}}{T_{в}^{(1)}}+\frac{1}{T_{нв}}\right) & \ldots & 0 & \frac{1}{T_{нв}} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \frac{2k^{(M)}}{T_{в}^{(M)}} & 0 & \ldots & 1-\left(\frac{2k^{(M)}}{T_{в}^{(M)}}+\frac{1}{T_{нв}}\right) & 0 & \ldots & \frac{1}{T_{нв}} \\ 0 & 0 & 0 & \ldots & 0 & 1 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & 0 & 0 & .. & 1 \end{bmatrix}. \qquad (3.17)$$

Вероятность попадания ИС в ненадежное состояние в течение времени конфликта будет рассчитываться как

$$P_{ненад} = \sum_{m=1}^{M} P_{3m}\left(T_{конф}\right), \qquad (3.18)$$

а вероятность надежности ИС, то есть вероятность непопадания в ненадежное состояние в течение времени конфликта, соответственно равной

$$P_{над} = 1 - \sum_{m=1}^{M} P_{3m}\left(T_{конф}\right), \qquad (3.19)$$

а с учетом (3.13)

$$P_{над} = 1 - \sum_{m=1}^{M} \left(P(0)P_{пер}\left(T_{конф}\right)\right)_{3m}. \qquad (3.20)$$

Компьютерная имитационная модель с использованием механизма гибридных автоматов (карты Харела). Предложенная математическая модель конфликта ИС и ИНВ учитывает только среднее значение среднестатистического числа уязвимостей, находящихся в ПО ИС, за период конфликта, тогда как в реальности среднестатистическое число уязвимостей в ПО ИС в течение этого периода может меняться (п. 2.2). Кроме того, реально распределение времени переходов в различные состояния может носить произвольный, отличающийся от пуассоновской модели, характер. Также, часто возникает необходимость рассматривать ситуацию, принципиально отличающуюся от дуэльной, когда

конфликт затрагивает несколько участников с каждой стороны (например, ИС атакуют не один, а несколько ИНВ).

Усложнение постановки задачи и необходимость учета всех значимых для описания информационного конфликта факторов неминуемо ведут к возрастающим трудностям при использовании аналитических математических моделей. Это определяет существенную роль средств и компьютерных технологий объектно-ориентированного моделирования для исследования закономерностей конфликта. Одним из доступных компьютерных средств и естественным для описания динамики ситуационного конфликта механизмом реализации компьютерных имитационных моделей информационного конфликта систем является использование формализма гибридных автоматов (карт состояний Харела) и тех возможностей, которые для этих целей предоставляет интегрированная среда MATLAB + Simulink + Stateflow [38-40,87].

Конфликтное взаимодействие ИС – ИНВ в терминах [87] можно описать при помощи SF-модели (рис. 3.6), основывающейся на ранее представленных совокупности диаграмм состояний ИС, системного администратора и ИНВ, приведенных выше (рис 3.1-3.3). Модель состоит из 3-х параллельно функционирующих объектов (*«Sysadmin»* и *«IS»* с одной стороны, *«INV»* с другой стороны), в которых размещены карты состояний, описывающие возможные значения учитываемых факторов и поведение (в зависимости от этих значений) всех сторон, участвующих в конфликте.

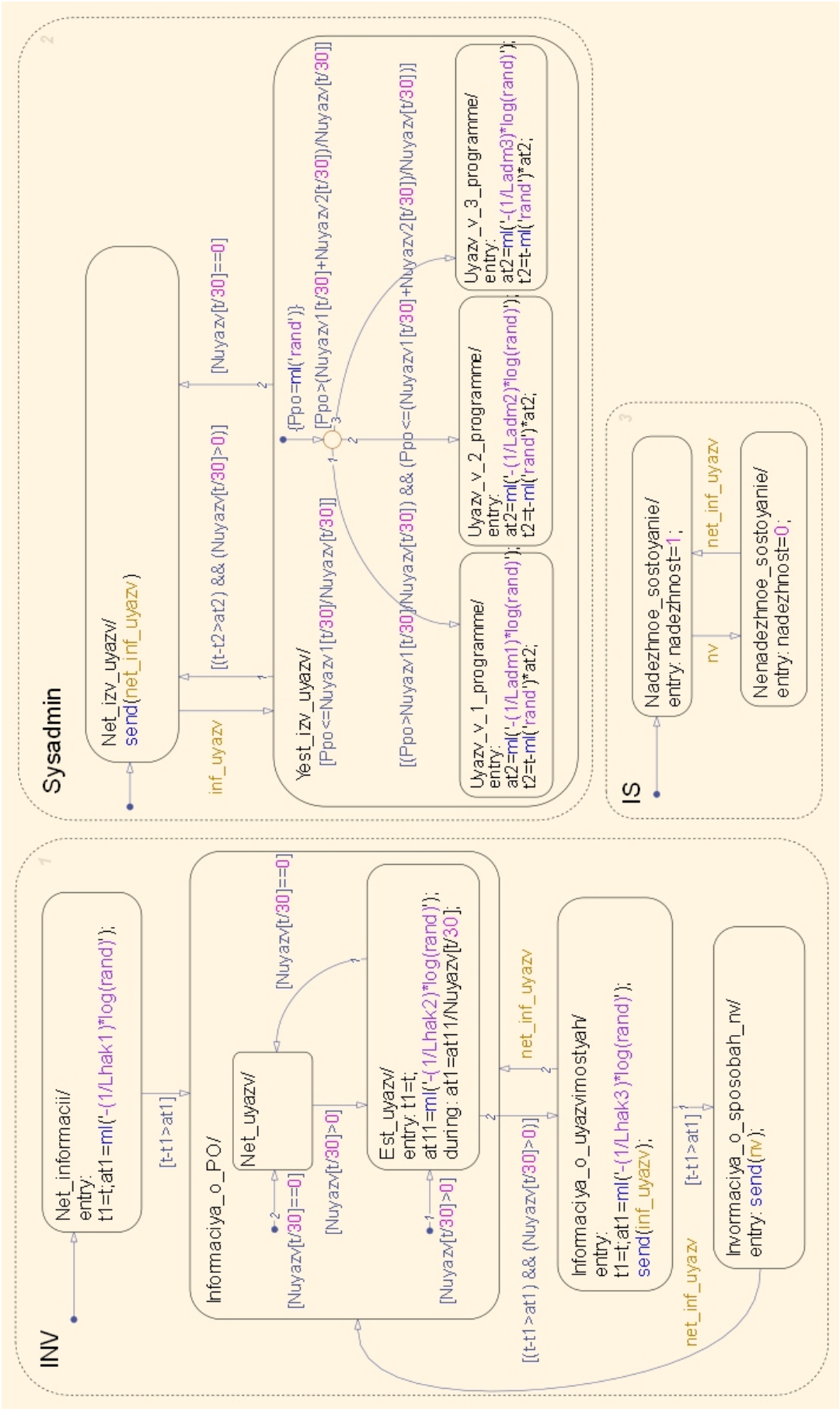

Рисунок 3.6 – SF-модель конфликта ИС без СЗИ и одного ИНВ

В отличие от моделей конфликта, рассмотренных в [87], в представленной модели ни одна из сторон не может добиться абсолютной победы, т.е. в случае перехода ИС в ненадежное состояние, она может снова вернуться в защищенное состояние (восстановиться). Поэтому в ходе эксперимента кроме расчета числа побед сторон конфликта (например, вероятность того, что ИС за период конфликта не перейдет в ненадежное состояние), может быть также рассчитана вероятность нахождения сторон конфликта в определенном состоянии (например, вероятность нахождения ИС в надежном состоянии).

Информационная система (блок *IS*) может находиться в 2-х основных состояниях:

1. Состояние *«Nadezhnoe sostoyanie»* – состояние, при котором ИС считается защищенной от НВ ИНВ (состояние *«Надежное состояние»* в объектно-ориентированной модели).

2. Состояние *«Nenadezhnoe sostoyanie»* – состояние, при котором ИС считается незащищенной от НВ ИНВ (состояние *«Ненадежное состояние»* в объектно-ориентированной модели).

Блок *«Sysadmin»* имитирует работу системного администратора по устранению уязвимостей, известных ИНВ, и предполагает возможность нахождения в 2-х состояниях:

1. Состояние *«Net_izv_INV_uyazv»* – состояние, при котором администратор ИС готовится к закрытию уязвимости в ПО (состояние *«В ПО ИС нет уязвимости, известной ИНВ»* в объектно-ориентированной модели).

2. Состояние *«Yest_izv_INV_uyazv»* – состояние, при котором администратор ИС закрывает уязвимость в ПО, известную ИНВ (состояние *«В ПО ИС есть уязвимость, известная ИНВ»* в объектно-ориентированной модели).

Для определенности предположим, что в ИС установлено 3 вида ПО. Тогда состояние *«Yest_izv_INV_uyazv»* следует разбить на 3 подсостояния (*«Uyazv_v_1_programme»*, *«Uyazv_v_2_programme»*, *«Uyazv_v_3_programme»*). Попадание в одно из этих состояний определяется следующим образом. При входе в состояние *«Yest_izv_INV_uyazv»* переменной *Ppo* присваивается случайная

величина, равномерно распределенная на отрезке от 0 до 1. Данный отрезок разбивается на 3 интервала, каждый из которых соответствует виду ПО, в котором была обнаружена уязвимость. Длина каждого интервала равна отношению среднестатистического числа уязвимостей в ПО, которому соответствует данный интервал, к среднестатистическому числу уязвимостей в ИС (В данном случае отрезок [0,1] разбивается на интервалы $[0, Nuyazv1/Nuyazv]$, $(Nuyazv1/Nuyazv, (Nuyazv1 + Nuyazv2)/Nuyazv]$ и $((Nuyazv1 + Nuyazv2)/Nuyazv, 1])$. Если значение переменной *Ppo* попадает в 1-й интервал, то блок *«Sysadmin»* попадает в подсостояние *«Uyazv_v_1_programme»*, если во в 2-й, то в *«Uyazv_v_2_programme»*, а если в 3-й, то в *«Uyazv_v_3_programme»*. Аналогичным образом можно моделировать случаи, когда в ИС большее или меньшее количество программ.

Упреждающий переход в состояние *«Net_izv_INV_uyazv»* приводит к генерации события *«net_inf_uyazv»*, которое переводит блок *«INV»* из любого состояния, кроме *«Net_informacii»,* в состояние *«Informaciya_o_PO»* (информация об уязвимостях и способах взлома, которой на тот момент обладал ИНВ, теряет актуальность). При этом блок *«IS»* переходит в состояние *«Nadezhnoe sostoyanie»*, что соответствует переходу ИС в надежное состояние.

Состояния, в которых может находиться сторона *«INV»*, полностью соответствуют состояниям ИНВ, определенным в рассмотренной выше объектно-ориентированной модели конфликта ИС и ИНВ:

1. Состояние *«Net_informacii»* – начальное состояние, при котором у ИНВ отсутствует какая-либо информация о системе.
2. Состояние *«Informaciya_o_PO»* – состояние, при котором у ИНВ есть информация о ПО ИС.
3. Состояние *«Informaciya_o_uyazvimosty»* – состояние, при котором у ИНВ есть информация о ПО ИС и об одной уязвимости в этом ПО.
4. Состояние *«Invormaciya_o_sposobah_nv»* – состояние, при котором у ИНВ есть информация о ПО ИС, хотя бы об одной уязвимости в этом ПО, а также

информация о способе использования известной ему уязвимости для осуществления НВ на ИС.

Состояние *«Informaciya_o_PO»* содержит в себе 2 подсостояния: *«Net_uyazv»* и *«Est_uyazv»*. В подсостояние *«Net_uyazv»* блок *«INV»* попадает в случае, если среднестатистическое число уязвимостей в ИС равно 0, в подсостояние *«Est_uyazv»,* если – больше 0. Упреждающий переход в состояние *«Informaciya_o_uyazvimosty»* приводит к генерации события *«inf_uyazv»*, которое переводит блок *«Sysadmin»* в состояние *«Yest_izv_INV_uyazv».* При упреждающем достижении последнего состояния происходит генерация события *«nv»*, переводящее блок *IS* в состояние *«Nenadezhnoe sostoyanie»*, что соответствует переходу ИС в ненадежное состояние.

Время перехода стороны *«INV»* в любое из возможных состояний описывается переменной $t1$. Время пребывания ИНВ (стороны *«INV»*) в состояниях *«Net_informaci»i* и *«Informaciya_o_uyazvimosty»* в отсутствии события *«net_inf_uyazv»* $at1$ является случайным и задается путем вызова m-функции $-(1/Lambda)\cdot\log(rand)$, формирующей случайное число, распределенное по показательному закону с параметром ИНВ $Lambda$ ($Lipnv1 = 1/T_{\text{по}}$, $Lipnv3 = 1/T_{\text{нв}}$ в зависимости от состояния). Время пребывания ИНВ в состоянии «*Informaciya_o_PO»* $at1$ является отношением m-функции к среднестатистическому числу уязвимостей в ИС, то есть равно $-(1/Lipnv2)\cdot\log(rand)/N_{ср}(t)$, где $Lipnv2 = 1/T_{уязв}$ (отношение общего времени, требующегося ИНВ для нахождения информации обо всех уязвимостях в ПО ИС, к числу этих уязвимостей). Переходы из одного состояния в другое осуществляются по условию истечения времени пребывания в каждом из состояний.

Время жизни уязвимости известной ИНВ $at2 = -(1/Lambda)\cdot\log(rand)$, где $Lambda$ равна $Ladm1$, $Ladm2$ или $Ladm3$ ($Ladm1 = k^{(1)}/T_{в}^{(1)}$, $Ladm2 = k^{(2)}/T_{в}^{(2)}$, $Ladm3 = k^{(3)}/T_{в}^{(3)}$) в зависимости от того, в какой программе была обнаружена эта уязвимость. Время обнаружения уязвимости, известной

ИНВ, $t2 = t - rand \cdot at2$, где t - текущее время (Время обнаружения уязвимости – случайная величина, принимающая с равной вероятностью значения из интервала от разницы текущего времени и времени жизни уязвимости до текущего времени). Следует отметить, что принципиальных ограничений на вид законов распределения в данной модели не существует.

Сравнение результатов моделирования. Чтобы сравнить имитационную и математическую модели, предлагается рассчитать вероятность надежности ИС и вероятность нахождения ИС в надежном состоянии для каждого полугодия (предполагается, что ИНВ пытается осуществить НВ на ИС в течение этого периода) в течение 11 лет, начиная с октября 2001 года (время выпуска операционной системы Windows XP), при условии, что в ИС установлена только операционная система Windows XP. Необходимые статистические данные по ПО берутся из [51,77,79]. Коэффициент работы системного администратора для определенности берется $k = 3$. Шаг дискретизации берется равным 0,01 дня. Необходимое количество испытаний для имитационной модели, в соответствии с [87], выбирается равным $N\_isp = 1000$.

Расчет предлагается осуществить для ИНВ 4-х разных уровней квалификации [81-85]:

- ИНВ 1-й категории ($T_{\text{по}} = 60$ дней, $T_{\text{уязв}} = 30$ дней, $T_{\text{нв}} = 30$ дней);
- ИНВ 2-й категории ($T_{\text{по}} = 20$ дней, $T_{\text{уязв}} = 10$ дней, $T_{\text{нв}} = 10$ дней);
- ИНВ 3-й категории ($T_{\text{по}} = 10$ дней, $T_{\text{уязв}} = 5$ дней, $T_{\text{нв}} = 5$ дней);
- ИНВ 4-й категории ($T_{\text{по}} = 5$ дней, $T_{\text{уязв}} = 1$ день, $T_{\text{нв}} = 1$ день).

Ниже приведены графики вероятности надежности (рис. 3.7) и вероятности нахождения в надежном состоянии (рис 3.8) ИС с операционной системой Windows XP при попытке НВ на нее ИНВ 1-й, 2-й, 3-й и 4-й категории.

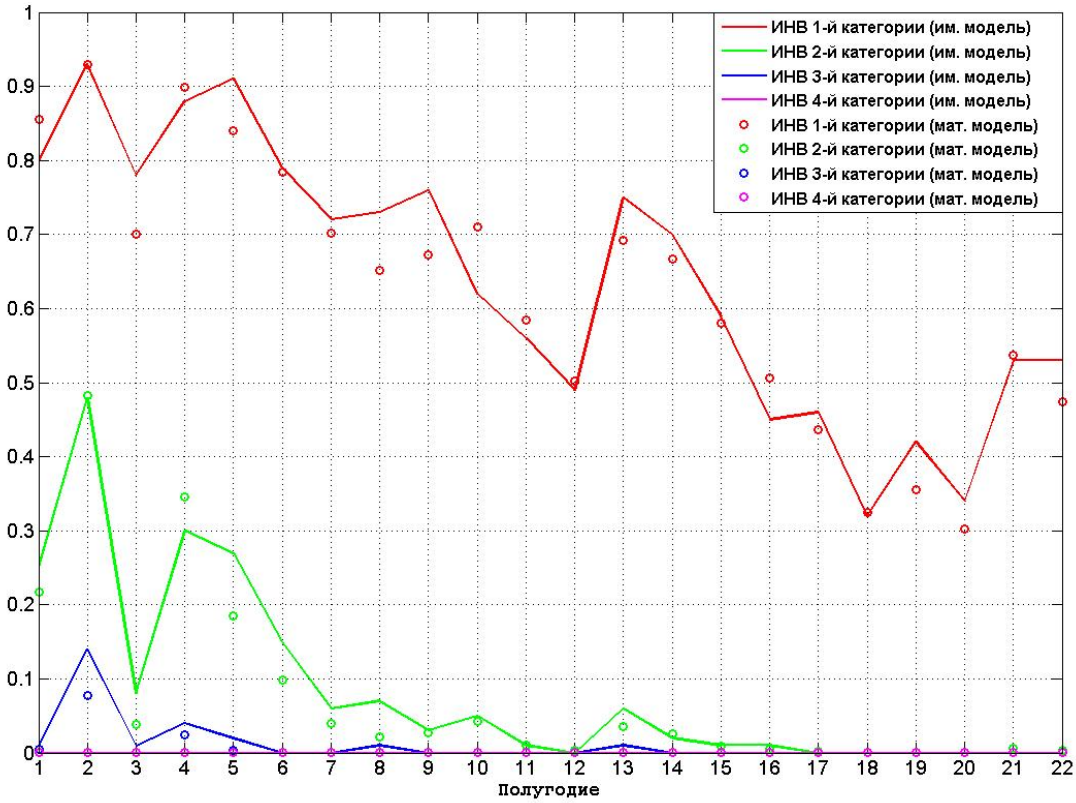

Рисунок 3.7 – Вероятность надежности ИС с операционной системой Windows XP при попытке НВ на нее ИНВ 1-й, 2-й, 3-й и 4-й категории и коэффициенте работы системного администратора $k = 3$

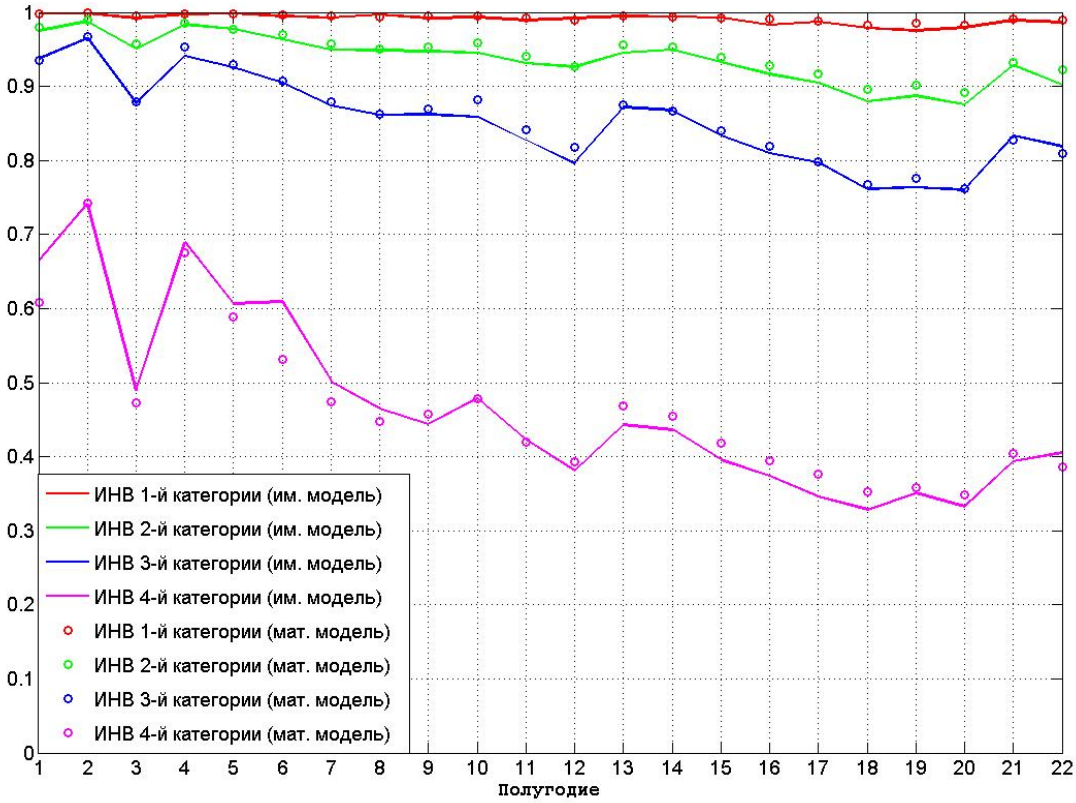

Рисунок 3.8 – Вероятность нахождения в надежном состоянии ИС с операционной системой Windows XP при попытке НВ на нее ИНВ 1-й, 2-й, 3-й и 4-й категории и коэффициенте работы системного администратора $k = 3$

Также, расчет выше приведенных величин предлагается осуществить для ИНВ 2-й категории при разных коэффициентах работы системного администратора: $k = 0{,}5$; $k = 1$; $k = 1{,}5$; $k = 3$ (рис 3.9, 3.10).

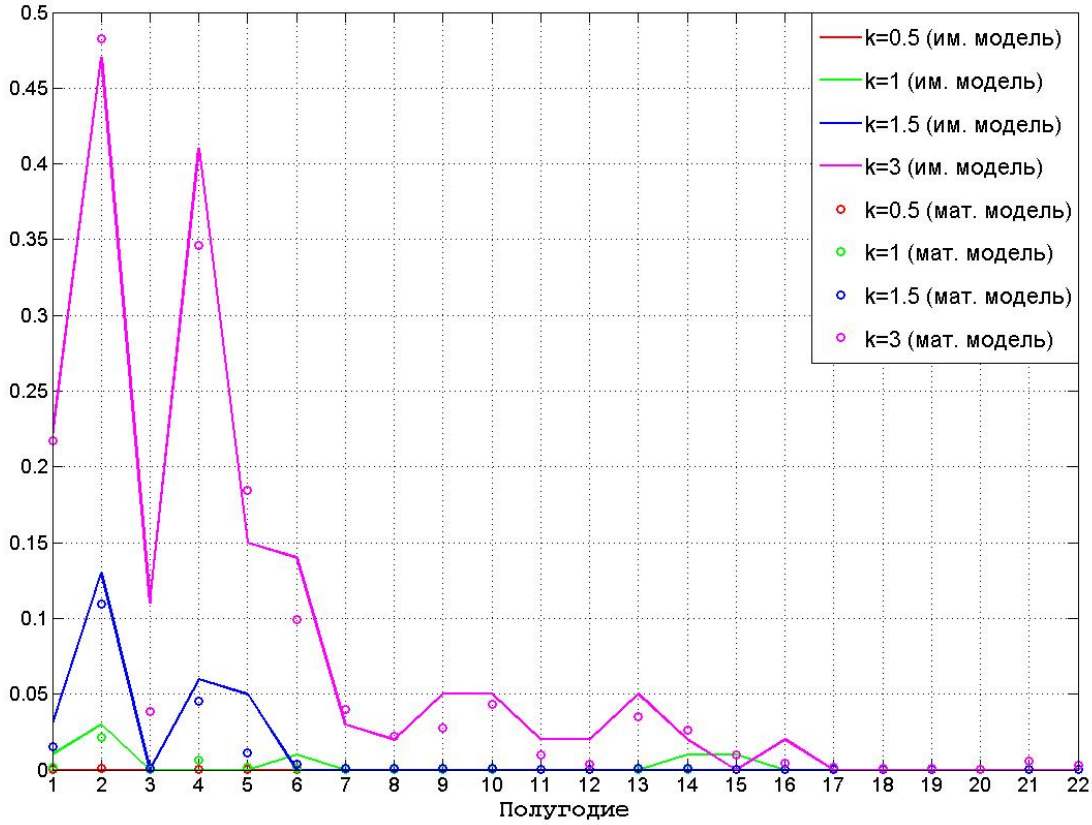

Рисунок 3.9 – Вероятность надежности ИС с операционной системой Windows XP при попытке НВ на нее ИНВ 2-й категории и коэффициентах работы системного администратора: $k = 0{,}5$; $k = 1$; $k = 1{,}5$; $k = 3$

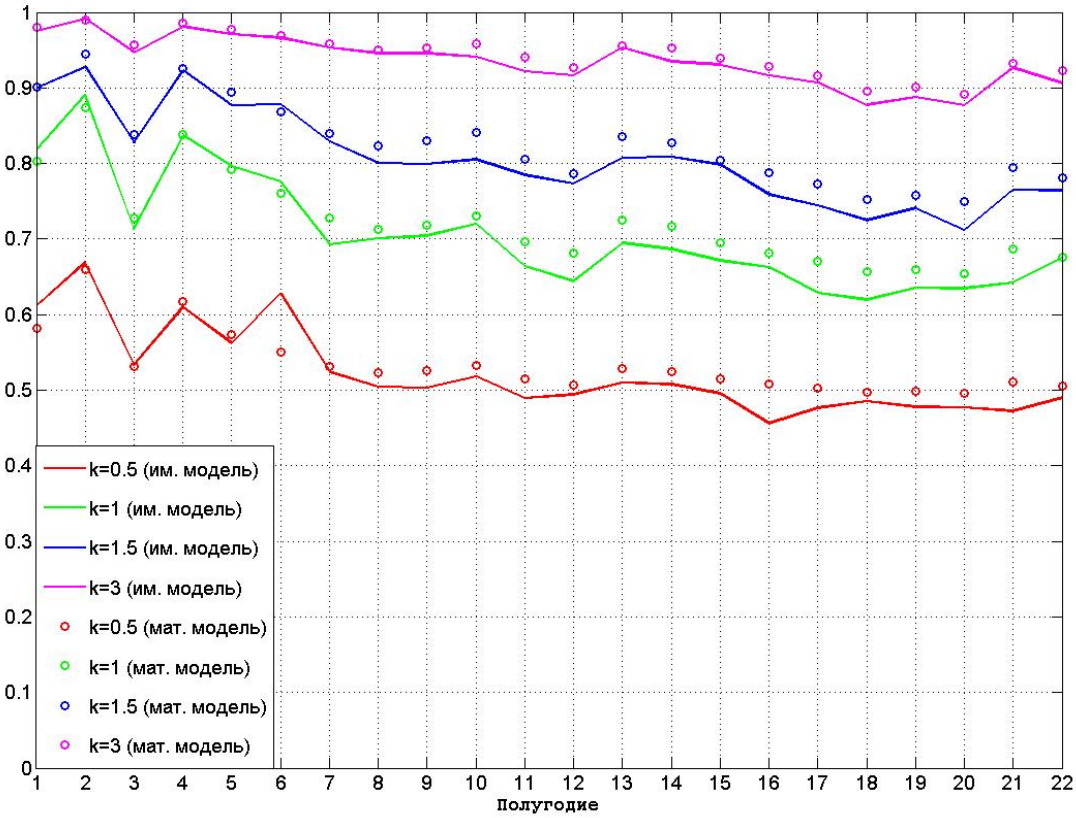

Рисунок 3.10 – Вероятность нахождения в надежном состоянии ИС с операционной системой Windows XP при попытке НВ на нее ИНВ 2-й категории и коэффициентах работы системного администратора: $k = 0{,}5$; $k = 1$; $k = 1{,}5$; $k = 3$

Максимальное среднее абсолютное отклонение вероятности надежности ИС, рассчитанной при помощи математической модели, от вероятности надежности ИС, рассчитанной при помощи имитационной модели, составило 4%, а максимальное среднее абсолютное отклонение вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи математической модели, от вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи имитационной модели – 7%. Разница в результатах при применении математической и имитационной моделей объясняется тем, что математическая модель не учитывает изменение числа уязвимостей в системе в течение конфликта.

С учетом того, что время конфликта предполагается равным полгода (180 дней), последний результат означает, что разница между средним временем

нахождения ИС в надежном состоянии, рассчитанным при помощи математической и имитационной моделей, равна приблизительно 13 дням, что весьма существенно при условии, если каждый день нахождения ИС в ненадежном состоянии несет большие материальные убытки компании, владеющей этой ИС. В конечном же счете целесообразность применения математической модели вместо имитационной может быть обоснованна, исходя из оценки рисков, к которым может привести однократное успешное негативное воздействие на ИС и нахождение ИС в ненадежном состоянии. Дополнительно стоит отметить, что при помощи обеих моделей может быть найдена вероятность нарушения надежности ИС при помощи уязвимостей в конкретном ПО и время необходимое ИНВ для успешного НВ на ИС, а при помощи имитационной модели также количество возможных успешных НВ на ИС и среднее максимальное время постоянного нахождения ИС в ненадежном состоянии (без возвращения в надежное состояние). Данные величины также могут охарактеризовать надежность использования ПО в ИС в условиях конфликтных взаимодействий.

3.2. Модели функционирования информационной системы со средствами защиты информации в условиях конфликтного взаимодействия с одним внешним источником негативных воздействий

Пусть имеется ИС с установленным СЗИ и ПО. ИНВ, преднамеренно негативно воздействующий на ИС, как было показано в главе 1, как правило, проводит предварительную компьютерную разведку программного обеспечения, установленного на ИС, исследует уязвимости в этом программном обеспечении и возможные способы использования этих уязвимостей [4,5], но в случае наличия в ИС СЗИ перед этим он должен определить, какое СЗИ установлено в ИС, какие уязвимости есть в этом СЗИ и какие существуют способы НВ на СЗИ для его преодоления и получения доступа к ПО, установленному в ИС. Соответственно,

квалификация и возможности ИНВ в этом случае описываются средним временем, которое требуется ИНВ:

- для получения информации о СЗИ ИС;
- для получения информации обо всех уязвимостях в СЗИ ИС;
- для получения информации об использовании уязвимости в СЗИ ИС для НВ на СЗИ ИС;
- для получения информации о ПО ИС;
- для получения информации обо всех уязвимостях в ПО ИС;
- для получения информации об использовании уязвимости в ПО ИС для НВ на ИС.

При этом динамика уязвимостей в ИС описывается математической моделью ИС с СЗИ, рассмотренной во 2-й главе.

Объектно-ориентированная модель. Для конструирования объектно-ориентированной модели конфликта так же, как и в случае с ИС без СЗИ, предлагается использовать аппарат языка UML [80]. Ниже на рисунках 3.11-3.14 приведены диаграммы состояний, описывающие поведение ИС, системного администратора и ИНВ.

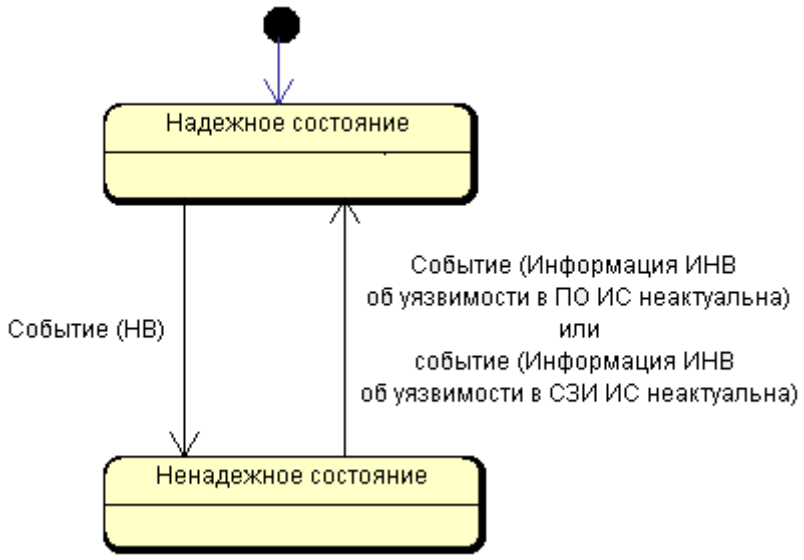

Рисунок 3.11 – Диаграмма состояний ИС с СЗИ в ходе конфликтного взаимодействия с ИНВ

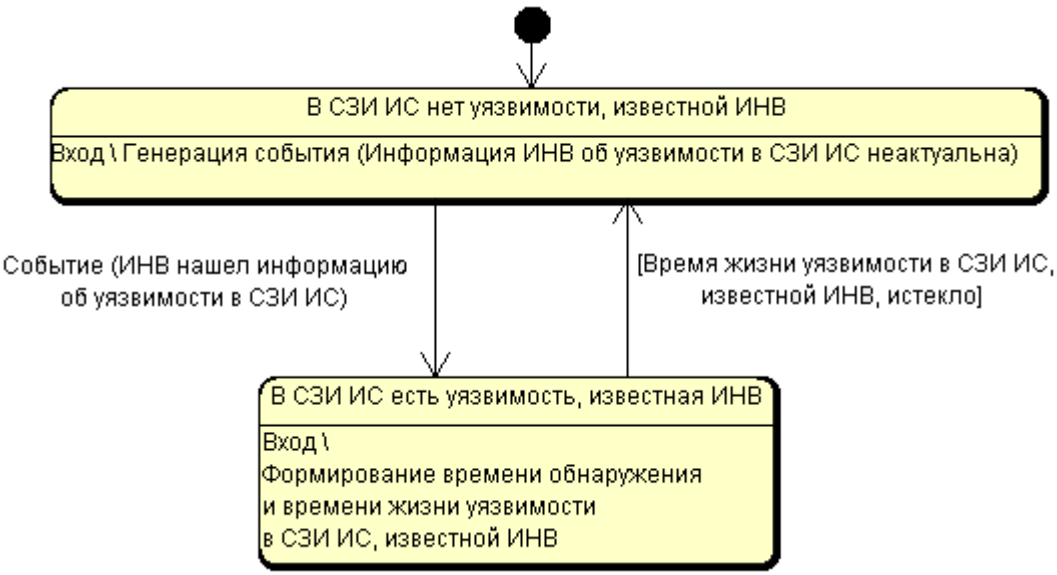

Рисунок 3.12 – Диаграмма состояний, описывающая работу системного администратора по закрытию уязвимостей в СЗИ ИС, в ходе конфликтного взаимодействия ИС с СЗИ с ИНВ

.

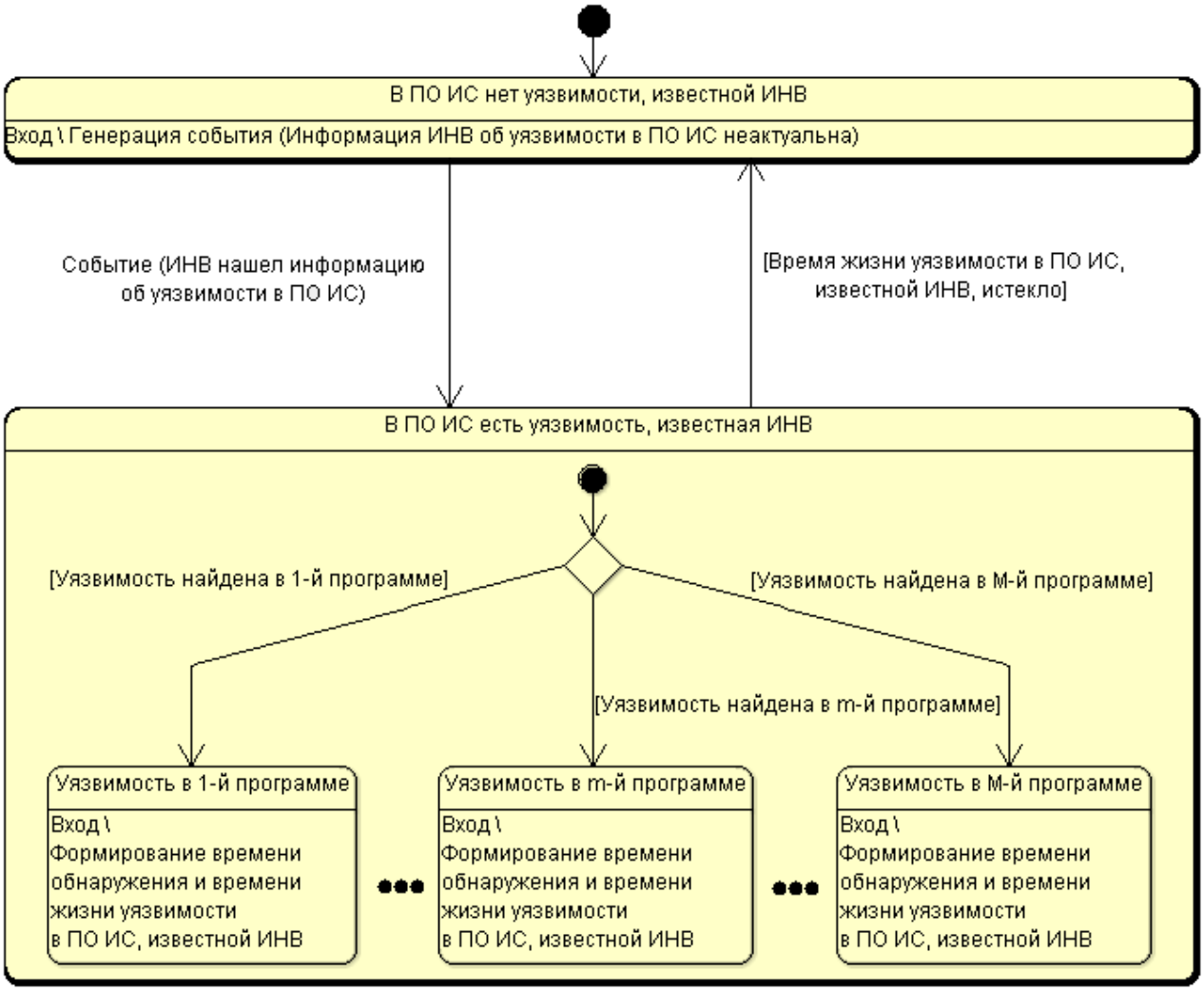

Рисунок 3.13 – Диаграмма состояний, описывающая работу системного администратора по закрытию уязвимостей в ПО ИС, в ходе конфликтного взаимодействия ИС с СЗИ с ИНВ

.

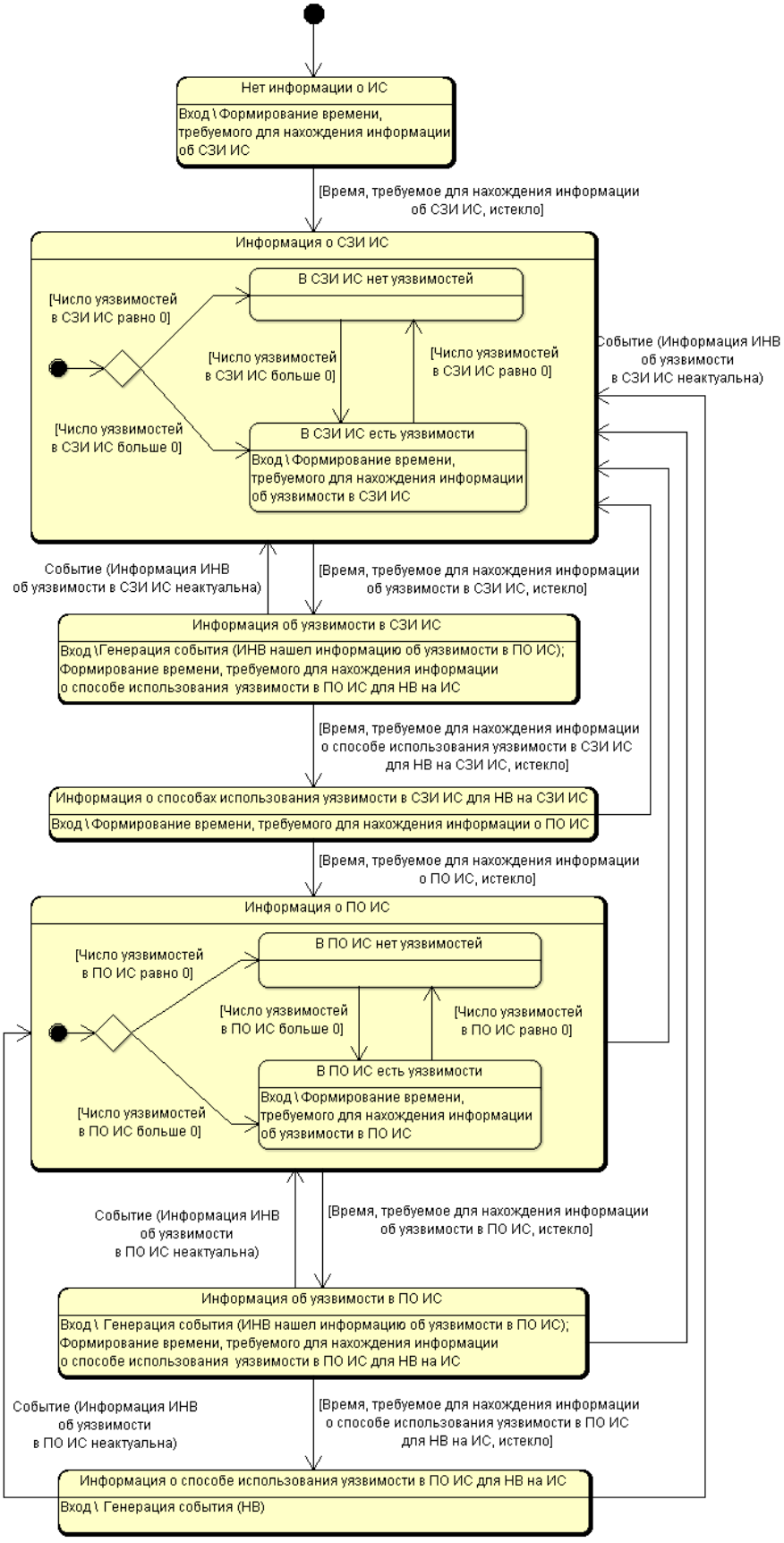

Рисунок 3.14 – Диаграмма состояний ИНВ в ходе конфликтного взаимодействия с ИС с СЗИ

Изменения объектно-ориентированной модели конфликта ИС и одного ИНВ по сравнению со случаем, когда ИС не имеет СЗИ, выглядят следующим образом.

1. Теперь ИС возвращается из состояния *«Ненадежное состояние»* в состояние *«Надежное состояние»* не только при возникновении события *«Информация ИНВ об уязвимости в ПО ИС неактуальна»,* но и при возникновении события *«Информация ИНВ об уязвимости в СЗИ ИС неактуальна»*.

2. Для моделирования процесса закрытия уязвимостей в СЗИ добавлена дополнительная диаграмма состояний (Диаграмма состояний, описывающая работу системного администратора по закрытию уязвимостей в СЗИ ИС, в ходе конфликтного взаимодействия ИС с СЗИ с ИНВ (рис. 3.12)), аналогичная диаграмме состояний, описывающей работу системного администратора по закрытию уязвимостей в ПО ИС (рис. 3.13), с учетом того, что по предположению в ИС есть только один вид СЗИ.

3. В диаграмму состояний ИНВ (рис 3.14) добавлено 3 состояния, отражающие его работу по разведке информации для НВ на СЗИ (*«Информация о СЗИ ИС»*, *«Информация об уязвимости в СЗИ ИС»*, *«Информация о способах использования уязвимости в СЗИ ИС для НВ на СЗИ ИС»*), сходные с аналогичными состояниями, отражающими процесс разведки информации для НВ непосредственно на ИС (за исключением того, что в состоянии *«Информация о способах использования уязвимости в СЗИ ИС для НВ на СЗИ ИС»* не генерируется событие *«НВ»*, а формируется время, требуемое для нахождения информации о ПО ИС), и предшествующие им.

4. В диаграмму состояний ИНВ (рис 3.14) добавлены переходы в состояние *«Информация об уязвимости в СЗИ ИС»* из всех последующих состояний по событию *«Информация ИНВ об уязвимости в СЗИ ИС неактуальна»*.

Математическая модель. Для учета математической моделью конфликта ИС и одного ИНВ наличия в ИС СЗИ в ней производятся изменения, подобные изменениям в объектно-ориентированной модели, а именно добавляются состояния, отражающие разведку информации о СЗИ ИС для НВ на СЗИ ИС, и переходы в состояния, соответствующие наличию информации о СЗИ ИС, из всех последующих состояний. Данная математическая модель с внесенными изменениями изображена на рисунке 3.15.

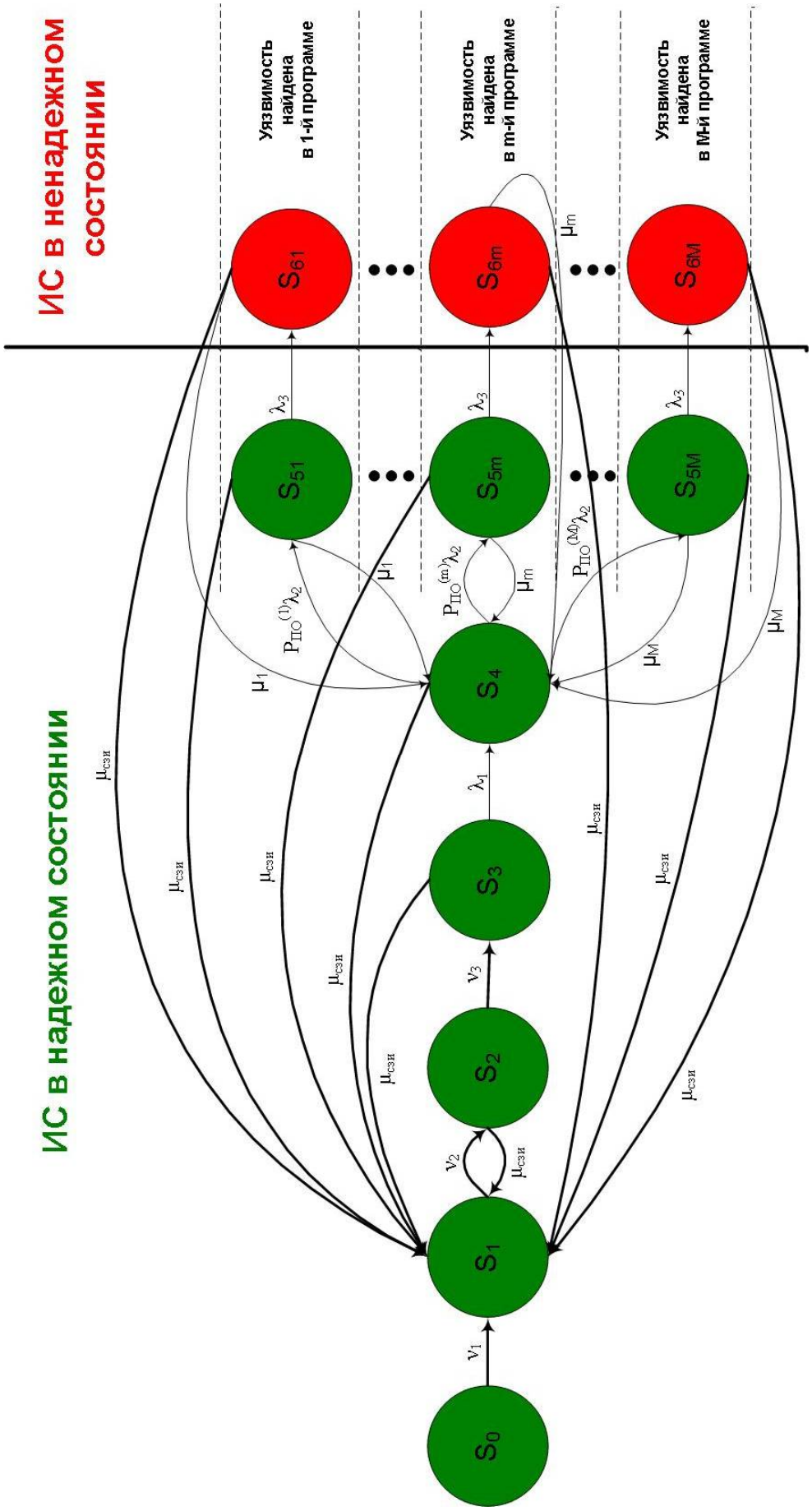

Рисунок 3.15 – Математическая модель конфликта ИС с СЗИ и ИНВ

Интенсивность перехода из состояния S_0 в состояние S_1

$$\nu_1 = \frac{1}{T_{сзи}}, \tag{3.21}$$

где $T_{сзи}$ - среднее время, требующееся ИНВ для нахождения информации о СЗИ ИС.

Интенсивность перехода из состояния S_1 в состояние S_2

$$\nu_2 = \frac{N_{ср\_конф}^{(СЗИ)}}{T_{уязв\_сзи}}, \tag{3.22}$$

где $T_{уязв\_сзи}$ - среднее время, требующееся ИНВ для нахождения информации об одной уязвимости в СЗИ ИС, а $N_{ср\_конф}^{(СЗИ)}$- среднеарифметическое среднестатистического числа уязвимостей $N^{(СЗИ)}(t)$, находящихся в СЗИ ИС, за время рассмотрения конфликта.

Интенсивность перехода из состояния S_2 в состояние S_3

$$\nu_3 = \frac{1}{T_{нв\_сзи}}, \tag{3.23}$$

где $T_{нв\_сзи}$ - среднее время, требующееся ИНВ для нахождения информации о способах использования уязвимостей в СЗИ ИС для НВ на СЗИ ИС.

Интенсивность перехода из состояний $S_2, S_3, S_4, S_{5m}(m \in 1..M), S_{6m}(m \in 1..M)$ в состояние S_1 выводится аналогично (3.9).

$$\mu_{сзи} = \frac{2k^{(СЗИ)}}{T_в^{(СЗИ)}}, \tag{3.24}$$

где $T_в^{(СЗИ)}$ - время, которое требуется вендору СЗИ для создания патча или временного решения, закрывающих уязвимость в СЗИ, с момента ее обнаружения, $k^{(СЗИ)}$ - коэффициент, отражающий работу системного администратора по устранению уязвимостей из СЗИ. Все остальные интенсивности переходов

определяются так же, как и в математической модели конфликта ИС без СЗИ с одним ИНВ (по формулам (3.1),(3.4)(3.5) и (3.9)).

Полученная цепь Маркова описывается вектором начального распределения вероятностей нахождения в различных состояниях $P(0)=\begin{bmatrix}1 & 0 & \dots & 0\end{bmatrix}$ и переходной матрицей $P_{пер}$ (с учетом (3.1),(3.4),(3.5),(3.9),(3.21-3.24)), которая рассчитывается из тех же соображений, что и переходная матрица для случая конфликта ИС без СЗИ с одним ИНВ.

Аналогичным образом рассчитывается время нахождения ИС в надежном состоянии

$$P_{нах\_над\_конф}=\frac{\int_0^{T_{конф}}\left(1-\sum_{m=1}^{M}\left(P(0)P_{пер}(t)\right)_{6m}\right)dt}{T_{конф}} \qquad (3.25)$$

Для расчета надежности ИС предлагается так же, как и в предыдущем случае, упростить модель, убрав переходы из состояний $S_{6m}(m\in 1..M)$ в состояния S_2 и S_4. Полученная математическая модель представлена на рисунке 3.16.

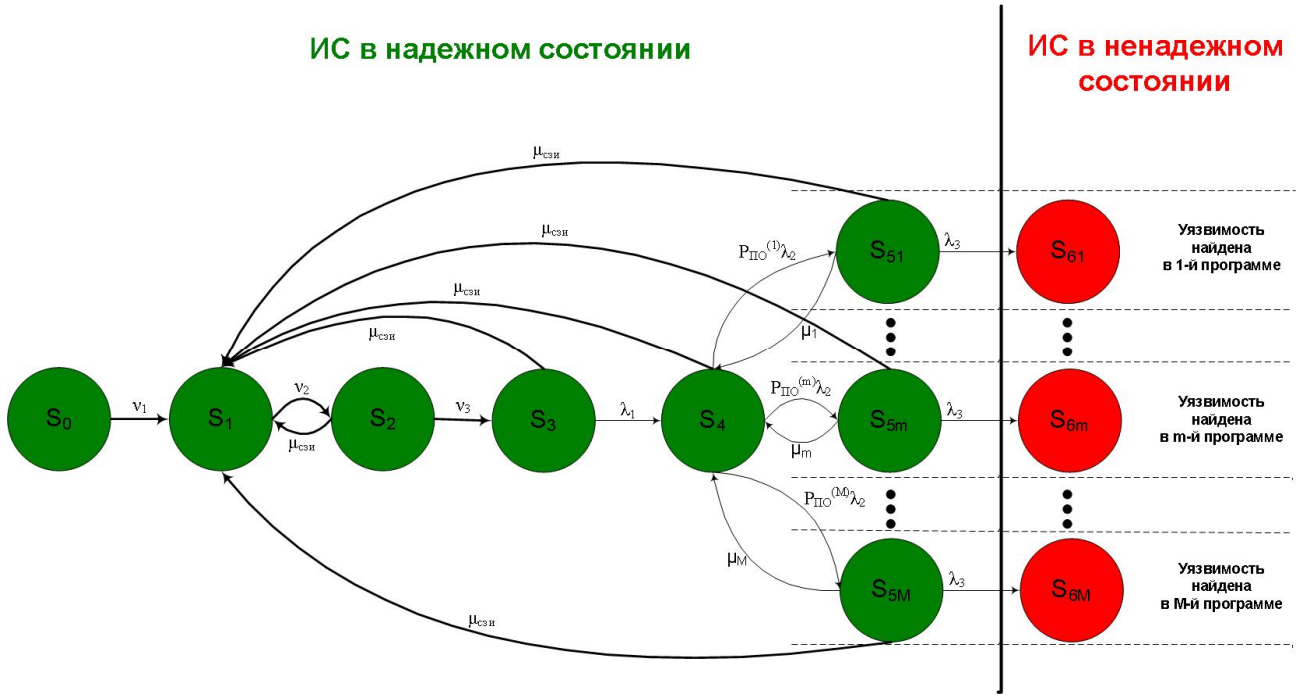

Рис 3.16 – Упрощенная математическая модель конфликта ИС с СЗИ и одного ИНВ

Вероятность надежности ИС в данном случае соответственно равна

$$P_{над} = 1 - \sum_{m=1}^{M} \left(P(0) P_{пер}\left(T_{конф}\right) \right)_{6m} . \qquad (3.26)$$

Компьютерная имитационная модель с использованием механизма гибридных автоматов (карты Харела). Для учета имитационной моделью конфликта ИС и одного ИНВ наличия в ИС СЗИ в ней производятся изменения, сходные с изменениями в объектно-ориентированной модели. Добавляется еще одно событие, переводящее блок *«IS»* в состояние *«Nadezhnoe_sostoyanie»* – *«net_inf_uyazv_szi»* (отражающее потерю актуальности информации, которой владеет ИНВ об уязвимости в СЗИ). Блок *«Sysadmin»* разбивается на 2 подблока, один из которых, как и прежде, моделирует процесс закрытия уязвимостей в ПО ИС, известных ИНВ, а другой, аналогичный (с учетом того, что в ИС установлен один вид СЗИ), моделирует процесс закрытия уязвимостей в СЗИ ИС, известных ИНВ. В блок *«INV»* добавляются состояния, отражающие разведку информации о СЗИ ИС для НВ на СЗИ ИС (*«Informaciya_o_SZI»*, *«Informaciya_o_uyazvimostyah_SZI»*, *«Informaciya_o_sposobah_nv_SZI»*), переходы блока *«INV»* в состояние, соответсвующее наличию информации о СЗИ ИС (*«Informaciya_o_SZI»*), из всех последующих состояний (переход осуществляется при возникновении события *«net_inf_uyazv_szi»*).

Отдельные блоки имитационной модели *«INV»*, *«Sysadmin»* и *«IS»* с внесенными изменениями изображены на рисунках 3.17, 3.18 и 3.19 соответственно. Общая SF-модель конфликта ИС с СЗИ и одного ИНВ изображена на рисунке 3.20.

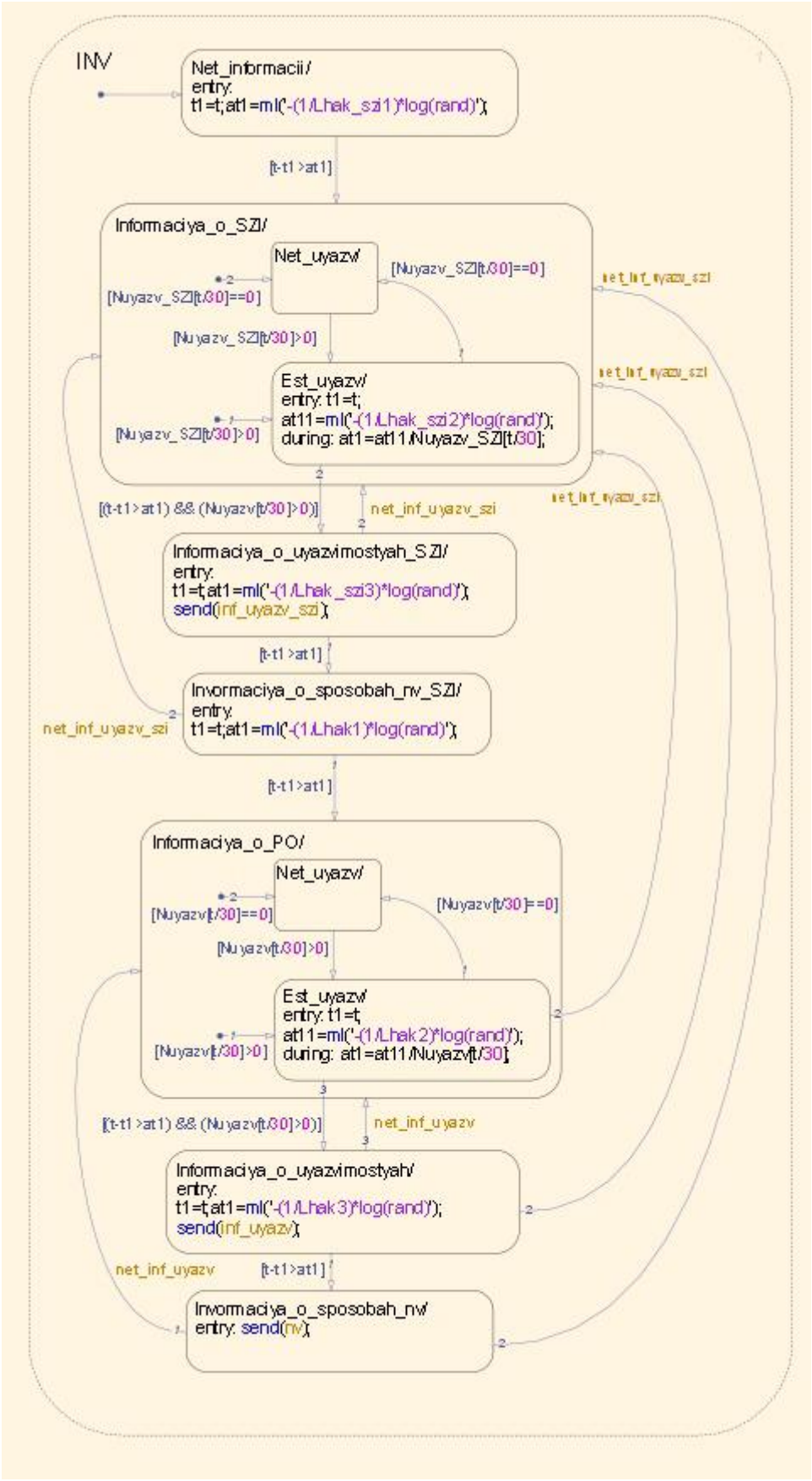

Рисунок 3.17 – Блок, отражающий действия ИНВ, (*INV*) имитационной модели конфликта ИС с СЗИ и ИНВ

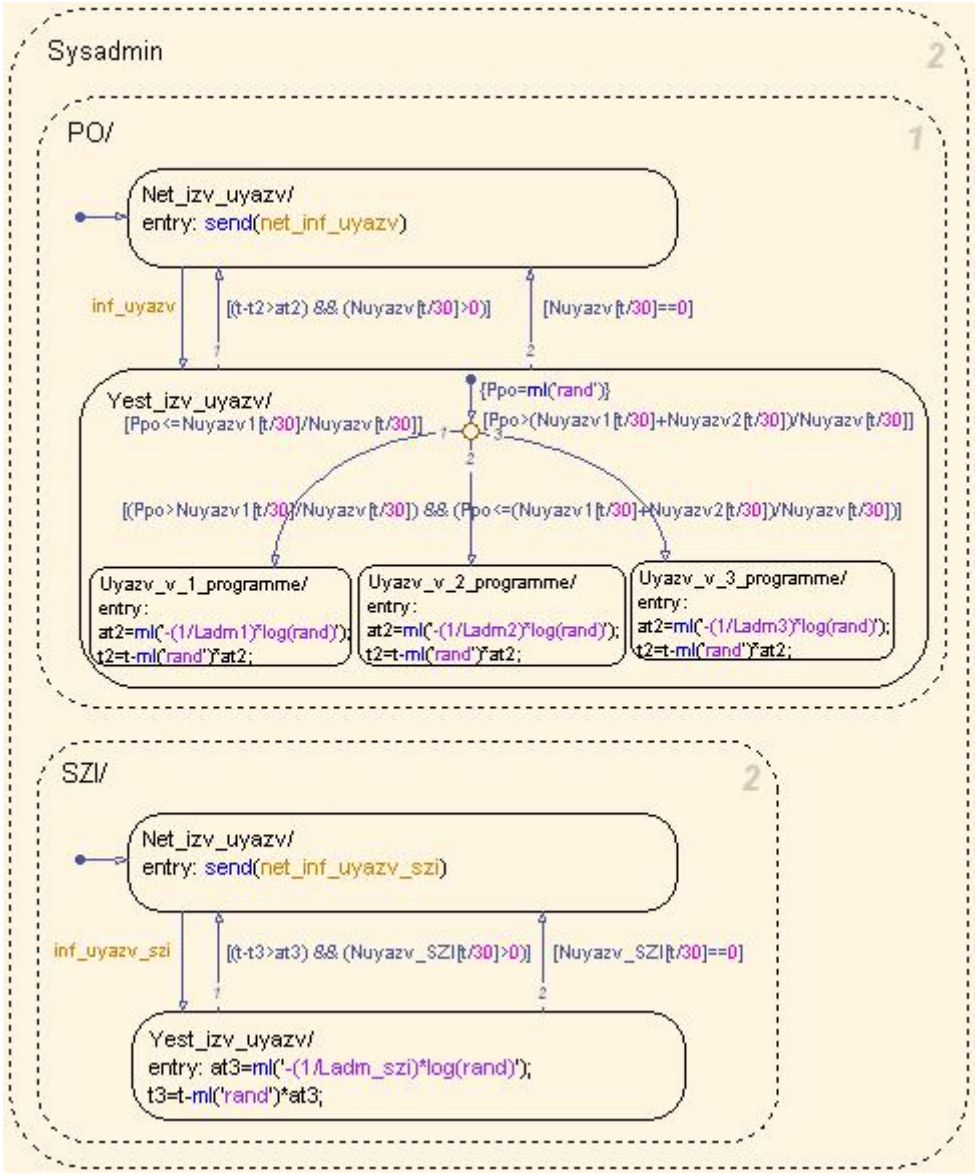

Рисунок 3.18 – Блок, отражающий действия системного администратора, (*Sysadmin*) имитационной модели конфликта ИС с СЗИ и ИНВ

Рисунок 3.19 – Блок, отражающий изменение состояний ИС, (*IS*) имитационной модели конфликта ИС с СЗИ и ИНВ

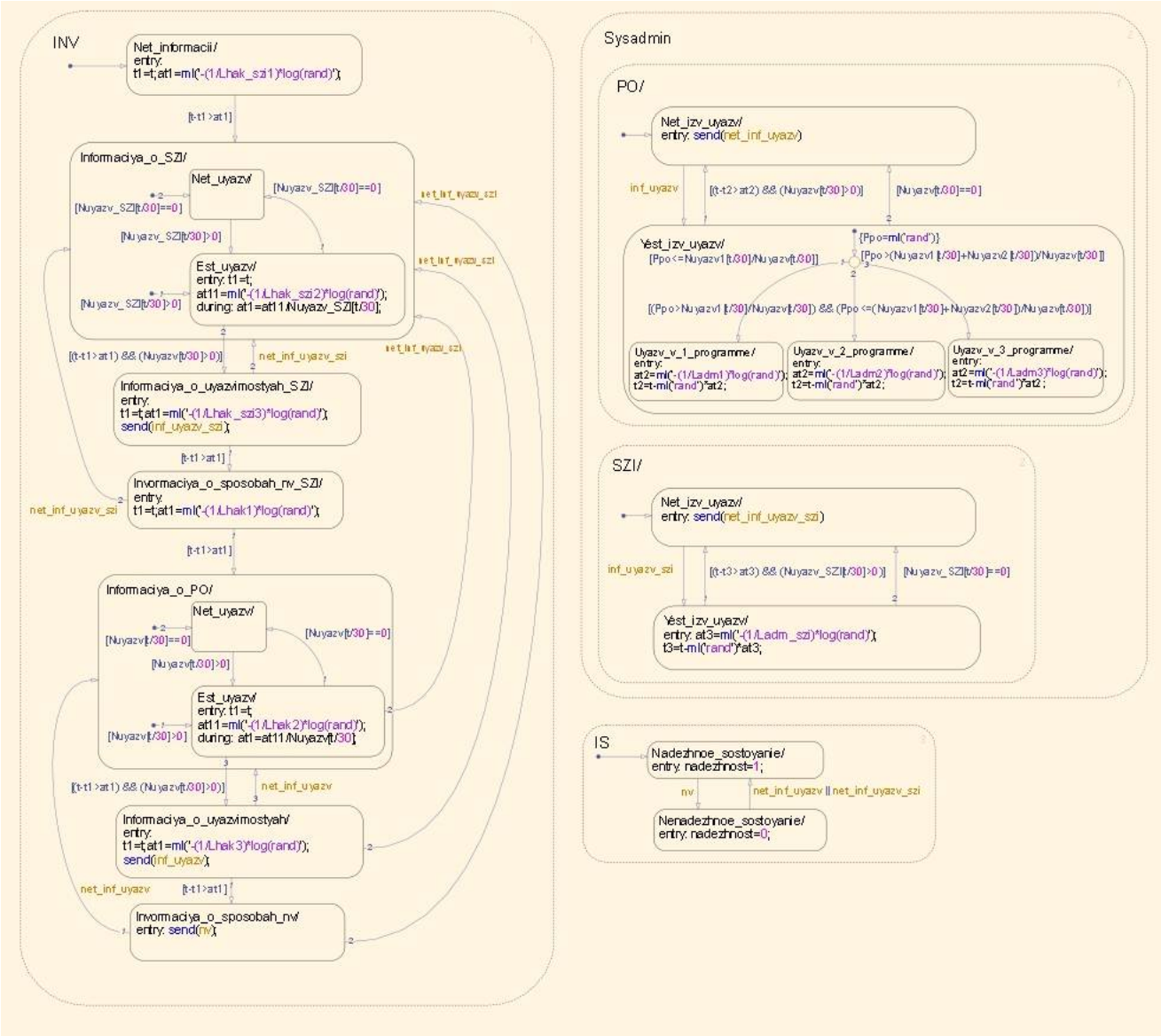

Рисунок 3.20 – SF-модель конфликта ИС с СЗИ и одного ИНВ

Сравнение результатов моделирования. В качестве примера предлагается рассчитать вероятность нахождения в надежном состоянии и вероятность надежности ИС с СЗИ типа сетевой экран семейства Cisco IOS 12.x с установленной в ней операционной системой Windows XP. Расчет предлагается производить для каждого полугодия в течение 11 лет, начиная с октября 2001 года (время выпуска операционной системы Windows XP).

Необходимые статистические данные по ПО берутся из [51,77,79]. Коэффициент работы системного администратора для определенности берется $k = 3$. Шаг дискретизации берется равным 0,01 дня. Необходимое количество

испытаний для имитационной модели, в соответствии с [87], выбирается равным $N\_isp = 1000$. Расчет предлагается осуществить для ИНВ 4-х разных уровней квалификации [81-85]:

- ИНВ 1-й категории ($T_{\text{по}} = T_{сзи} = 60$ дней, $T_{\text{уязв}} = T_{уязв\_сзи} = 30$ дней, $T_{\text{нв}} = T_{нв\_сзи} = 30$ дней);
- ИНВ 2-й категории ($T_{\text{по}} = T_{сзи} = 20$ дней, $T_{\text{уязв}} = T_{уязв\_сзи} = 10$ дней, $T_{\text{нв}} = T_{нв\_сзи} = 10$ дней);
- ИНВ 3-й категории ($T_{\text{по}} = T_{сзи} = 10$ дней, $T_{\text{уязв}} = T_{уязв\_сзи} = 5$ дней, $T_{\text{нв}} = T_{нв\_сзи} = 5$ дней);
- ИНВ 4-й категории ($T_{\text{по}} = T_{сзи} = 5$ дней, $T_{\text{уязв}} = T_{уязв\_сзи} = 1$ день, $T_{\text{нв}} = T_{нв\_сзи} = 1$ день);

Ниже приведены графики вероятности надежности (рис. 3.21) и вероятности нахождения в надежном состоянии (рис. 3.22) ИС с операционной системой Windows XP при попытке НВ на нее ИНВ 1-й, 2-й, 3-й и 4-й категории.

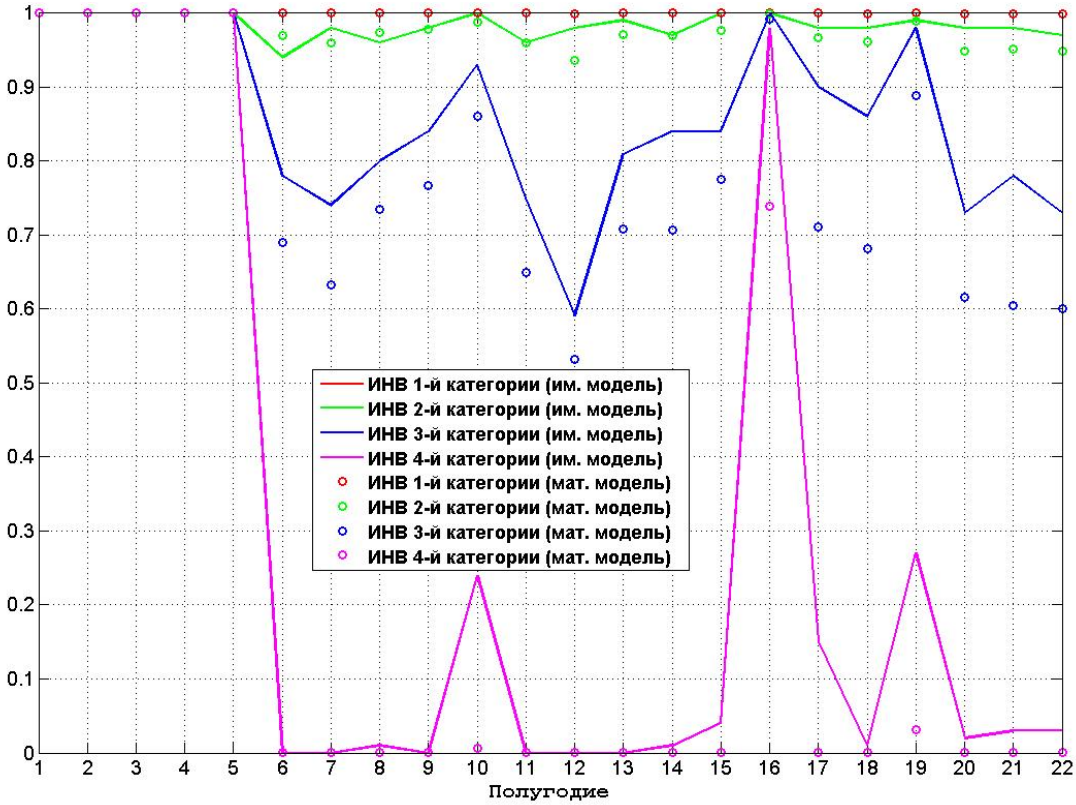

Рисунок 3.21 – Вероятность надежности ИС с СЗИ типа сетевой экран семейства Cisco IOS 12.x и операционной системой Windows XP при попытке НВ на нее ИНВ 1-й, 2-й, 3-й и 4-й категории и коэффициенте работы системного администратора $k = 1$

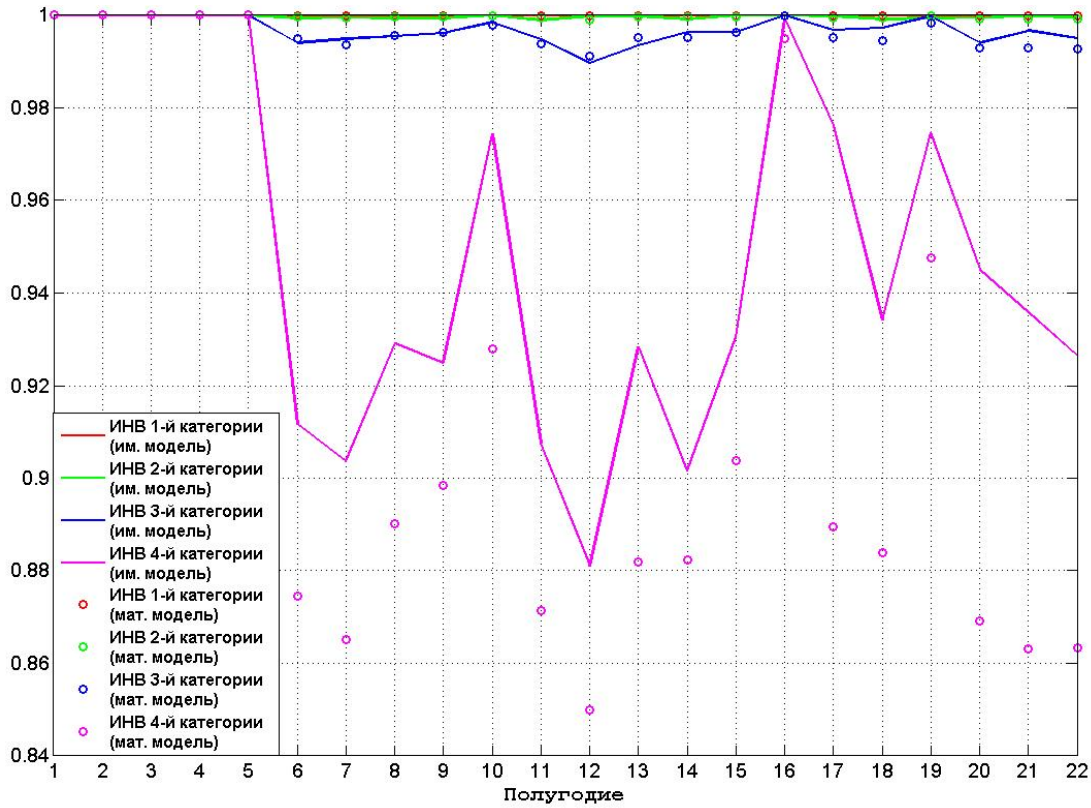

Рисунок 3.22 – Вероятность нахождения в надежном состоянии ИС с СЗИ типа сетевой экран семейства Cisco IOS 12.x и операционной системой Windows XP при попытке НВ на нее ИНВ 1-й, 2-й, 3-й и 4-й категории и коэффициенте работы системного администратора $k = 1$

Максимальное среднее абсолютное отклонение вероятности надежности ИС, рассчитанной при помощи математической модели, от вероятности надежности ИС, рассчитанной при помощи имитационной модели, составило 8%, а максимальное среднее абсолютное отклонение вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи математической модели, от вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи имитационной модели – 3%. С учетом того, что время конфликта предполагается равным полгода (180 дней), последний результат означает, что разница между средним временем нахождения ИС в надежном состоянии, рассчитанным при помощи математической и имитационной моделей, равна приблизительно 5 дням, что весьма существенно при условии, если каждый день нахождения ИС в ненадежном состоянии несет большие риски компании, владеющей этой ИС.

3.3 Модели функционирования информационной системы без средств защиты информации в условиях конфликтного взаимодействия с коалицией внешних источников негативных воздействий без инсайдера

Пусть имеется ИС с установленным ПО. На ИС негативно воздействует коалиция ИНВ, состоящая из R ИНВ. Каждый ИНВ, как и в модели конфликта ИС без СЗИ с одним ИНВ, последовательно добывает информацию о ПО, установленном в ИС, об уязвимости в этом ПО и о способах использования этой уязвимости для НВ на ИС, но при этом в коалиции ИНВ происходит обмен информацией, то есть если один из ИНВ добыл информацию о ПО, об уязвимости или о способах ее использования, то эта информация становится известна и остальным участникам коалиции [39].

Объектно-ориентированная модель. Объектно-ориентированная модель конфликта ИС без СЗИ и коалиции ИНВ без инсайдера строится аналогично объектно-ориентированной модели конфликта ИС без СЗИ и одного ИНВ. При этом в диаграмму состояний ИНВ (рис. 3.23) добавляются переходы, связанные с получением информации от других ИНВ, входящих в коалицию, то есть добавляются следующие переходы.

1. Из состояния *«Нет информации»* в состояние «*Есть информация о ПО ИС»* при событии *«ИНВ нашел информацию о ПО ИС»*
2. Из состояния *«Есть информация о ПО ИС»* в состояние *«Есть информация об уязвимости в ПО ИС»* при событии *«ИНВ нашел информацию об уязвимости в ПО ИС»*
3. Из состояния *«Есть информация об уязвимости в ПО ИС»* в состояние *«Есть информация о способе использования уязвимости в ПО ИС для НВ на ИС»* при событии *«НВ»*.

Событие *«ИНВ нашел информацию о ПО ИС»* генерируется при входе любого ИНВ в состояние *«Есть информация о ПО ИС»*. В общую объектно-ориентированную модель конфликта включается столько же диаграмм состояний ИНВ сколько ИНВ входят в коалицию, негативно воздействующую на ИС.

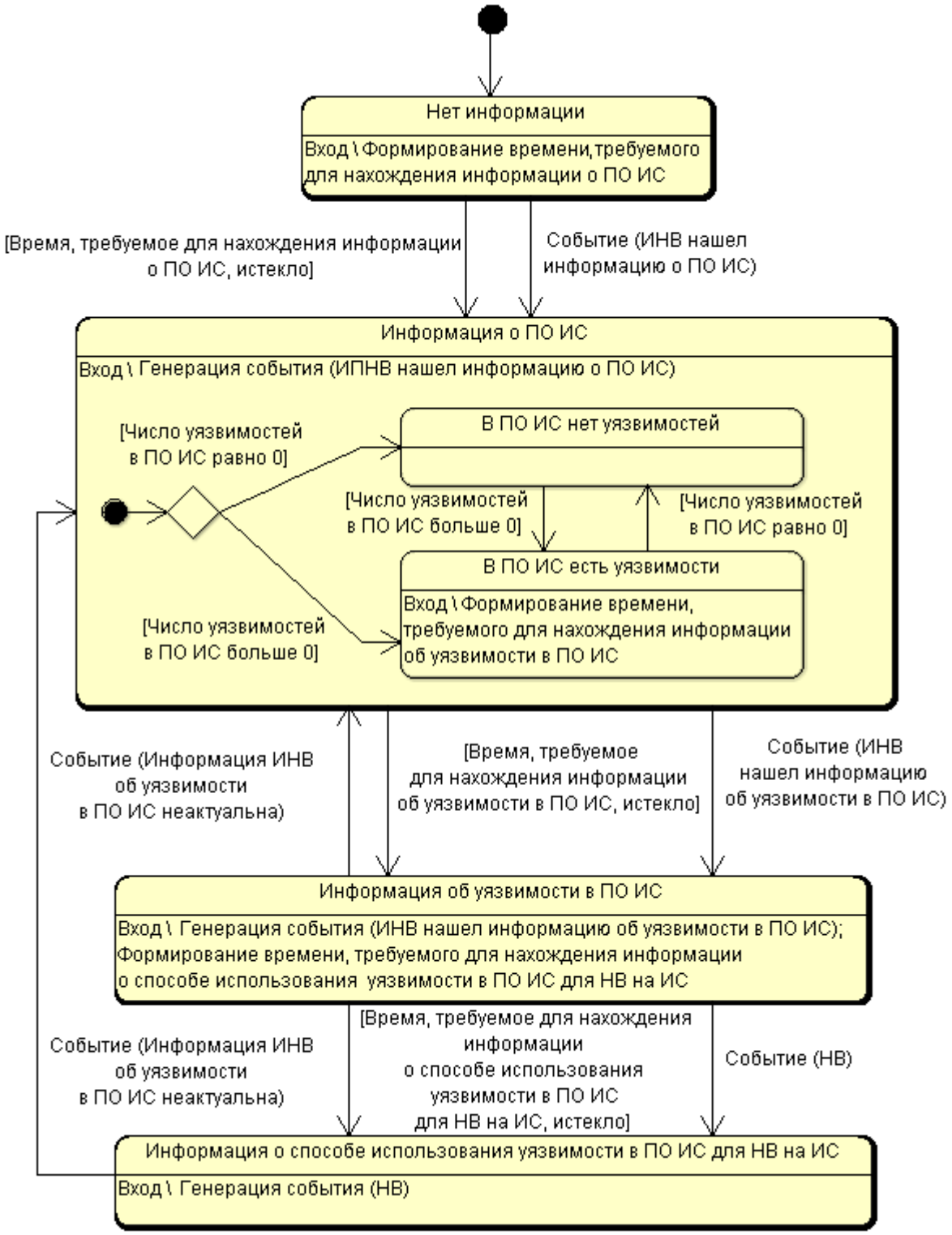

Рисунок 3.23 – Диаграмма состояний ИНВ в ходе конфликтного взаимодействия ИС без СЗИ с коалицией ИНВ без инсайдера

Математическая модель. Математическая модель конфликта ИС без СЗИ с коалицией ИНВ без инсайдера аналогична модели конфликта ИС без СЗИ с

одним ИНВ. При этом интенсивности переходов $\lambda_1, \lambda_2, \lambda_3$, описывающих последовательное добывание информации о ПО ИС, об уязвимости в ПО ИС и о способе ее использования для НВ на ИС, согласно [88] рассчитываются как суммы этих же интенсивностей для каждого ИНВ ($\lambda_1^{(r)}, \lambda_2^{(r)}, \lambda_3^{(r)}$, $r \in 1..R$), которые, соответственно, рассчитываются при помощи формул.

$$\lambda_1 = \sum_r^R \lambda_1^{(r)}\ ,\ \lambda_2 = \sum_r^R \lambda_2^{(r)}\ ,\ \lambda_3 = \sum_r^R \lambda_3^{(r)}\ ,$$

$$\lambda_1^{(r)} = \frac{1}{T_{по}^{(r)}},\ \lambda_2^{(r)} = \frac{N_{ср\_конф}}{T_{уязв}^{(r)}},\ \lambda_3^{(r)} = \frac{1}{T_{нв\_сзи}^{(r)}}\quad , \qquad (3.26)$$

где r - номер ИНВ, R - общее число ИНВ, входящих в коалицию, $T_{по}^{(r)}$ - среднее время, требующееся r-му ИНВ для получения информации о ПО ИС, $T_{уязв}^{(r)}$ - среднее время, требующееся r-му ИНВ для получения информации о всех уязвимостях в ПО ИС, $T_{нв}^{(r)}$ - среднее время, требующееся r-му ИНВ для получения информации о способе использования уязвимости в ПО ИС для НВ на ИС.

Компьютерная имитационная модель с использованием механизма гибридных автоматов (карты Харела). Изменения в имитационной модели конфликта ИС без СЗИ и коалиции ИНВ без инсайдера аналогичны изменениям в объектно-ориентированной модели. Добавляются несколько дополнительных блоков *«INV»* (в зависимости от количества ИНВ), каждый из которых соответствует конкретному ИНВ (рис. 3.24).В каждом блоке, отвечающем за действия конкретного ИНВ, добавляются переходы, связанные с обменом информацией между ИНВ (с возникновением событий *«inf_po»* (*«ИНВ нашел информацию о ПО ИС»*), *«inf_uyazv»*, *«nv»*). В каждом блоке *«INV»* при входе в состояние *«Informaciya_o_PO»* генерируется событие *«inf_po»*. В блоки *«Sysadmin»* и *«IS»* изменения не вносятся [39]. SF-модель конфликта ИС без СЗИ и коалиции ИНВ изображена на рисунке 3.25.

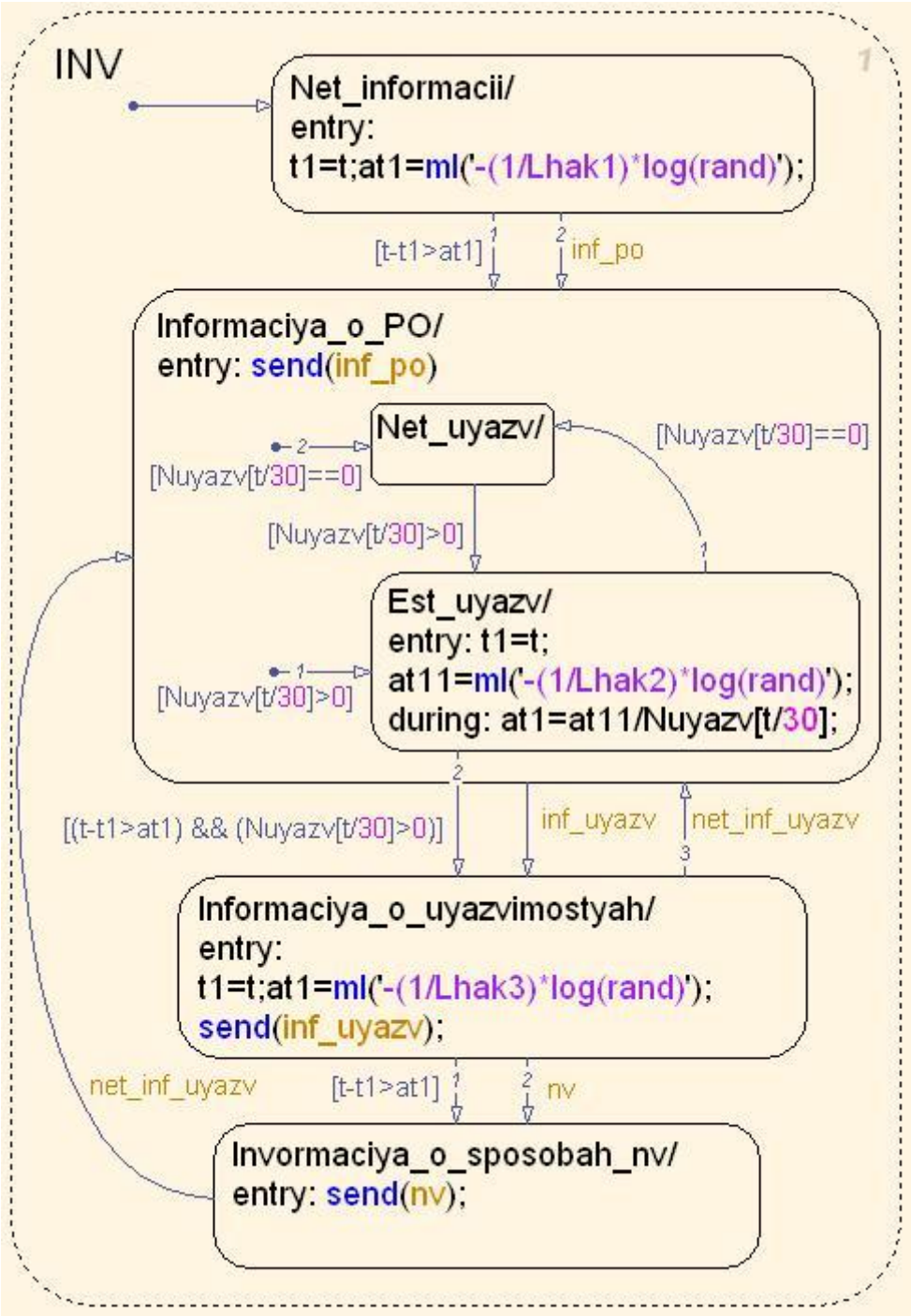

Рисунок 3.24 – Блок, отражающий действия одного ИНВ (*«INV»*), имитационной модели конфликта ИС без СЗИ и коалиции ИНВ без инсайдера

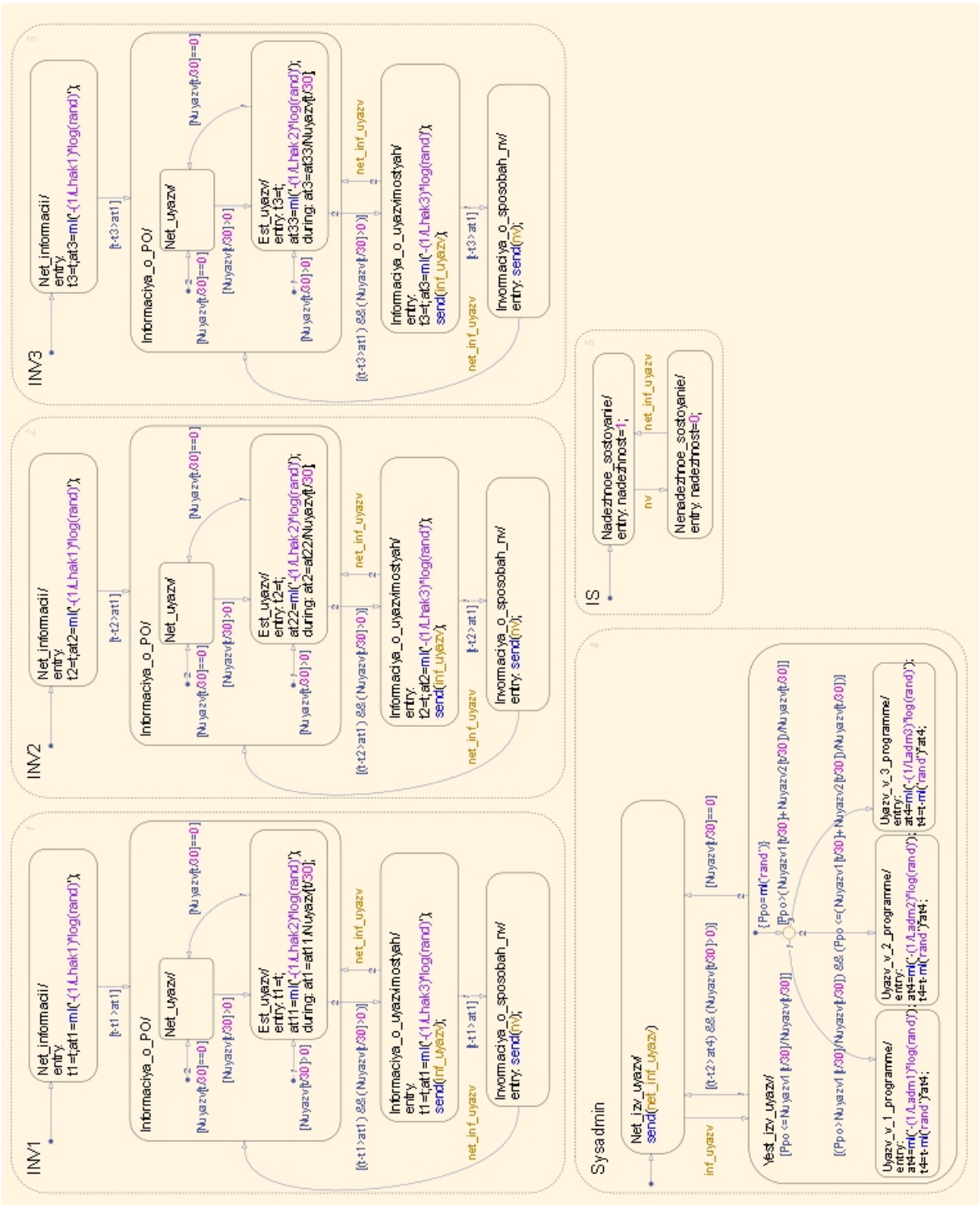

Рисунок 3.25 – SF-модель конфликта ИС без СЗИ и коалиции ИНВ

Сравнение результатов моделирования. Чтобы сравнить имитационную и математическую модели, предлагается рассчитать вероятность надежности ИС и вероятность нахождения ИС в надежном состоянии для каждого полугодия в течение 11 лет, начиная с октября 2001 года (время выпуска операционной системы Windows XP), при условии, что в ИС установлена только операционная система Windows XP. Необходимые статистические данные по ПО берутся из [51,77,79]. Коэффициент работы системного администратора для определенности берется $k = 1$. Шаг дискретизации берется равным 0,01 дня. Необходимое количество испытаний для имитационной модели, в соответствии с [87], выбирается равным $N\_isp = 1000$. Расчет предлагается осуществить для 4-х разных коалиций ИНВ:

- 3 источника негативных воздействий 1-й категории;
- 3 источника негативных воздействий 2-й категории;
- 3 источника негативных воздействий 3-й категории;
- 3 источника негативных воздействий 4-й категории.

Ниже приведен график вероятности нахождения в надежном состоянии (рис 3.26) ИС с операционной системой Windows XP при попытке НВ на нее коалиций, состоящих из 3-х ИНВ 1-й категории, из 3-х ИНВ 2-й категории, из 3-х ИНВ 3-й категории и из 3-х ИНВ 4-й категории.

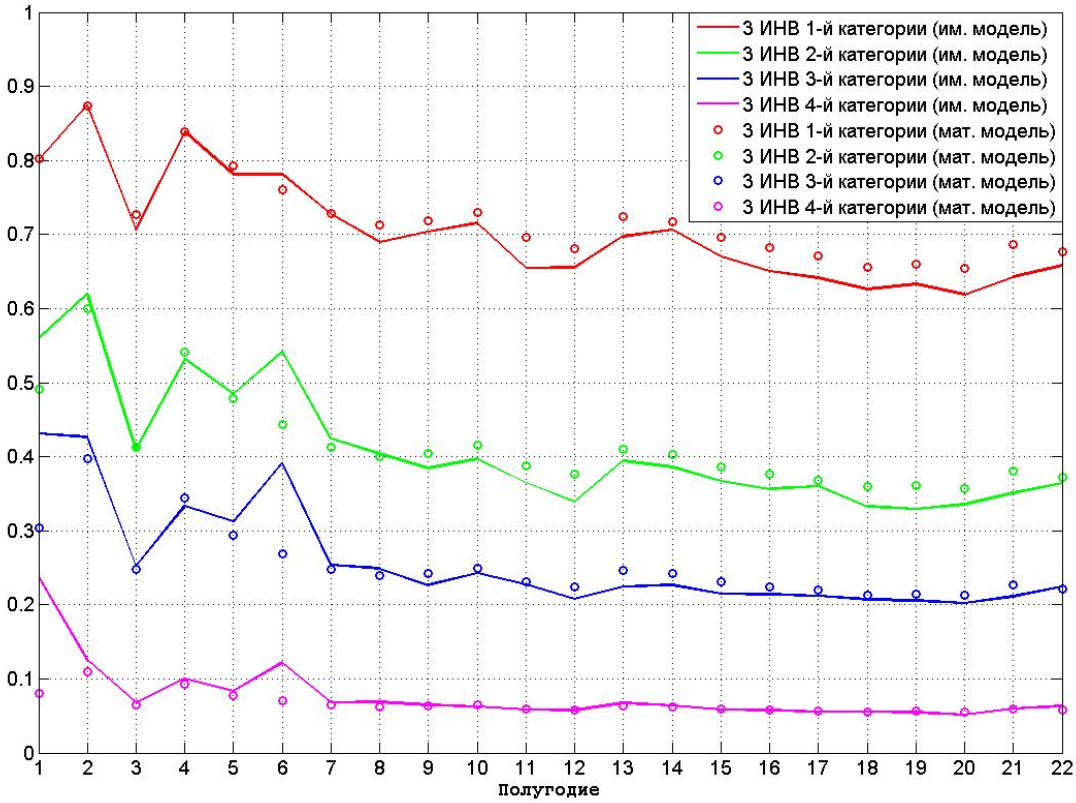

Рисунок 3.26 – Вероятность нахождения в надежном состоянии ИС с операционной системой Windows XP при попытке НВ на нее коалиций, состоящих из 3-х ИНВ 1-й категории, из 3-х ИНВ 2-й категории, из 3-х ИНВ 3-й категории и из 3-х ИНВ 4-й категории, и коэффициенте работы системного администратора $k = 1$

Максимальное среднее абсолютное отклонение вероятности надежности ИС, рассчитанной при помощи математической модели, от вероятности надежности ИС, рассчитанной при помощи имитационной модели, составило 7%, а максимальное среднее абсолютное отклонение вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи математической модели, от вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи имитационной модели – 2%. С учетом того, что время конфликта предполагается равным полгода (180 дней), последний результат означает, что разница между средним временем нахождения ИС в надежном состоянии, рассчитанным при помощи математической и имитационной моделей, равна приблизительно 4 дням, что весьма существенно при условии, если каждый день нахождения ИС в

ненадежном состоянии несет большие материальные убытки компании, владеющей этой ИС.

3.4 Модели функционирования информационной системы без средств защиты информации в условиях конфликтного взаимодействия с коалицией внешних источников негативных воздействий с инсайдером

Пусть имеется ИС с установленным ПО. На ИС негативно воздействует коалиция ИНВ, состоящая из R ИНВ и одного инсайдера. Как и в модели конфликта ИС без СЗИ и коалиции ИНВ без инсайдера, ИНВ последовательно добывают информацию о ПО, установленном в ИС, об уязвимости в этом ПО и о способах использования этой уязвимости для НВ на ИС и обмениваются ей. Инсайдер в свою очередь последовательно добывает информацию о ПО, установленном в ИС, и об уязвимости в этом ПО, сообщая ее ИНВ. Его отличие от ИНВ также заключается в том, что он может осуществить поиск соответствующей информации гораздо быстрее, чем ИНВ, поскольку в отличие от ИНВ инсайдер имеет прямой доступ к ИС [39].

Объектно-ориентированная модель. Объектно-ориентированная модель конфликта ИС без СЗИ и коалиции ИНВ с инсайдером аналогична модели конфликта ИС без СЗИ и коалиции ИНВ без инсайдера, но включает также диаграмму состояний инсайдера (рис. 3.27), которая отличается от диаграмм состояний ИНВ тем, что в ней в соответствии с постановкой задачи отсутствует состояние *«Информация о способе использования уязвимости в ПО ИС для НВ на ИС»*.

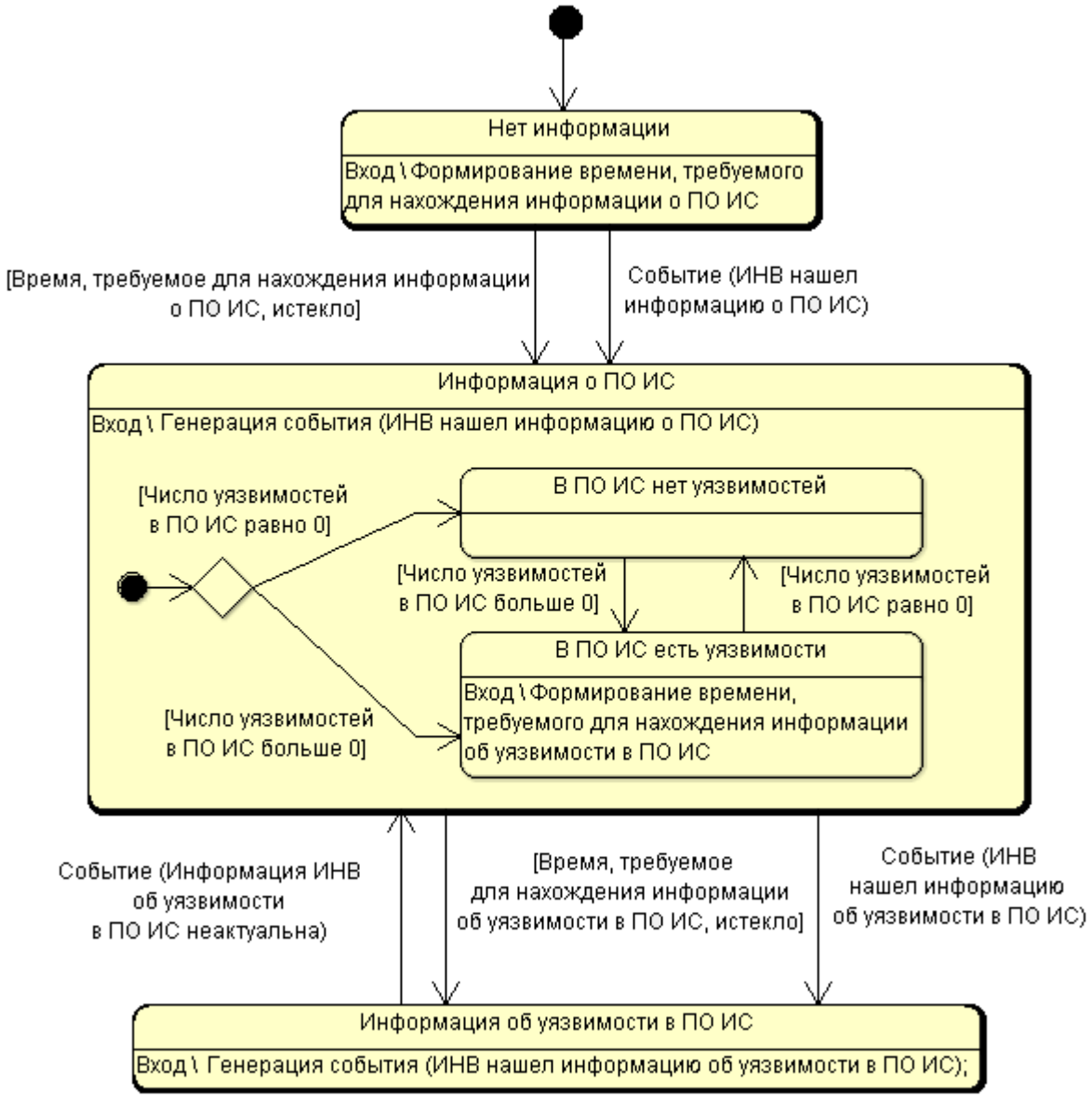

Рисунок 3.27 – Диаграмма состояний инсайдера в ходе конфликтного взаимодействия ИС без СЗИ с коалицией ИНВ с инсайдером

Математическая модель. Математическая модель конфликта ИС без СЗИ и коалиции ИНВ с инсайдером аналогична модели конфликта ИС без СЗИ и коалиции ИНВ без инсайдера, но интенсивности переходов λ_1, λ_2, описывающих последовательное добывание информации о ПО ИС, об уязвимости в ПО ИС, включают в себя также соответствующие интенсивности инсайдера ($\lambda_1^{(инс)}, \lambda_2^{(инс)}$)

$$\lambda_1 = \sum_r^R \lambda_1^{(r)} + \lambda_1^{(инс)},\ \lambda_2 = \sum_r^R \lambda_2^{(r)} + \lambda_2^{(инс)},$$

$$\lambda_1^{(инс)} = \frac{1}{T_{по}^{(инс)}}, \qquad \lambda_2^{(инс)} = \frac{N_{ср\_конф}}{T_{уязв}^{(инс)}}, \tag{3.27}$$

где $T_{по}^{(инс)}$ - среднее время, требующееся инсайдеру для получения информации о ПО ИС, $T_{уязв}^{(инс)}$ - среднее время, требующееся инсайдеру для получения информации обо всех уязвимостях в ПО ИС.

Компьютерная имитационная модель с использованием механизма гибридных автоматов (карты Харела) Изменения в имитационной модели конфликта ИС без СЗИ и коалиции ИНВ с инсайдером аналогичны изменениям в объектно-ориентированной модели. А именно добавляется блок *«INS»* (рис. 3.28) аналогичный блоку *«INV»*, но не имеющий состояния *«Invormaciya_o_sposobah_nv»* [39].

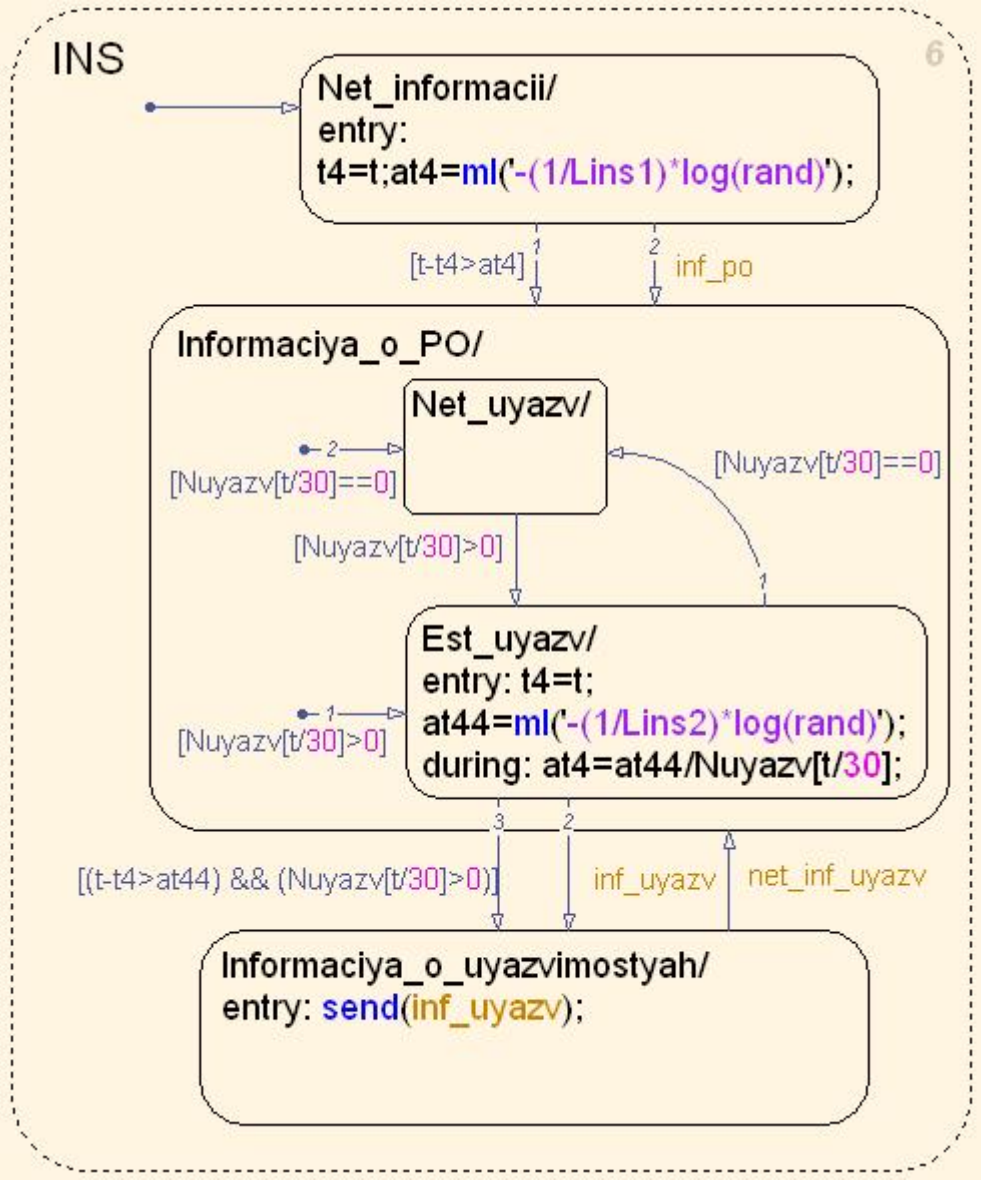

Рисунок 3.28 – Блок, отражающий действия инсайдера (*«INS»*), имитационной модели конфликта ИС без СЗИ и коалиции ИНВ с инсайдером

Сравнение результатов моделирования. Чтобы сравнить имитационную и математическую модели, предлагается рассчитать вероятность надежности ИС и вероятность нахождения ИС в надежном состоянии для каждого полугодия в течение 11 лет, начиная с октября 2001 года (время выпуска операционной системы Windows XP), при условии, что в ИС установлена только операционная система Windows XP. Необходимые статистические данные по ПО берутся из [51,77,79]. Коэффициент работы системного администратора для определенности берется $k = 1$. Шаг дискретизации берется равным 0,01 дня. Необходимое

количество испытаний для имитационной модели, в соответствии с [87], выбирается равным $N\_isp = 1000$.

Расчет предлагается осуществить для 4-х разных коалиций ИНВ:

- 3 источника негативных воздействий 1-й категории и инсайдер;
- 3 источника негативных воздействий 2-й категории и инсайдер;
- 3 источника негативных воздействий 3-й категории и инсайдер;
- 3 источника негативных воздействий 4-й категории и инсайдер.

Параметры инсайдера берутся следующими: $T_{\text{по}} = 1/24$ дня, $T_{\text{уязв}} = 3/24$ дня.

Ниже приведен график вероятности нахождения в надежном состоянии (рис 3.29) ИС с операционной системой Windows XP при попытке НВ на нее коалиций, состоящих из 3-х ИНВ 1-й категории и инсайдера, из 3-х ИНВ 2-й категории и инсайдера, из 3-х ИНВ 3-й категории и инсайдера, и из 3-х ИНВ 4-й категории и инсайдера.

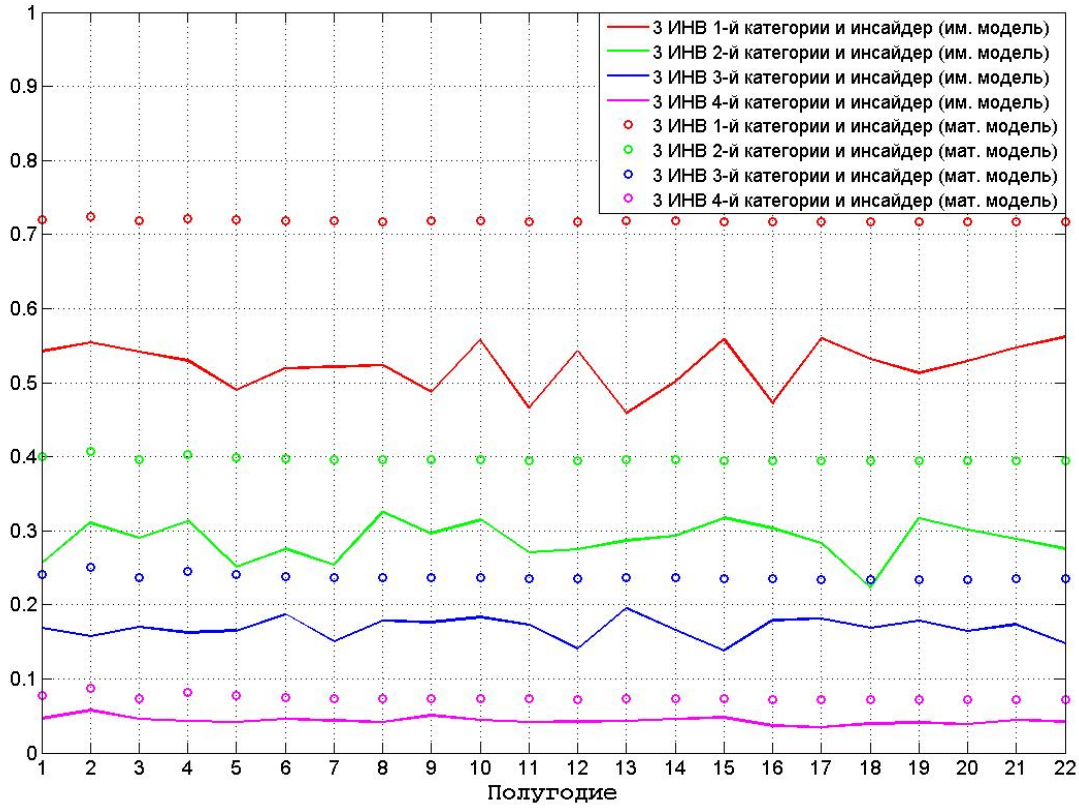

Рисунок 3.29 – Вероятность нахождения в надежном состоянии ИС с операционной системой Windows XP при попытке НВ на нее коалиций, состоящих из 3-х ИНВ 1-й категории и инсайдера, из 3-х ИНВ 2-й категории и инсайдера, из 3-х ИНВ 3-й категории и инсайдера, и из 3-х ИНВ 4-й категории и инсайдера, и коэффициенте работы системного администратора $k = 1$

Максимальное среднее абсолютное отклонение вероятности надежности ИС, рассчитанной при помощи математической модели, от вероятности надежности ИС, рассчитанной при помощи имитационной модели, составило 7%, а максимальное среднее абсолютное отклонение вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи математической модели, от вероятности нахождения ИС в надежном состоянии, рассчитанной при помощи имитационной модели – 20%. С учетом того, что время конфликта предполагается равным полгода (180 дней), последний результат означает, что разница между средним временем нахождения ИС в надежном состоянии, рассчитанным при помощи математической и имитационной моделей, равна приблизительно 36 дням, что крайне существенно, следовательно, при исследовании надежности ИС,

которая может подвергнуться негативным воздействиям со стороны ИНВ, которым помогает инсайдер внутри компании, владеющей этой ИС, необходимо использовать имитационную модель, а не математическую.

Характеризуя данную и все остальные разработанные математические и компьютерные модели стоит отметить, что они легко могут быть усовершенствованы на основе более точного учета поведения ИНВ и системного администратора, устройства ИС, а так же параметров, влияющих на это поведение (например: учет наличия эксплойтов для уязвимостей, наличие в ИС систем обнаружения вторжений или средств обмана ИНВ и т.д.). Для таких изменений нет необходимости менять всю концепцию оценки надежности использования ПО в ИС, а достаточно добавить новые состояния и(или) переходы (а в имитационную модель также, возможно - новые блоки и подблоки). Принципиальная же разница между компьютерными имитационными и математическими моделями заключается в том, что первые в отличие от вторых:

- учитывают зависимость среднестатистического числа уязвимостей в ИС от времени (в математических моделях используется значение данной величины, усредненное по времени конфликта);
- допускают произвольный характер переходов между состояниями сторон (при использовании математической модели вероятности переходов между состояниями имеют показательный закон распределения);
- позволяют рассматривать конфликтные ситуации с любыми вариантами отношений между ИНВ, оказывающими преднамеренное НВ на ИС (различные варианты коалиций ИНВ и вариант отсутствия коалиций между ИНВ);
- позволяют рассчитать дополнительные величины, характеризующие надежность ИС, такие как количество возможных успешных НВ на ИС и среднее максимальное время постоянного нахождения ИС в ненадежном состоянии (без возвращения в надежное состояние).

3.5. Общий алгоритм анализа вероятностных характеристик надежности использования программного обеспечения информационной системы в условиях преднамеренных негативных воздействий

Алгоритм прогноза вероятности надежности ИС без учета характера НВ, предложенный во 2-й главе, может быть усовершенствован для случая ПНВ при помощи разработанных математических и имитационных моделей. Применение данных моделей позволяет учесть параметры ИНВ, а также определить кроме вероятности надежности ИС вероятность нахождения ИС в надежном состоянии и ряд других важных величин, например, вероятность нарушения надежности ИС при помощи уязвимостей в конкретном ПО, время, необходимое ИНВ для успешного НВ на ИС, количество успешных НВ на ИС, среднее максимальное время постоянного нахождения ИНВ в ненадежном состоянии (без возвращения в надежное состояние). Для реализации этой задачи предлагается следующий алгоритм, общая схема которого приведена на рис. 3.30.

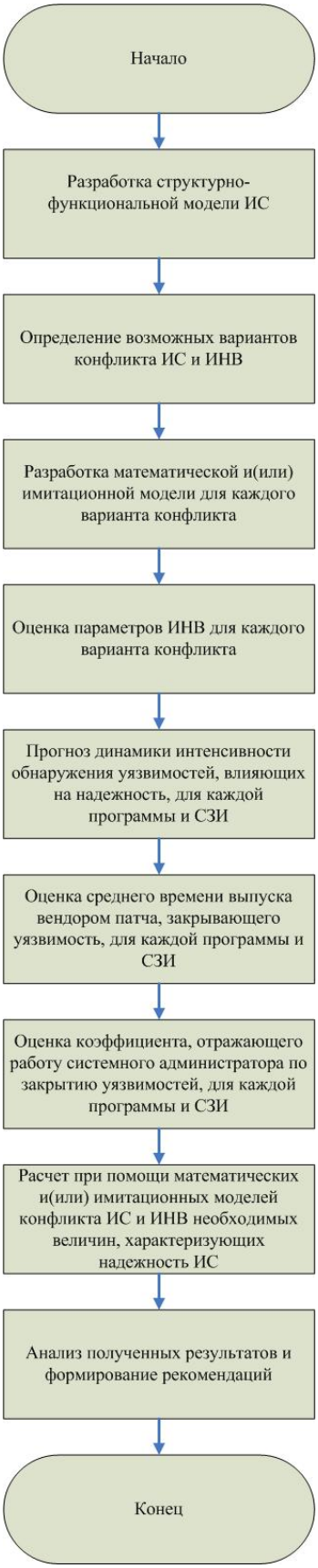

Рисунок 3.30 – Блок-схема общего алгоритма анализа вероятностных характеристик надежности использования ПО в ИС в условиях внутренних уязвимостей и ПНВ

При реализации алгоритма выполняются следующие этапы анализа системы:

1. Разработка структурно-функциональной модели ИС: определение ПО, установленного в ИС, наличия СЗИ и их конфигурации (защищают от внешнего и/или внутреннего ИНВ).

2. Определение возможных вариантов конфликта ИС и ИНВ (конфликт ИС без СЗИ с одним ИНВ, конфликт ИС с СЗИ с одним ИНВ, конфликт ИС без СЗИ с коалицией ИНВ без инсайдера, конфликт ИС без СЗИ с коалицией ИНВ с инсайдером).

3. Разработка математической и(или) имитационной модели для каждого варианта конфликта.

4. Оценка параметров ИНВ для каждого варианта конфликта.

5. Прогноз интенсивности обнаружения уязвимостей, влияющих на надежность, для каждой программы и СЗИ

6. Оценка среднего времени выпуска вендором патча, закрывающего уязвимость, для каждой программы и СЗИ.

7. Оценка коэффициента, отражающего работу системного администратора по закрытию уязвимостей, для каждой программы и СЗИ.

8. Расчет среднего числа уязвимостей в каждой программе и СЗИ ИС в течение периода прогноза, исходя из разработанной модели ИС.

9. Расчет при помощи математических и(или) имитационных моделей конфликта ИС и ИНВ необходимых величин, характеризующих надежность ИС.

10. Оценка полученных результатов и формирование рекомендаций.

Таким образом полученный алгоритм в отличие от других динамических подходов к оценке надежности ИС [29-31] позволяет одновременно использовать прогноз в отношении интенсивности обнаружения уязвимостей в ПО, учесть работу вендоров и системных администраторов, устройство ИС (наличие различного ПО и СЗИ), параметры ИНВ, негативно воздействующих на ИС, различные варианты конфликтов ИС и ИНВ (обмен информацией в коалиции ИНВ, наличие инсайдера и т.п.), а также саму специфику конфликтных

взаимодействий, когда упреждающие действия с одной стороны могут привести к невозможности другой стороной осуществить поставленные цели. Также в отличие от многих других динамических подходов [30,31] данный алгоритм при оценке и прогнозе ряда параметров использует общедоступную статистику, опубликованную в сети Интернет (например [51,77,79]), сбор которой может быть автоматизирован. Кроме того, данный алгоритм позволяет рассчитать не 1 или 2, а множество величин, характеризующих надежность ИС.

Выводы по главе

1. Разработаны математические и имитационные модели конфликтов ИС и ИНВ, позволяющие в отличие от моделей [30,31] учитывать: различный состав и структуру построения ИС (наличие различного ПО, наличие СЗИ и т.п.); динамику уязвимостей в ИС и влияние на надежность ИС работы системного администратора и вендоров; различные варианты конфликтов ИС и ИНВ, основанные на реальных ситуациях (например: использование реальных этапов НВ, возможность ИНВ вступать в коалиции и находить инсайдеров).

2. Принципиальная разница между компьютерными имитационными и математическими моделями конфликтов ИС и ИНВ заключается в том, что первые в отличие от вторых учитывают зависимость среднестатистического числа уязвимостей в ИС от времени (в математических моделях используется значение данной величины, усредненное по времени конфликта); допускают произвольный характер переходов между состояниями сторон (при использовании математической модели вероятности переходов между состояниями имеют показательный закон распределения); позволяют рассматривать конфликтные ситуации с любыми вариантами отношений между ИНВ, оказывающими преднамеренное НВ на ИС (различные варианты коалиций ИНВ и вариант отсутствия коалиций между ИНВ); позволяют рассчитать дополнительные величины, характеризующие надежность ИС, такие как количество возможных успешных НВ на ИС и среднее максимальное время постоянного нахождения ИС в ненадежном состоянии (без возвращения в надежное состояние).

3. Разработанные математические и имитационные модели конфликтов ИС и ИНВ могут быть легко усовершенствованы путем добавления новых состояний и переходов (а так же для имитационных моделей - блоков и подблоков).

4. Выбор между математической и имитационной моделью обусловлен необходимой точностью оценки параметров надежности ИС.

5. Разработан алгоритм прогноза вероятностных характеристик надежности ИС в условиях внутренних уязвимостей и ПНВ, позволяющий, в отличие от известных [29-31], одновременно: использовать прогноз в отношении интенсивности обнаружения уязвимостей; учесть работу вендоров; учесть работу системных администраторов; учесть устройство ИС (наличие различного ПО, наличие СЗИ и т.п.); учесть параметры ИНВ; учесть различные варианты конфликтов ИС и ИНВ, основанные на реальных ситуациях (например: использование реальных этапов ПНВ, учет возможностей ИНВ вступать в коалиции и находить инсайдеров); использовать доступные источники данных для оценки параметров, влияющих на надежность ИС; рассчитать ряд величин, характеризующих надежность ИС, таких как вероятность надежности ИС, вероятность нахождения ИС в надежном состоянии, вероятность нарушения надежности ИС при помощи уязвимостей в конкретном ПО, время, необходимое ИНВ для успешного НВ на ИС, количество возможных успешных НВ на ИС, среднее максимальное время постоянного нахождения ИС в ненадежном состоянии (без возвращения в надежное состояние).

Глава 4. Оценка надежности использования программного обеспечения в информационных системах удостоверяющих центров и их пользователей

4.1. Общая структура типовых информационных систем удостоверяющих центров и их пользователей

Оценки вероятностных характеристик надежности использования программного обеспечения предлагается рассмотреть на примере типовых информационных систем, использующихся удостоверяющими центрами и их пользователями.

Удостоверяющий центр (УЦ) - юридическое лицо или индивидуальный предприниматель, осуществляющие функции по созданию и выдаче сертификатов ключей проверки электронных подписей, а также иные функции, предусмотренные федеральным законом № 63-ФЗ [89]. Сертификат ключа проверки электронной подписи - электронный документ или документ на бумажном носителе, выданные УЦ либо доверенным УЦ и подтверждающие принадлежность ключа проверки электронной подписи владельцу сертификата ключа проверки электронной подписи (пользователю УЦ) [89].

Электронная подпись используется в электронном документообороте между различными организациями, юридическими и физическими лицами. В частности в РФ электронная подпись уже применяется в электронном документообороте между кредитными организациями и кредитными бюро, между налоговыми органами и налогоплательщиками, в области государственного управления [90].

Далее будет рассмотрен типовой УЦ, использующий наиболее распространённую в нашей стране технологию КриптоПро – КриптоПро УЦ версии 1.5 [91]. Типовой УЦ включает в себя следующие компоненты [91], которые можно рассматривать как отдельные ИС:

- Центр сертификации;

- Центр регистрации;
- Автоматизированное рабочее место (АРМ) администратора;
- Межсетевой экран;
- Сервер публикации отозванных сертификатов для пользователей УЦ.

Кроме того, через сеть Интернет с УЦ связаны ИС пользователей УЦ, Которые должны иметь возможность в режиме онлайн проверять списки отозванных сертификатов на сервере публикации, а в ряде случаев (в зависимости от принятого регламента УЦ) также иметь доступ к сервису центра регистрации.

Общая структурная схема УЦ вместе с ИС пользователей УЦ изображена на рисунке 4.1 (изображена ИС 1-го пользователя УЦ).

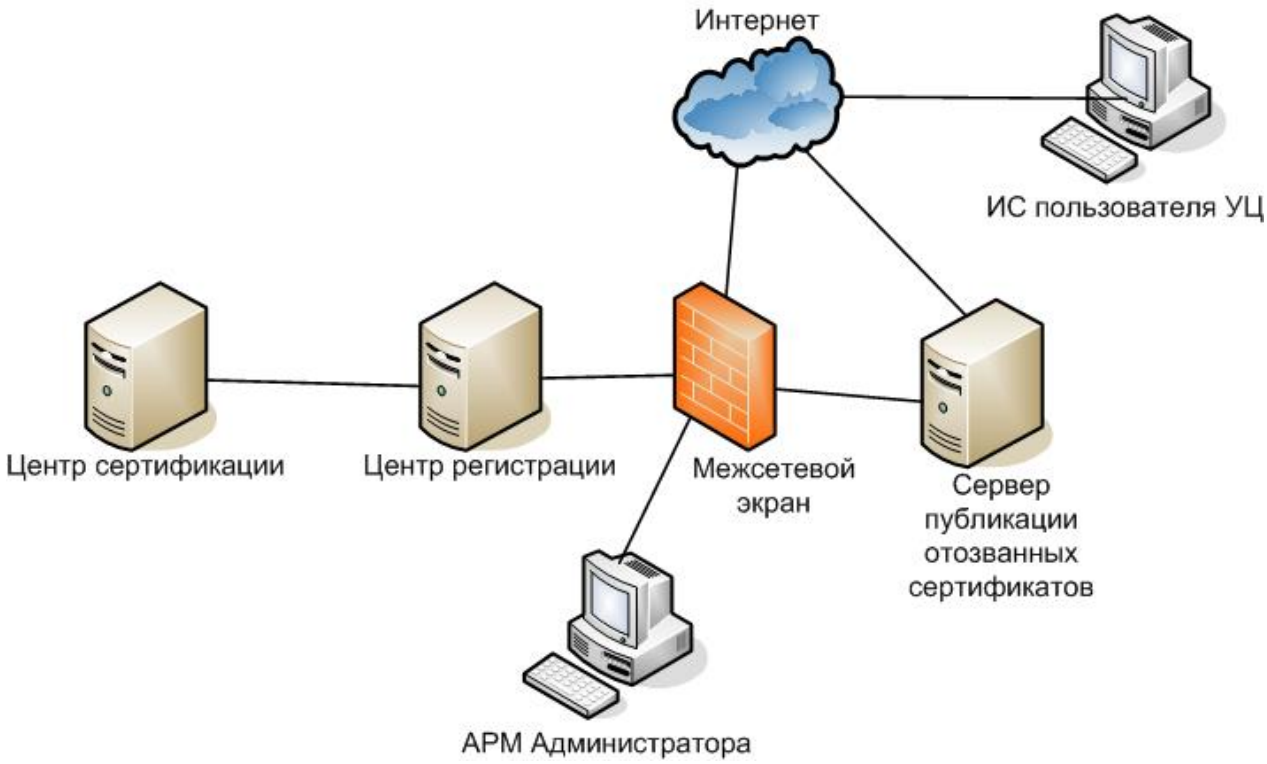

Рисунок 4.1 – Общая структурная схема УЦ вместе с ИС пользователей УЦ

В ходе исследований был осуществлен прогноз в отношении надежности 3-х базовых элементов, представленных на данной структурной схеме: ИС пользователя

УЦ, сервера публикации отозванных сертификатов и центра регистрации. Прогноз произведен для типовых вариантов данных элементов [91] на период от сентября 2013 года до февраля 2014 года включительно (рассматриваемые попытки НВ со стороны ИНВ изображены на рисунке 4.2).

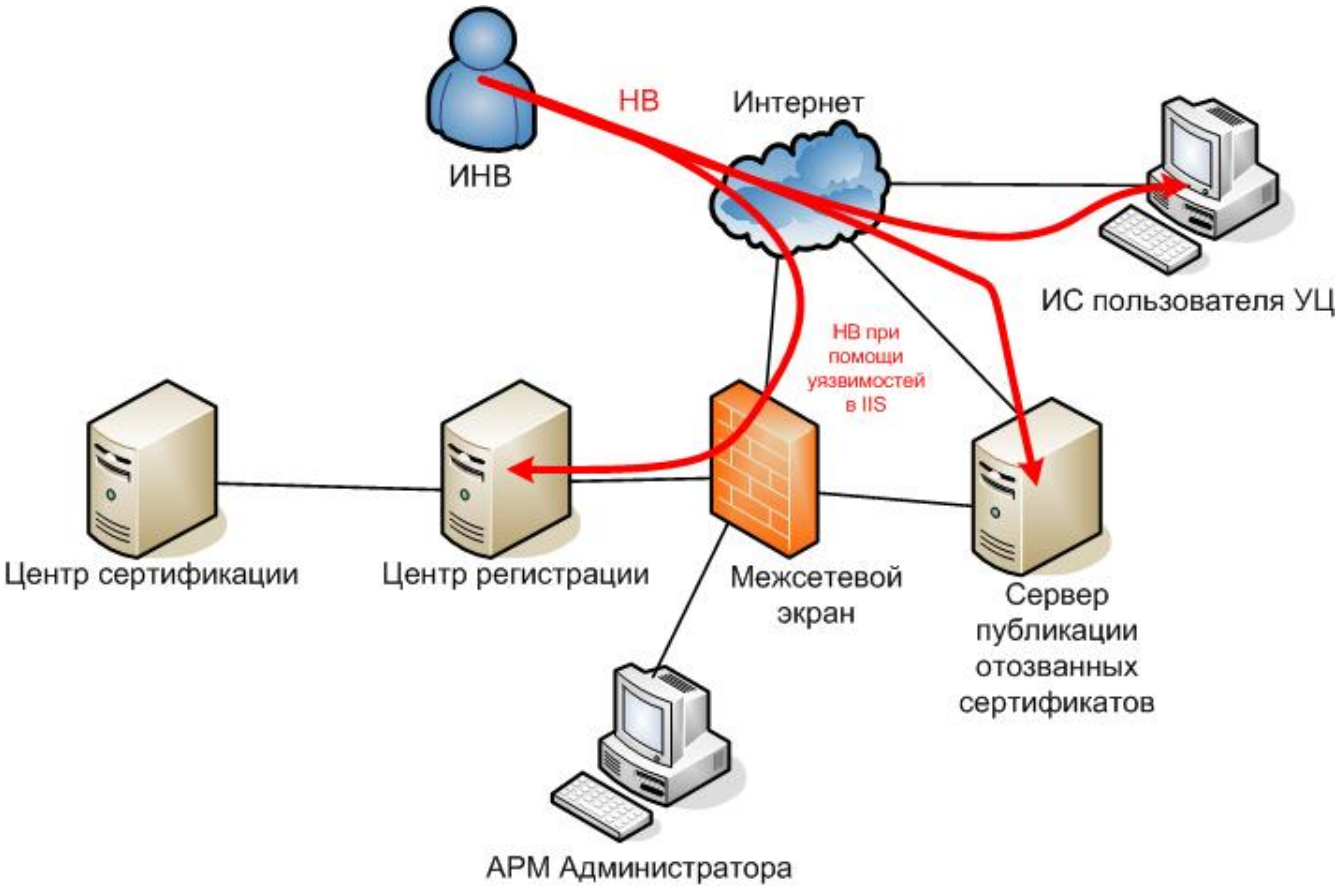

Рисунок 4.2 – Возможные попытки НВ со стороны ИНВ на структурные звенья ИС УЦ и его пользователей

4.2. Оценка надежности типовой информационной системы пользователя удостоверяющего центра

Предлагается рассмотреть типовую ИС пользователя УЦ, в которой установлено следующее ПО (здесь и далее рассматривается не все ПО, которое

может быть установлено в ИС, а только основное, имеющееся в большинстве рассматриваемых типовых элементов и содержащее в себе наибольшее количество уязвимостей):

- Windows XP SP3 (операционная система);
- Microsoft Office 2007;
- Adobe Reader X 10.0;

и не установлено СЗИ, препятствующих непосредственному НВ на ПО ИС.

ИС может быть описана простейшей математической моделью, по типу предложенных в п. 2.3, и представленной для данного конкретного случая на рисунке 4.3

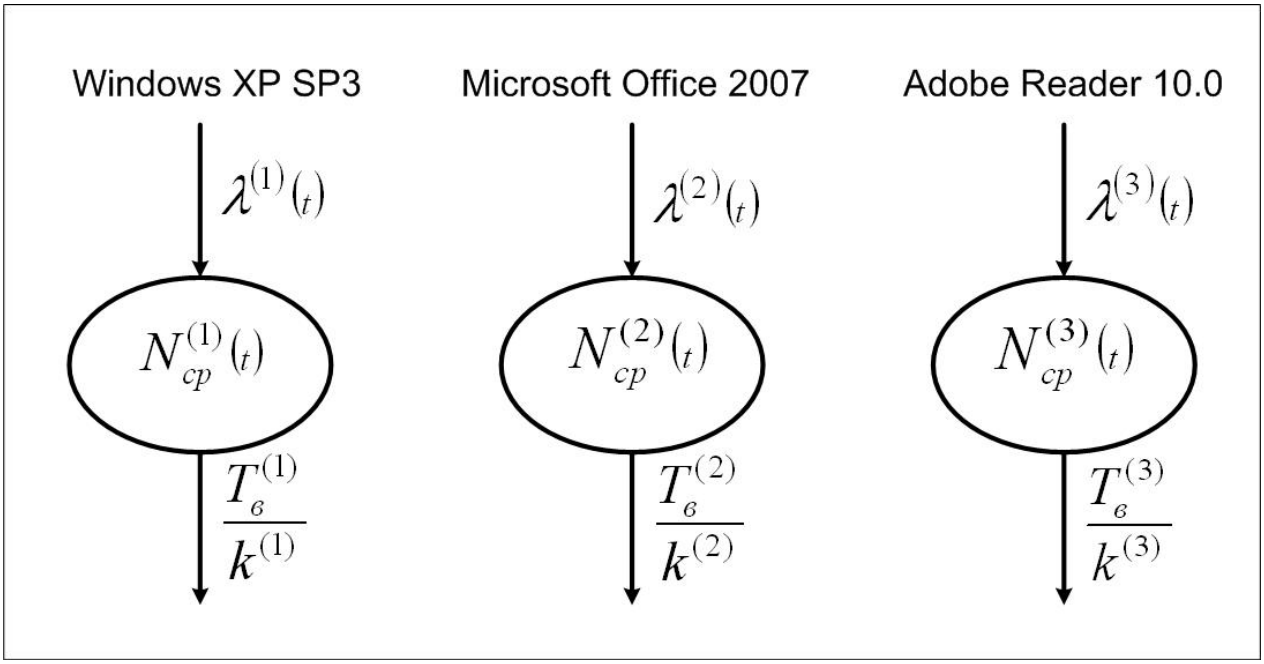

Рисунок 4.3 –Модель типовой ИС пользователя УЦ

Необходимые статистические данные для прогноза интенсивности обнаружения уязвимостей в каждом конкретном ПО были получены на основе данных [51]. Для прогноза интенсивности обнаружения уязвимостей использовался алгоритм, предложенный в п. 2.1. Данные прогноза представлены в таблице 4.1

Таблица 4.1 – Прогноз интенсивности обнаружения уязвимостей в ПО типовой ИС пользователя УЦ на период с сентября 2013 года до февраля 2014 года включительно

ПО / Месяц	Windows XP SP3	Microsoft Office 2007	Adobe Reader 10.0
сентябрь 2013	1,7	1,5	2,8
октябрь 2013	1,0	1,4	15,1
ноябрь 2013	1,3	1,3	19,6
декабрь 2013	2,4	1,2	13,3
январь 2014	4,3	1,0	8,8
февраль 2014	7,2	0,7	12,4

Исходя из статистики [51,77,79], среднее время устранения уязвимостей вендором из Windows XP SP3 равно 19,4 дня, из Microsoft Office 2007 – 54 дня, из Adobe Reader 10.0 – 8,8 дня. Поскольку точные данные по работе системных администраторов типовых ИС пользователей УЦ и ИНВ, которые могли бы негативно воздействовать на них, отсутствуют, предлагается осуществить прогноз для коэффициента работы системного администратора от 0 до 3 с шагом 0,1 и следующих ИНВ [81-85]:

- ИНВ 1-й категории ($T_{\text{по}} = 60$ дней, $T_{\text{уязв}} = 30$ дней, $T_{\text{нв}} = 30$ дней);
- ИНВ 2-й категории ($T_{\text{по}} = 20$ дней, $T_{\text{уязв}} = 10$ дней, $T_{\text{нв}} = 10$ дней);
- ИНВ 3-й категории ($T_{\text{по}} = 10$ дней, $T_{\text{уязв}} = 5$ дней, $T_{\text{нв}} = 5$ дней);
- ИНВ 4-й категории ($T_{\text{по}} = 5$ дней, $T_{\text{уязв}} = 1$ день, $T_{\text{нв}} = 1$ день).

Предполагается, что на ИС будет оказывать НВ только один ИНВ, кроме того, как уже было сказано, СЗИ, препятствующие непосредственному НВ на ПО в ИС, отсутствуют, следовательно, для моделирования конфликта ИС и ИНВ можно использовать имитационную модель, предложенную в п. 3.1.

Далее приведен прогноз вероятности надежности и вероятности нахождения в надежном состоянии типовой ИС пользователя УЦ (рис. 4.4-4.5).

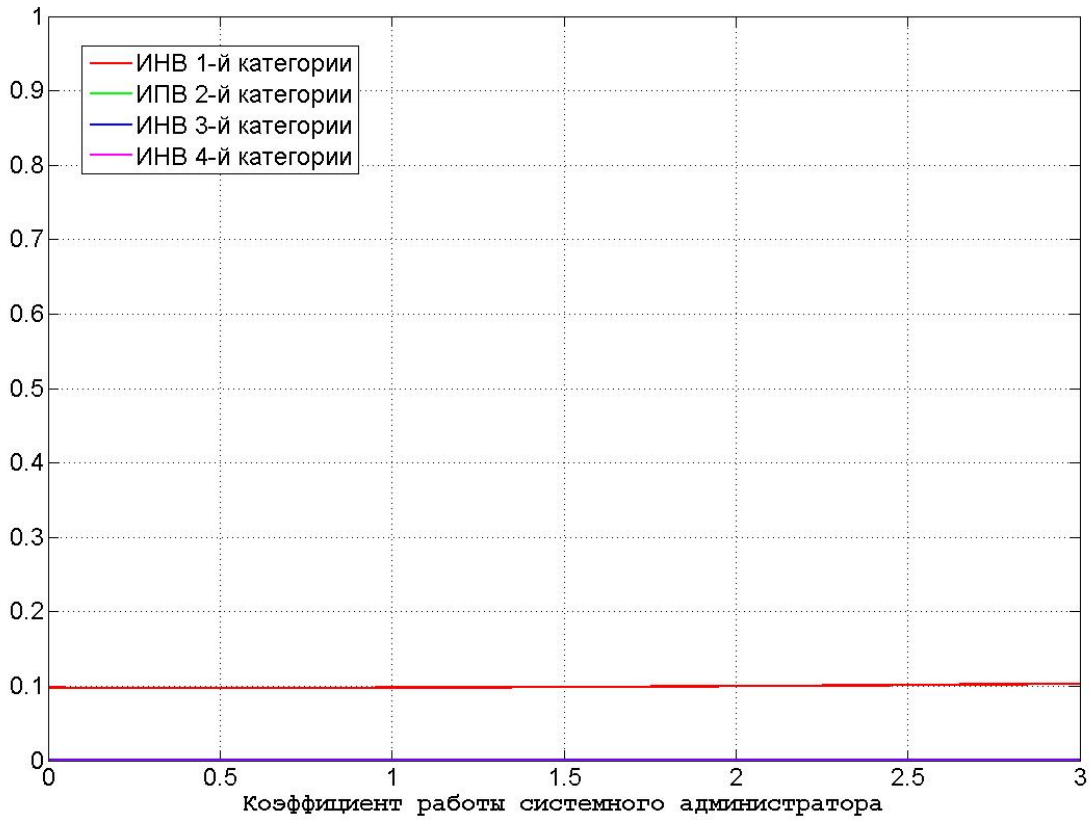

Рисунок 4.4 – Прогноз вероятности надежности типовой ИС пользователя УЦ на период с сентября 2013 года по февраль 2014 года включительно

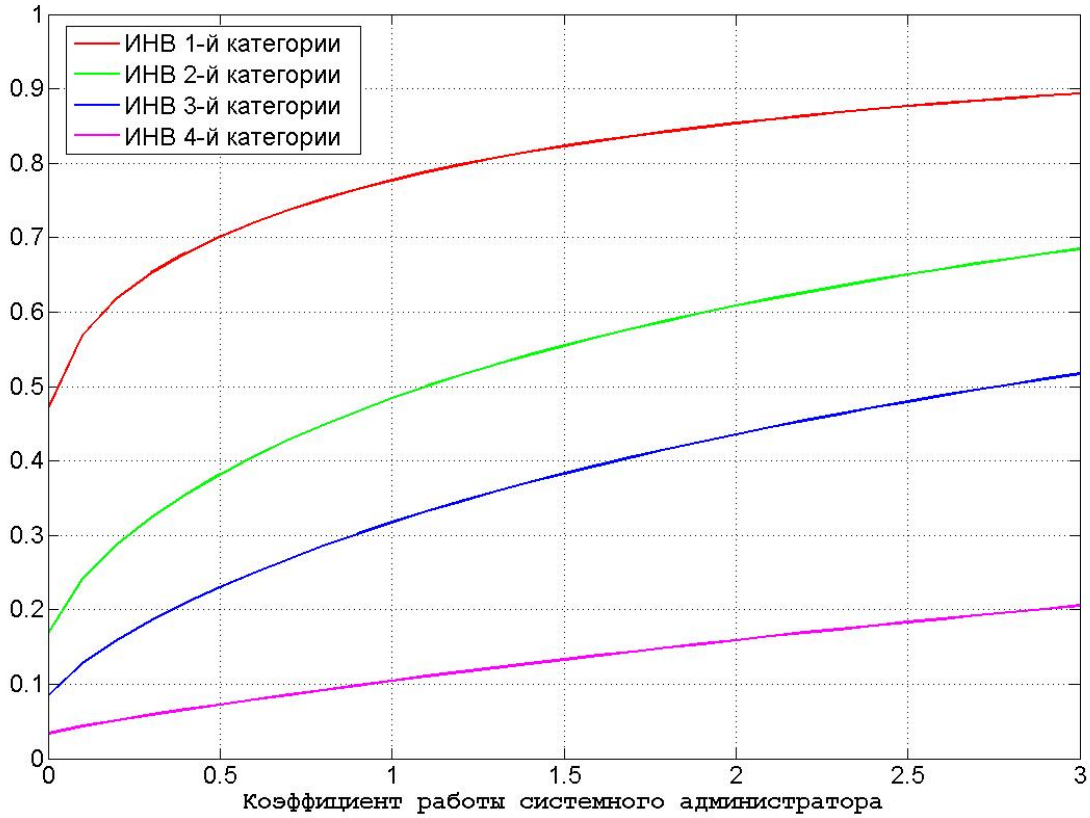

Рисунок 4.5 – Прогноз вероятности нахождения типовой ИС пользователя УЦ в надежном состоянии на период с сентября 2013 года по февраль 2014 года включительно

Анализируя графики 4.4 и 4.5, можно сделать следующие выводы.

1. Если на типовую ИС пользователя УЦ будет пытаться осуществить НВ ИНВ 1-й категории, то в 90% случаев он сможет нарушить правильную работу ИС. Если же на ИС будут негативно воздействовать ИНВ более высокой категории, то правильная работа ИС будет нарушена практически в 100% случаев.

2. Если системный администратор типовой ИС пользователя УЦ совсем не обновляет ПО, установленное в ней (коэффициент работы системного администратора равен 0), то даже при попытках НВ со стороны ИНВ 1-й категории в работа ИС будет нарушена в среднем в течение половины периода прогноза (приблизительно 90 дней), в остальных случаях это время будет больше.

3. Если в типовой ИС пользователя УЦ настроено автоматическое обновление ПО (коэффициент работы системного администратора равен 1), то даже при попытках НВ со стороны ИНВ 1-й категории работа ИС будет нарушена в среднем в течение 1/5 периода прогноза (приблизительно 36 дней), в остальных случаях это время будет больше.

4. Если контролем надежности типовой ИС пользователя УЦ занимается служба безопасности (крайне редкий случай), в которую входит большое число высококвалифицированных сотрудников, и применяются дополнительные организационные и технические меры по устранению уязвимостей из ПО (коэффициент работы системного администратора равен 3), то даже при попытках НВ со стороны ИНВ 1-й категории работа ИС будет нарушена в среднем в течение 1/10 периода прогноза (приблизительно 18 дней), в остальных случаях это время будет больше.

Таким образом, типовая ИС пользователя является ненадежной, и требуются дополнительные меры по обеспечению её защиты. Для повышения надежности рассматриваемой ИС рекомендуется ограничить ПО ИС только доверенным и реализующим требуемый функционал (в том числе ограничить сетевые протоколы и сетевые взаимодействия только необходимыми доверенными узлами), что обеспечивается соответствующими настройками операционной системы и установить СЗИ, интенсивность обнаружения уязвимостей в котором мала и которое препятствует непосредственному НВ на ПО ИС, в противном случае никаких гарантий работоспособности ИС в течение периода прогноза дать нельзя.

4.3. Оценка надежности типового сервера публикации отозванных сертификатов

Предлагается рассмотреть типовой сервер публикации отозванных сертификатов [92], в котором установлено следующее ПО:

- Ubuntu Linux 11.10 (операционная система);

- WEB-сервер Apache 2.2.16;

- Система управления WEB-контентом Typo3 4.5.6;

и не установлено СЗИ, препятствующих непосредственному НВ на ПО ИС (используются встроенные в это ПО средства защиты с настройками «по умолчанию»).

Данная ИС может быть описана простейшей математической моделью(данная модель не рассматривает негативные воздействия, связанные с надежностью TCP/IP протоколов сети Интернет и уязвимостями программ, их реализующих, соответственно, для их учета требуются дополнительные исследования), предложенной в п. 2.3 (рисунок 4.6).

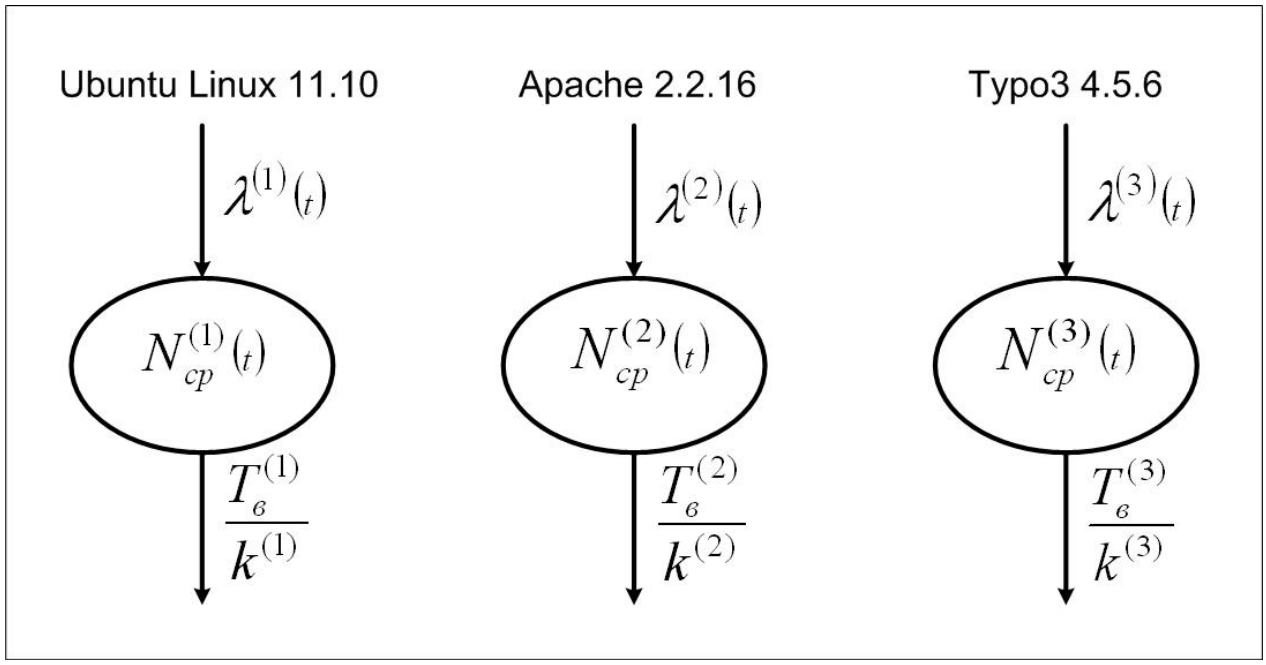

Рисунок 4.6 – Модель типового сервера публикации отозванных сертификатов

Необходимые статистические данные для прогноза интенсивности обнаружения уязвимостей в каждом конкретном ПО были выбраны из [51]. Для прогноза интенсивности обнаружения уязвимостей использовался алгоритм,

предложенный в п. 2.1. Данные прогноза представлены в таблице 4.2.

Таблица 4.2 – Прогноз интенсивности обнаружения уязвимостей в ПО типового сервера публикации отозванных сертификатов на период с сентября 2013 года до февраля 2014 года включительно

ПО / Месяц	Ubuntu Linux 11.10	Apache 2.2.16	Adobe Reader 10.0
сентябрь 2013	0,0	0,3	0,0
октябрь 2013	0,0	1,1	0,0
ноябрь 2013	3,6	2,2	0,7
декабрь 2013	7,7	2,0	2,1
январь 2014	6,7	1,5	2,2
февраль 2014	9,3	0,8	1,6

Исходя из статистики [51,73,75], среднее время устранения уязвимостей вендором из Ubuntu Linux 11.10 равно 6,6 дня, из Apache 2.2.16 – 1,5 дня, из Typo3 4.5.6 – 1,2 дня. Поскольку точные данные по работе системных администраторов типовых серверов публикации отозванных сертификатов и ИНВ, которые могли бы негативно воздействовать на них, отсутствуют, предлагается осуществить прогноз для коэффициента работы системного администратора от 0 до 3 с шагом 0,1 и следующих ИНВ [81-85]:

- ИНВ 1-й категории ($T_{\text{по}} = 60$ дней, $T_{\text{уязв}} = 30$ дней, $T_{\text{нв}} = 30$ дней);
- ИНВ 2-й категории ($T_{\text{по}} = 20$ дней, $T_{\text{уязв}} = 10$ дней, $T_{\text{нв}} = 10$ дней);
- ИНВ 3-й категории ($T_{\text{по}} = 10$ дней, $T_{\text{уязв}} = 5$ дней, $T_{\text{нв}} = 5$ дней);
- ИНВ 4-й категории ($T_{\text{по}} = 5$ дней, $T_{\text{уязв}} = 1$ день, $T_{\text{нв}} = 1$ день).

Предполагается, что на ИС будет оказывать НВ только один ИНВ, кроме того, как уже было сказано, СЗИ, препятствующие непосредственному НВ на ПО в ИС,

отсутствуют, следовательно, для моделирования конфликта ИС и ИНВ можно использовать имитационную модель, предложенную в п. 3.1.

Далее приведен прогноз вероятности надежности и вероятности нахождения в надежном состоянии типового сервера публикации отозванных сертификатов (рис. 4.7-4.8).

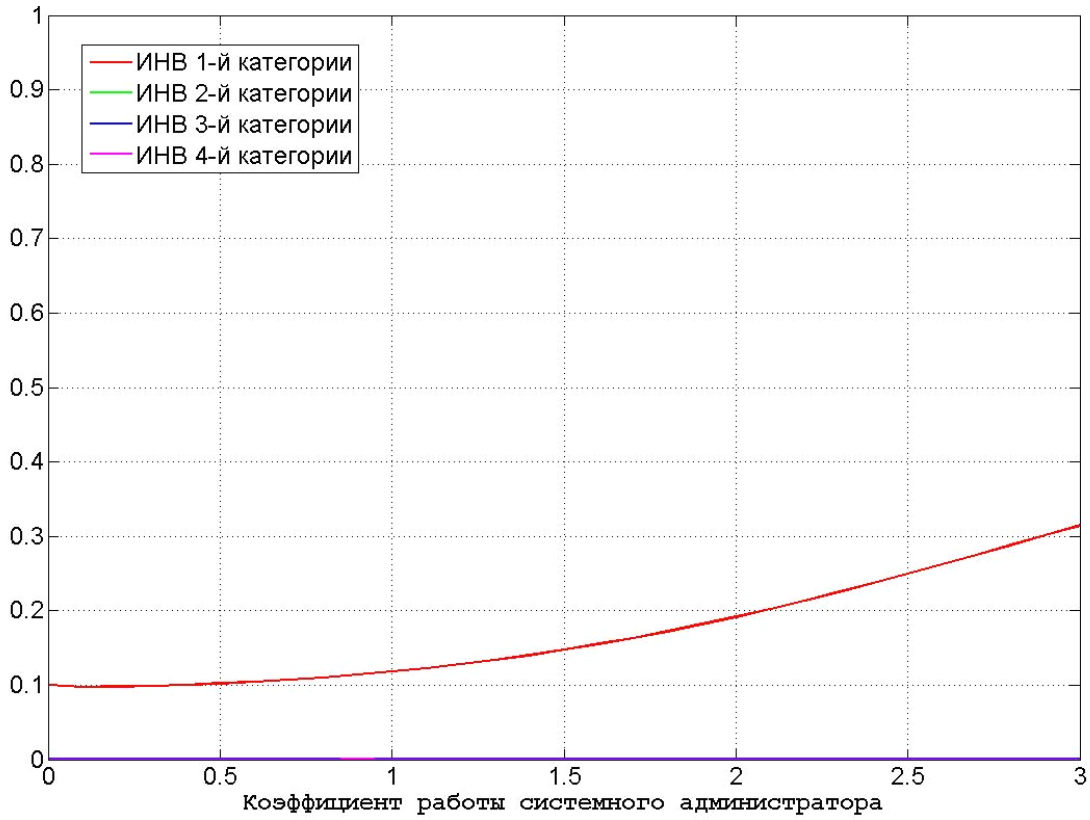

Рисунок 4.7 – Прогноз вероятности надежности типового сервера публикации отозванных сертификатов на период с сентября 2013 года по февраль 2014 года включительно

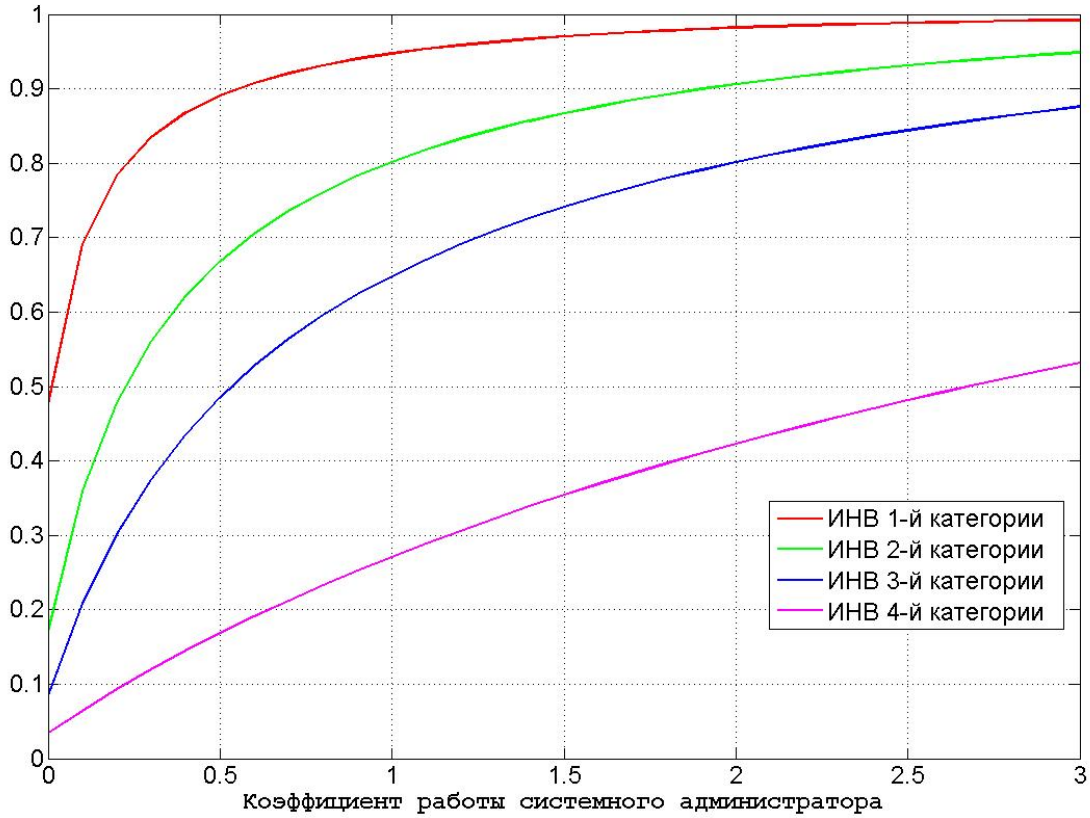

Рисунок 4.8 – Прогноз вероятности нахождения типового сервера публикации отозванных сертификатов в надежном состоянии на период с сентября 2013 года по февраль 2014 года включительно

Анализируя графики 4.7 и 4.8, можно сделать следующие выводы:

1. Если на типовой сервер публикации отозванных сертификатов будет пытаться осуществить НВ ИНВ 1-й категории, то даже при коэффициенте работы системного администратора, равном 3, приблизительно в 70% случаев он сможет нарушить правильную работу ИС. Если же на ИС будут негативно воздействовать ИНВ более высокой категории, то правильная работа ИС будет нарушена практически в 100% случаев.

2. Если системный администратор, обслуживающий типовой сервер публикации отозванных сертификатов, совсем не обновляет ПО, установленное в нем (коэффициент работы системного администратора равен 0), то даже при

попытках НВ со стороны ИНВ 1-й категории работа ИС будет нарушена в среднем в течение половины периода прогноза (приблизительно 90 дней), в остальных случаях это время будет больше.

3. Если ПО типового сервера публикации отозванных сертификатов обновляется автоматически (коэффициент работы системного администратора равен 1), то при попытках НВ со стороны ИНВ 1-й категории в работа ИС будет нарушена в среднем в течение 0,05 периода прогноза (приблизительно 9 дней), при попытках НВ со стороны ИНВ 2-й категории – в течение 0,2 периода прогноза (приблизительно 36 дней), при попытках НВ со стороны ИНВ 3-й категории – в течение 0,35 периода прогноза (приблизительно 63 дня), при попытках НВ со стороны ИНВ 4-й категории – в течение 0,73 периода прогноза (приблизительно 131 день)

4. Если контролем надежности типового сервера публикации отозванных сертификатов занимается служба безопасности (крайне редкий случай), в которую входит большое число высококвалифицированных сотрудников, и применяются дополнительные организационные и технические меры по устранению уязвимостей из ПО (коэффициент работы системного администратора равен 3), то при попытках НВ со стороны ИНВ 4-й категории работа ИС будет нарушена в среднем в течение приблизительно половины периода прогноза (приблизительно 90 дней).

Поскольку исключать попытки НВ со стороны ИНВ 4-й категории не-возможно, и, более того, в большинстве случаев коэффициент работы системного администратора, обслуживающего сервер публикации отозванных сертификатов, не превышает 1 (а часто вообще равен 0), то в отношении данной ИС можно сделать вывод о ее ненадежности (возможно нарушение работы в течение периода от 9 дней и более в зависимости от коэффициента работы системного администратора, обслуживающего сервер публикации отозванных сертификатов, и от категории ИНВ, оказывающего НВ на данную ИС).

Нарушение работоспособности ИС пользователя УЦ не повлияет на электронный документооборот с применением цифровой подписи между другими пользователями УЦ, в то время как нарушение работоспособности сервера публикации отозванных сертификатов полностью нарушит его, та как получатель документа не сможет удостовериться в действительности цифровой подписи, которой данный документ будет подписан. Поэтому применение дополнительных мер повышения надежности рассматриваемой ИС с точки зрения работоспособности всей системы в целом является не менее актуальным, чем для ИС пользователя. Для обеспечения высоконадежной работы рассматриваемой ИС рекомендуется (помимо применения рекомендуемых выше решений) использовать дублирующие серверы и механизмы публикации и проверки работоспособности критической для системы информации.

4.4. Оценка надежности типового центра регистрации

В обычной конфигурации КриптоПро УЦ 1.5 [91] на центре регистрации поднят сервис удалённого рабочего места пользователя УЦ, который реализован с использованием Microsoft Internet Information Services (IIS). Доступ к данному сервису (в типовом исполнении) открыт через межсетевой экран любому пользователю сети Интернет. Например, указанная возможность присутствует на УЦ Росреестра [93], на УЦ Министерства здравоохранения [94]. В публичном пространстве отсутствует какая-либо информация об уязвимостях в типовых межсетевых экранах, которыми может быть защищен центр регистрации, но известно, что они не анализируют запросы, передаваемые через протокол службы Microsoft Internet Information Services (далее IIS), а в ИС центра регистрации данный протокол может быть открыт, следовательно, математическая модель ИС центра регистрации будет включать только СМО, моделирующую динамику уязвимостей в IIS. Типовой центр регистрации может быть построен на основе операционной

системы Windows Server 2003, в которой используется IIS 6 (математическая модель ИС представлена на рис. 4.9), либо на основе операционной системы Windows Server 2008, в которой используется IIS 7 (математическая модель ИС представлена на рис. 4.10).

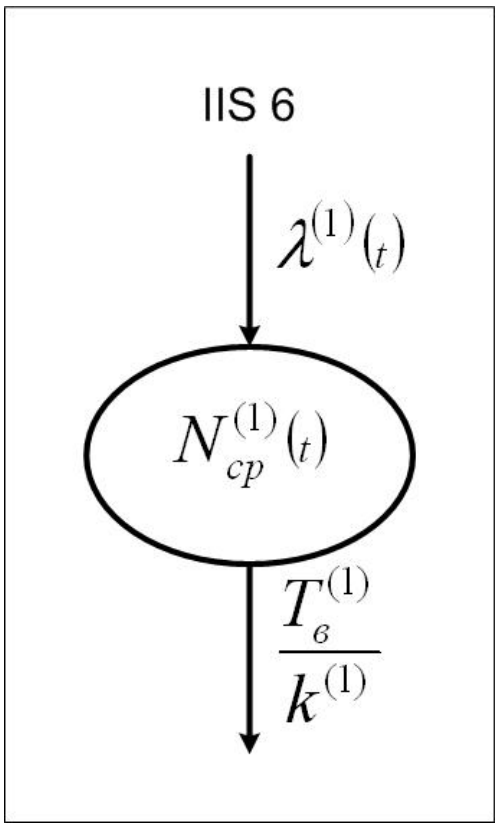

Рисунок 4.9 – Модель ИС регистрационного центра (1-й вариант)

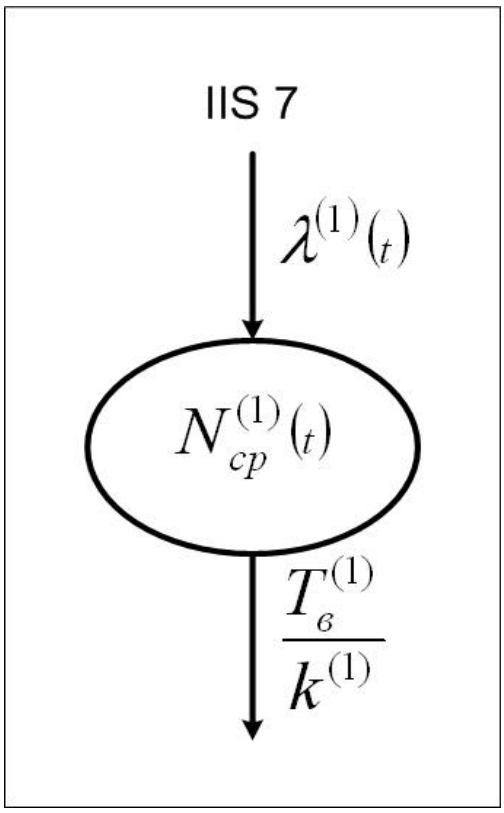

Рисунок 4.10 – Модель ИС регистрационного центра (2-й вариант)

Предлагается осуществить прогноз для 2-х данных вариантов конфигурации типового центра регистрации и сравнить их между собой.

Необходимые статистические данные для прогноза интенсивности обнаружения уязвимостей в каждом конкретном ПО берутся из [51]. Для прогноза интенсивности обнаружения уязвимостей использовался алгоритм, предложенный в подглаве 2.1. Данные прогноза представлены в таблице 4.3

Таблица 4.3 – Прогноз интенсивности обнаружения уязвимостей в ПО 2-х вариантов типового центра регистрации на период с сентября 2013 года до февраля 2014 года включительно

ПО / Месяц	IIS 6	IIS 7
сентябрь 2013	0,0	0,1
октябрь 2013	0,0	0,1
ноябрь 2013	0,1	0,1
декабрь 2013	0,1	0,1
январь 2014	0,2	0,0
февраль 2014	0,3	0,0

Исходя из статистики [51,77,79], время устранения уязвимостей вендором из IIS 6 равно 1,8 дня, из IIS 7 – 0,5 дня. Поскольку точные данные по работе системных администраторов типовых центров регистрации и ИНВ, которые могли бы негативно воздействовать на них, отсутствуют, предлагается осуществить прогноз для коэффициента работы системного администратора от 0 до 3 с шагом 0,1 и следующих ИНВ [81-85]:

- ИНВ 1-й категории ($T_{\text{по}} = 60$ дней, $T_{\text{уязв}} = 30$ дней, $T_{\text{нв}} = 30$ дней);
- ИНВ 2-й категории ($T_{\text{по}} = 20$ дней, $T_{\text{уязв}} = 10$ дней, $T_{\text{нв}} = 10$ дней);
- ИНВ 3-й категории ($T_{\text{по}} = 10$ дней, $T_{\text{уязв}} = 5$ дней, $T_{\text{нв}} = 5$ дней);
- ИНВ 4-й категории ($T_{\text{по}} = 5$ дней, $T_{\text{уязв}} = 1$ день, $T_{\text{нв}} = 1$ день);

Предполагается, что на ИС будет оказывать НВ только один ИНВ, поэтому, исходя из математической модели ИС, предлагается использовать имитационную модель, предложенную в п. 3.1 (модифицированную для 1-го вида ПО).

Далее приведен прогноз вероятности надежности и вероятности нахождения в надежном состоянии типового центра регистрации в случае, когда на нем

установлена операционная система Windows Server 2003 (используется IIS 6) и в случае, когда на нем установлена операционная система Windows Server 2008 (используется IIS 7).

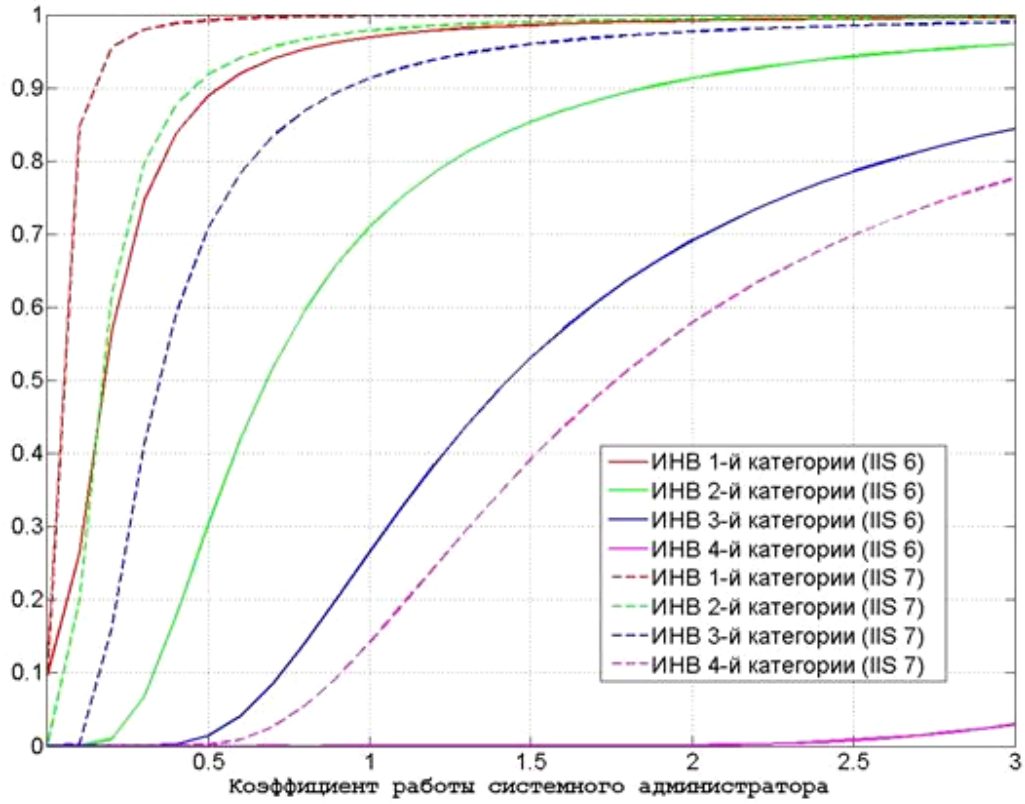

Рисунок 4.11 – Прогноз вероятности надежности типового центра регистрации на период с сентября 2013 года по февраль 2014 года включительно в случае, когда на нем установлена операционная система Windows Server 2003 (используется IIS 6), и в случае, когда на нем установлена операционная система Windows Server 2008 (используется IIS 7)

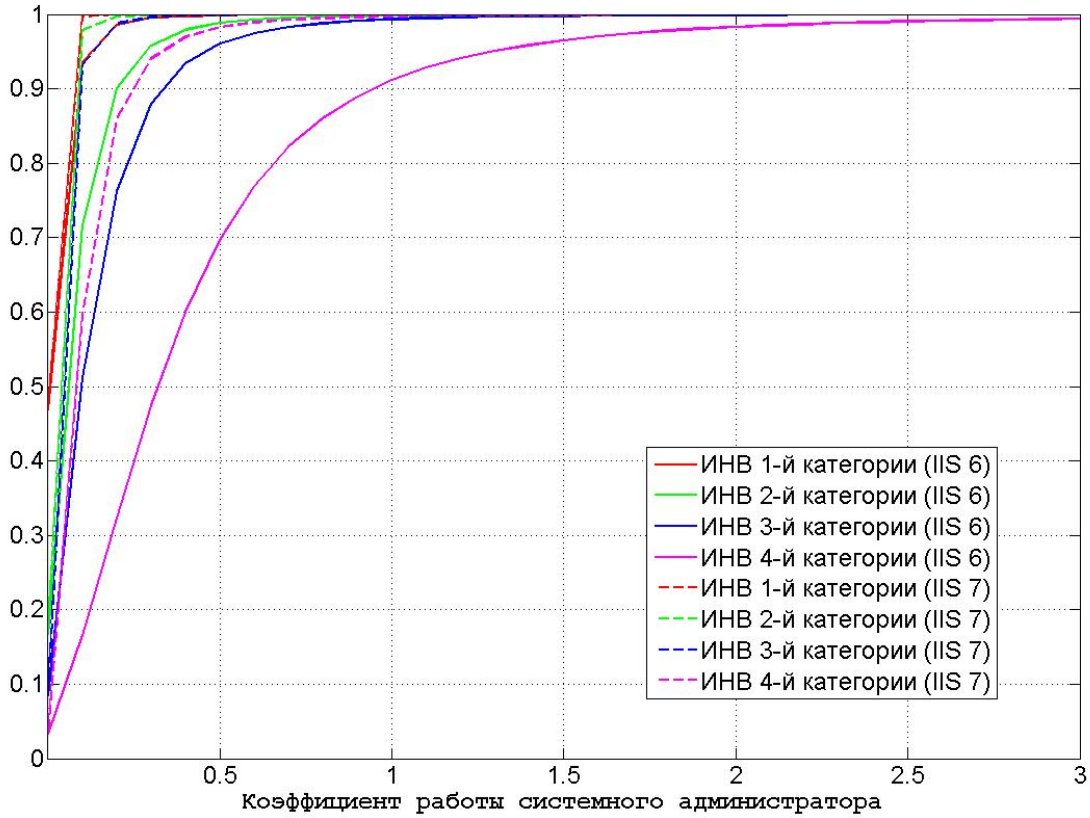

Рисунок 4.12 – Прогноз вероятности нахождения типового центра регистрации в надежном состоянии на период с сентября 2013 года по февраль 2014 года включительно в случае, когда на нем установлена операционная система Windows Server 2003 (используется IIS 6), и в случае, когда на нем установлена операционная система Windows Server 2008 (используется IIS 7)

Данные прогноза (рис. 4.11-4.12) показывают, что при НВ ИНВ 1-й категории вероятность надежности типового центра регистрации с операционной системой Windows Server 2008 выше на 13% по сравнению с вероятностью надежности типового центра регистрации с операционной системой Windows Server 2003., при НВ ИНВ 2-й категории – в 14,6 раза, при НВ ИНВ 3-й категории – более чем в 7×10^3 раз, а при НВ ИНВ 4-й категории – более чем в 7×10^7 раз. Если считать критичной удачную попытку НВ на ИС, нарушающую ее работоспособность вне зависимости от того, на какой промежуток времени она будет нарушена, то с учетом

того, что коэффициент работы системного администратора обычно не превышает 1, типовой центр регистрации с операционной системой Windows Server 2003 и включенной службой IIS стоит считать ненадежным при возможных НВ со стороны ИНВ любых категорий, а типовой центр регистрации с операционной системой Windows Server 2008 и включенной службой IIS стоит считать надежным только при возможных НВ со стороны ИНВ не выше 1-й категории (с небольшими допущениями - не выше 2-й категории). Если же оценивать временные характеристики надежности ИС центра регистрации (время нахождения ИС в надежном состоянии), то при коэффициенте работы системного администратора, равном 1, его можно считать надежным в случае, если на нем установлена операционная система Windows Server 2008, и надежным с ограничениями (если вероятность НВ ИНВ 4-й категории на центр регистрации крайне мала) в случае, если на нем установлена операционная система Windows Server 2003.

При нарушении работы центра регистрации пользователи УЦ не смогут ни оперативно получить новые сертификаты, ни приостановить действие скомпрометированных ключей и их сертификатов. Кроме того, может быть нарушена работа АРМ администратора, которое в своей работе также использует указанный сервис центра регистрации. Таким образом, для повышения надежности рассматриваемой ИС рекомендуется закрыть доступ к IIS из незащищенных сетей, что в большинстве УЦ уже сделано [91], но не во всех [93,94].

Анализируя прогноз надежности для всей рассматриваемой системы электронного документооборота с использованием электронной подписи УЦ – пользователь УЦ, можно сделать вывод, что наиболее слабым (ненадежным) звеном являются ИС пользователей и сервер публикации отозванных сертификатов. Полученные результаты свидетельствуют о необходимости принятия дополнительных мер по повышению надежности ИС, реализующих документооборот с использованием электронной подписи.

Выводы по главе

1. На основе предложенных моделей и алгоритмов оценки надежности программного обеспечения выполнены исследования для базовых элементов типовой ИС УЦ (сервера публикации отозванных сертификатов и центра регистрации) и типовой ИС пользователя УЦ.

2. В ходе выполненных расчетов с использование предложенного методического обеспечения показано, что наиболее ненадежным звеном системы документооборота с использованием электронной подписи являются ИС пользователей, но при этом вся система сохраняет свою работоспособность. А в УЦ наиболее ненадёжным звеном является сервер публикации отозванных сертификатов, и, более того, нарушение его работоспособности является наиболее критичным для системы электронного документооборота в целом. Для усиления его надежности предлагается использовать СЗИ, не позволяющие внешним ИНВ иметь непосредственный доступ к ПО, установленному на сервере публикации отозванных сертификатов. Центр регистрации УЦ является ненадежным в случае, если открыт протокол службы IIS, при этом при использовании IIS 6-й версии (Windows Server 2003) его надежность существенно (до 7×10^7 раз) меньше, чем при использовании IIS 7-й версии (Windows Server 2008). Для увеличения надежности центра регистрации предлагается закрыть доступ к IIS из незащищенных сетей.

3. ИС пользователей УЦ в подавляющем большинстве являются ненадежными (правильная работа ИС может быть нарушена в 90% случаев и более в зависимости от категории ИНВ, пытающегося оказать НВ на ИС). Для увеличения их надежности предлагается использовать выделенные защищенные рабочие места, включающие: лицензионное и только необходимое ПО, СЗИ, обеспечивающее контроль доступа и целостность ПО (имеющее низкую интенсивность обнаружения уязвимостей и закрывающее непосредственный доступ к уязвимостям в ПО),

конфигурации ИС и её аппаратных элементов, сетевое взаимодействие которых ограничено белым списком проверенных сетевых узлов.

4. Полученные результаты свидетельствуют о необходимости принятия дополнительных мер по повышению надежности ИС, реализующих документооборот с использованием электронной подписи, и в которых неприемлемы существующие риски нарушения работоспособности.

Заключение

В ходе выполнения диссертационной работы поставлены и решены следующие научные задачи:

- выявлены наиболее важные факторы, влияющие на надежность использования ПО в ИС, определены основные требования к разрабатываемым алгоритмам и моделям анализа надежности использования ПО в ИС в условиях конфликтных взаимодействий и произведен анализ современных подходов к оценке надежности использования ПО в ИС на предмет учета данных факторов и требований;
- разработаны и исследованы модели и алгоритмы оценки и прогнозирования наиболее важных факторов, влияющих на надежность использования ПО в ИС в условиях конфликтных взаимодействий;
- разработаны и исследованы алгоритмы и модели оценки надежности использования ПО в ИС в условиях конфликтных взаимодействий, учитывающие наиболее важные факторы и соответствующие основным требованиям, предъявленным в ходе анализа.

В ходе проведенных исследований получены следующие основные результаты и выводы, выносимые на защиту.

1. Разработан двухэтапный нейросетевой алгоритм статистического анализа и прогнозирования нестационарных временных последовательностей, основанный на применении на первом этапе специальной процедуры интерполяции экспериментальных данных в виде разложения по радиально-базисным функциям с нахождением коэффициентов разложения с использованием метода регуляризации, а также на втором этапе – процедуры прогнозирования на основе нейронной сети в виде многослойного персептрона, обученной по интерполированным данным. Использование указанного алгоритма для прогноза интенсивности уязвимостей программного обеспечения позволяет обеспечить повышение точности прогноза в среднем на 10% , а в отдельных случаях и до 70% по сравнению с известными.

2. Предложены математические модели динамики состояний программ и информационной системы в целом с учетом возможных уязвимостей и общий алгоритм оценки надежности использования программного обеспечения, учитывающие зависимости интенсивности обнаружения уязвимостей от времени, временных характеристик закрытия уязвимостей от работы производителя программного обеспечения и администратора информационной системы, что позволяет повысить степень обоснованности получаемых оценок.

3. Разработаны объектно-ориентированные и математические модели использования информационных технологий и информационной системы в целом в динамике конфликтного взаимодействия, обеспечивающие оценку надежности в дуэльных ситуациях при выполнении ограничения в виде пуассоновского характера потоков, описывающих переходы между состояниями информационной системы и источника внешних воздействий.

4. Предложены компьютерные имитационные модели использования информационных технологий и информационной системы в целом в динамике конфликтного взаимодействия, основанные на применении формализма гибридных автоматов Харела, обеспечивающие оценку надежности информационных систем в условиях воздействия коалиции внешних источников при произвольном характере статистики переходов между состояниями информационной системы и источников внешних воздействий.

5. Даны практические рекомендации относительно использования базовых элементов типовой ИС удостоверяющего центра и типовой ИС пользователя, обеспечивающие повышение надежности использования устанавливаемого типового программного обеспечения.

6. Одним из достоинств разработанных моделей и алгоритмов является их достаточно простая адаптация к новым вариантам конфликтных ситуаций, а также к более глубокому учету процессов функционирования информационных систем. Следовательно, данные модели и алгоритмы могут быть использованы как основа для последующих исследований в области надежности использования программного обеспечения в информационных системах.

7. Результаты диссертационной работы имеют практическое значение для проведения исследований надежности реальных информационных технологий и информационных систем, а также отдельного программного обеспечения. Оценка результатов данных исследований позволит:

- пользователям информационных систем - выявить слабые места в политике обеспечения надежности (оценить работу системного администратора, выявить программное обеспечение, использование которого нежелательно, и т.п.), оценить материальные и иные риски, которым может подвергнуться информационная система, а также выработать рекомендации по их уменьшению;

- разработчикам программного обеспечения - оценить надежность использования их продуктов, выявить наиболее уязвимые из них, и, соответственно, рациональнее распределить финансовые и иные ресурсы при поддержке уже существующего программного обеспечения и разработке нового;

- организациям, осуществляющим аттестацию информационных систем и сертификацию программного обеспечения – точнее оценить реальные процессы функционирования информационных систем в условиях конфликтных взаимодействий, выработать на основе разработанных моделей и алгоритмов новую методологию, более полно учитывающую данные процессы.

Таким образом, в диссертации решены все поставленные задачи научного исследования и цель работы достигнута.

Список литературы

1. Уткин, Л. В. Методы и модели анализа надежности и безопасности информационных систем при неполной информации: автореф. дис. на соиск. учен. степ. докт. техн. наук : 05.13.18 / Уткин Лев Владимирович. – С.Пб., 2001.

2. Daintith, J. "IT" / J. Daintith // A Dictionary of Physics. – Oxford University Press, 2012.

3. Ethical Hacking and Countermeasures: Attack Phases / M. Bellegarde, M. Orvis, S. Helba. – EC-Council Press, 2010.

4. Скембрей, Дж. Секреты хакеров. Безопасность сетей – готовые решения / Дж. Скембрей, Ст. Мак-Клар, Дж. Курц. – М.: Вильямс, 2001. – 656 с.

5. Шелухин, О. И. Обнаружение вторжений в компьютерные сети [сетевые аномалии] / О. И. Шелухин, Д. Ж. Сакалема, А. С. Филинова. – М.: Горячая линия – Телеком, 2013. – 220 с.

6. Alhazmi, O.H. Modeling the Vulnerability Discovery Process / O.H. Alhazmi, Y.K. Malaiya. // Proc. Int. Symp. Software Reliability Eng, Nov. 2005, pp. 129-138.

7. ГОСТ 28195-89 Оценка качества программных средств. – М.: Изд-во стандартов, 1989. – 38 с.

8. ГОСТ Р ИСО/МЭК 9126-93 Оценка программной продукции. Характеристики качества и руководства по их применению. – М.: Изд-во стандартов, 1994. – 15 с.

9. A Complete Guide to the Common Vulnerability Scoring SystemVersion 2.0 [Электронный ресурс] // FIRST. – Режим доступа: http://www.first.org/cvss/cvss-guide.

10. ГОСТ 27.002-89 Надежность в технике. Основные понятия. Термины и определения. – М.: Издательство стандартов, 1990. – 37 с.

11. The Laws of Vulnerabilities: Six Axioms for Understanding Risk [Электронный ресурс] // Qualys. – Режим доступа: http://www.qualys.com/docs/Laws-Report.pdf.

12. The Laws of Vulnerabilities 2.0 [Электронный ресурс] // Qualys. – Режим доступа: http://www.qualys.com/docs/Laws_2.0.pdf.

13. Microsoft Security Intelligence Report v14 [Электронный ресурс] // Mictosoft. – Режим доступа: http://www.microsoft.com/security/sir/.

14. Microsoft Security Intelligence Report v15 [Электронный ресурс] // Mictosoft. – Режим доступа: http://www.microsoft.com/security/sir/.

15. Internet Security Threat Report 2011 Trends [Электронный ресурс] // Symantec. – Режим доступа: http://owasp.com/images/7/70/Symantec_ISTR_17.pdf.

16. Липаев, В.В. Надежность программного обеспечения /В.В. Липаев – М.: Радио и связь, 1998. – 200 с.

17. Musa, J.D. Software Reliability: Measurement, Prediction, Application / J.D. Musa, A. lannino, K. Okumoto // N.Y. McGraw Hill, 1990.

18. Shooman, M.L. Software Engineering: Reliability, Developmen and Management / M.L. Shooman // N.Y. McGraw-Hill. 1983.

19. Методика определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных: утверждена Федеральной службой по техническому и экспортному контролю Российской Федерации 14 февраля 2008 г.

20. Об утверждении требований к средствам электронной подписи и требований к средствам удостоверяющего центра: приказ Федеральной Службы Безопасности №796 от 27.12.2011.

21. ГОСТ Р ИСО/МЭК 15408-1-2008 Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель. – М.: Госстандарт России, 2008. – 35 с.

22. Нестеров, С. Анализ и управление рисками в информационных системах на базе операционных систем Microsoft / С. Нестеров [Электронный ресурс]. – Режим доступа: http://www.intuit.ru/studies/courses/531/387/info.

23. Симонов, С. Анализ рисков, управление рисками / С. Симонов // Jet Info. Информационный бюллетень, 1999. № 1. с. 2-28.

24. Симонов, С.В. Технологии и инструментарий для управления рисками / С.В. Симонов // Jet Info. - № 2. - 2003. – с. 3-32.

25. The logic behind CRAMM’s assessment of measures of risk and determination of appropriate countermeasures [Электронный ресурс]. – Режим доступа: http://www.cramm.com/downloads/techpapers.htm.

26. Thomas R. Peltier. Information security risk analysis / R. Peltier Thomas. – Auerbach Pub, 2001. – P 281.

27. OCTAVE (Operationally Critical Threat, Asset, and Vulnerability EvaluationSM) [Электронный ресурс] // CERT. – Режим доступа: www.cert.org/octave.

28. Using vulnerability assessment tools to develop an OCTAVE Risk Profile [Электронный ресурс] // SANS Institute. InfoSec Reading Room. – Режим доступа: http://www.sans.org/reading-room/whitepapers/auditing/vulnerability-assessment-tools-develop-octave-risk-profile-1353.

29. Щеглов, А.Ю. Безопасность современных ОС «в цифрах» [Электронный ресурс] / А. Ю. Щеглов. – Режим доступа: http://www.itsec.ru/articles2/Inf_security/bezopasnost-OS.

30. Застрожнов, И. И. Модель конфликта злоумышленника и системы защиты информации / И.И. Застрожнов, Д.И. Коробкин, А.А. Окрачков, Е.А. Рогозин. - Вестник Воронежского государственного технического университета. 2009. Т. 5. № -6. С. 142-149.

31. Климов, И. З. Оценка надежности систем защиты информации от несанкционированного доступа / И. З. Климов, А. А. Пономарев. - Вестник Ижевского государственного технического университета. 2008. №-3. С. 102-103.

32. Козирацкий, Ю.Л. Модели информационного конфликта средств поиска и обнаружения / С.А. Будников, А.Ю. Козирацкий, Ю.Л. Козирацкий и др. Под ред. Козирацкого Ю.Л. – М: Издательство «Радиотехника», 2013 г. – 232 с.

33. Вялых, А.С. Динамика уязвимостей в современных защищенных информационных системах / А.С. Вялых, С.А. Вялых // Вестник Воронежского

государственного ун-та. Серия: Системный анализ и информационные технологии. - 2011. - № 2. - С. 59-63.

34. Вялых, А.С. Оценка эффективности сигнатурных методов обнаружения вредоносных программ / А.С. Вялых, С.А. Вялых // Вестник Воронежского государственного ун-та. Серия: Системный анализ и информационные технологии. - 2011. - № 2. - С. 64-66.

35. Вялых, А.С. Оценка уязвимости современных информационных систем / А.С. Вялых, С.А. Вялых // Информатика : проблемы, методология, технологии : матер. XI Международ. науч. – метод. конф., Воронеж, 10-11 февр. 2011 г. – Воронеж : ИПЦ ВГУ, 2011. – Т. 1. – С. 168-172.

36. Вялых, А.С. Оценка эффективности обнаружения вредоносных программ / А.С. Вялых, С.А. Вялых // Информатика : проблемы, методология, технологии : матер. XI Международ. науч. – метод. конф., Воронеж, 10-11 февр. 2011 г. – Воронеж : ИПЦ ВГУ, 2011. – Т. 1. – С. 172-176.

37. Вялых, А.С. Оценка возможностей атаки на информационную систему / Вялых А.С., Вялых С.А. // Кибернетика и высокие технологии XXI века : матер. XII международ. науч.-тех. конф., Воронеж, 11-12 мая 2011 г. – Воронеж : ИПЦ ВГУ, 2011. – Т.1. – С. 91-96.

38. Сирота, А.А. Имитационная модель ситуационного конфликта информационной системы и злоумышленника / А.А. Сирота, А.С. Вялых, С.А. Вялых // Информатика : проблемы, методология, технологии : матер. XII Международ. науч. – метод. конф., Воронеж, 9-10 февр. 2012 г. – Воронеж : ИПЦ ВГУ, 2012. – Т. 1. – С. 359-361.

39. Вялых, А.С. Имитационная модель конфликта информационной системы и коалиции злоумышленников / А.С. Вялых, С.А. Вялых, А.А. Сирота // Кибернетика и высокие технологии XXI века : матер. XIII международ. науч.-тех. конф., Воронеж, 15-16 мая 2012 г. – Воронеж : НПФ «САКВОЕЕ» ООО, 2012. – Т.2. – С. 413-424.

40. Вялых, А.С. Оценка уязвимости информационной системы на основе ситуационной модели динамики конфликта / А.С. Вялых, С.А. Вялых, А.А. Сирота // Информационные технологии. - 2012. - № 9. - С. 16-21.

41. Сирота, А.А. Прогнозирование динамики обнаружения уязвимостей программного обеспечения при помощи нейросетевых алгоритмов обработки информации / А.А. Сирота, А.С. Вялых, С.А. Вялых // Информатика : проблемы, методология, технологии : матер. XIII Международ. науч. – метод. конф., Воронеж, 7-8 февр. 2013 г. – Воронеж : ИПЦ ВГУ, 2013. – Т. 3. – С. 224-228.

42. Вялых, А.С. Использование нейросетевых алгоритмов обработки информации для прогнозирования динамики обнаружения уязвимостей в современном программном обеспечении / А.С. Вялых, С.А. Вялых, А.А. Сирота // Кибернетика и высокие технологии XXI века : матер. XIV международ. науч.-тех. конф., Воронеж, 14-15 мая 2013 г. – Воронеж : НПФ «САКВОЕЕ» ООО, 2013. – Т.2. – С. 417-422.

43. Вялых, А.С. Нейросетевой алгоритм обработки информации для прогнозирования надежности программного обеспечения / А.С. Вялых, С.А. Вялых, А.А. Сирота // Вестник Воронежского государственного ун-та. Серия: Системный анализ и информационные технологии. - 2013. - № 2. – С. 140-143.

44. Ozment, A. Vulnerability discovery & software security: dissertation for the degree of Ph.D. / Andy Ozment. – University of Cambridge, 2007. – 139 p.

45. Frei, S. Security econometrics - the dynamics of (in)security: dissertation for the degree of Doctor of Science / Stefan Frei. – ETH Zurich, 2009. – 184 p.

46. Rescorla, E. Is finding security holes a good idea? / E. Rescorla // Security and Privacy, Jan-Feb 2005. – pp. 14-19.

47. Ozment, A. Improving Vulnerability Discovery Models: Problems with Definitions and Assumptions / A. Ozment // In the proceedings of the Third Workshop on Quality of Protection (QoP’07). October 29, 2007: Alexandria, VA, US.

48. Joh, H. Defining and Assessing Quantitative Security Risk Measures Using Vulnerability Lifecycle and CVSS Metrics / H. Joh, Y.K. Malaiya // SAM'11, The 2011 International Conference on Security and Management, 2011. – pp.10-16.

49. Okamura, H. Quantitative Security Evaluation for Software System from Vulnerability Database / H. Okamura, M. Tokuzane, T. Dohi // Journal of Software Engineering and Applications, Vol. 6 No. 4A, 2013. – pp. 15-23.

50. Frei, S. Large-scale vulnerability analysis / S. Frei, M. May, U. Fiedler, B. Plattner // In LSAD '06: Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense. – New York, NY, USA, 2006. – pp. 131–138.

51. National Vulnerability Database [Электронный ресурс] // National Institute of Standards and Technology. – Режим доступа: http://nvd.nist.gov.

52. McKinney, D. Vulnerability Bazaar / D. McKinney // IEEE Security and Privacy, vol. 5, no. 6, 2007. – pp. 69-73.

53. Miller, C. The legitimate vulnerability market: Inside the secretive world of 0-day exploit sales / C. Miller // in Workshop on the Economics of Information Security (WEIS 2007), 2007.

54. Bohme, R. Vulnerability Markets. What is the Economic Value of a Zero-Day Exploit? / R. Bohme // in Private Investigations (Proc. of 22nd Chaos Communication Congress), 2005.

55. Report on the Underground Economy 2008 [Электронный ресурс] // Symantec. – Режим доступа: http://eval.symantec.com/mktginfo/enterprise/white_papers/b-whitepaper_underground_economy report_11-2008-14525717.en-us.pdf.

56. Vulnerability Contributor Program [Электронный ресурс] // iDefense. – Режим доступа: http://labs.idefense.com/vcp.

57. Zero Day Initiative [Электронный ресурс] // TippingPoint. – Режим доступа: http://www.zerodayinitiative.com.

58. Keizer, G. Bug bounties not dangerous, 3Com claims [Электронный ресурс] / G. Keizer. – Режим доступа: http://news.techworld.com/security/9768/bug-bounties-not-dangerous-3com-claims/?intcmp=nws-af-rtd-home.

59. Основы математического анализа: Учебник для физ. специальностей и специальности "Прикладная математика" ун-тов / В. А. Ильин, Э. Г. Позняк . – М.

: Наука, 1980- .— (Курс высшей математики и математической физики / под ред. А.Н. Тихонова [и др.] ; Вып. 2) .Ч. 2 . – 1980 . – 447 с.

60. Alhazmi, O.H. Quantitative vulnerability assessment of systems software / O.H. Alhazmi, Y.K. Malaiya // Proceedings of 51st Annual Reliability and Maintainability Symposium, Alexandria, VA, USA, January 2005. – pp. 615-620.

61. Alhazmi, O.H. Security Vulnerabilities in Software Systems: A Quantitative Perspective / O.H. Alhazmi, Y.K. Malaiya, I. Ray // Proceedings Annual IFIP WG11.3 Working Conference on Data and Information Security, Storrs, CT, USA, August 2005. – pp. 281-294.

62. Alhazmi, O.H. Prediction Capability of Vulnerability Discovery Models / O.H. Alhazmi, Y.K. Malaiya // Proceedings of 52nd Annual Reliability and Maintainability Symposium, Newport Beach, CA, USA, January 2006. – pp. 86-91.

63. Alhazmi, O.H. Measuring and Enhancing Prediction Capabilities of Vulnerabilities Discovery Models for Apache and IIS HTTP Servers / O.H. Alhazmi, Y.K. Malaiya // Proceedings of 17th IEEE International Symposium on Software Reliability Engineering, Raleigh, NC, USA, November 2006. – pp. 343-352.

64. Woo, S.-W. Assessing Vulnerabilities in Apache and IIS HTTP Servers / S.-W. Woo, O.H. Alhazmi, Y.K. Malaiya // Proceedings the 2nd IEEE International Symposium on Dependable Autonomic and Secure Computing, Indianapolis, IN, USA, September 2006. – pp. 103-110.

65. Woo, S.-W. An Analysis of the Vulnerability Discovery Process in Web Browsers / S.-W. Woo, O.H. Alhazmi, Y.K. Malaiya // Proceedings of 10th IASTED International Conference on Software Engineering and Applications, Dallas, TX, USA, November 2006. – pp. 172-177.

66. Alhazmi, O.H. Assessing Vulnerabilities in Software Systems: A Quantitative Approach: dissertation for the degree of Doctor of Philosophy / Omar Alhazmi. – CS Dept, Colorado State University, 2006. – 165 p.

67. Alhazmi, O.H. Measuring, Analyzing and Predicting Security Vulnerabilities in Software Systems / O.H. Alhazmi, Y.K. Malaiya, I. Ray // Computers and Security Journal, Volume 26, Issue 3, May 2007. – pp. 219-228.

68. Alhazmi, O.H. Application of Vulnerability Discovery Models to Major Operating Systems / O.H. Alhazmi, Y.K. Malaiya // IEEE Trans. Reliability, March 2008. – pp. 14-22.

69. Joh, H. Seasonality in Vulnerability Discovery in Major Operating Systems and Web Applications / H. Joh, Y.K. Malaiya // Fast Abstract, Proc. Int. Symp. Software Reliability Eng., Nov. 2008. – pp. 297-298.

70. Joh, H. Seasonal Variation in the Vulnerability Discovery Process / H. Joh, Y.K. Malaiya // Proc. 2nd IEEE Int. Conf. Software Testing, Verification, and Validation, April 2009. – pp. 191-200.

71. Оссовский, С. Нейронные сети для обработки информации / С. Оссовский. – М.: Финансы и статистика, 2002. – 344 с.

72. Хайкин, С. Нейронные сети: полный курс, 2-е изд., испр. : Пер. с англ. / С. Хайкин. – М.: ООО "И.Д. Вильямс", 2006. – 1104 с.

73. Севастьянов, Л.А., Ловецкий К.П., Ланеев Е.Б. Регулярные методы и алгоритмы расчета обратных задач в моделях оптических структур: Учебное пособие / Л.А. Севастьянов, К.П. Ловецкий, Е.Б. Ланеев. – М.: РУДН, 2008. – 135 с.

74. Тихонов, А.Н. Нелинейные некорректные задачи / А.Н. Тихонов, А.С. Леонов, А.Г. Ягола. – М.:Наука, 1995.

75. Тихонов, А.Н. Численные методы решения некорректных задач / А.Н. Тихонов, А.В. Гончарский, В.В. Степанов, А.Г. Ягола. – М.:Наука, 1990. – 232 с.

76. Саати, Т. Элементы теории массового обслуживания и ее приложения / Т. Саати. – М.: Мир, 1991. – 397 с.

77. The Open Source Vulnerability Database [Электронный ресурс]. – Режим доступа: http://osvdb.org.

78. Хачатурова, С. М. Математические методы системного анализа (электронный учебник) [Электронный ресурс] // С.М Хачатурова. – Режим доступа: http://ermak.cs.nstu.ru/mmsa/main/Proba.htm.

79. Microsoft Corp [Электронный ресурс]. – Режим доступа: http://microsoft.com.

80. Фаулер, М. UML. Основы. 2-ое изд. Краткое руководство по унифицированному языку моделирования / М. Фаулер, К. Скотт.: Пер. с англ. СПб.: Издательство: «Символ-Плюс», 2006. – 192 с.

81. Форум на сайте xakep.ru [Электронный ресурс] // xakep.ru. – Режим доступа: http://forum.xakep.ru.

82. Форум на сайте hackzone.ru [Электронный ресурс] // hackzone.ru. – Режим доступа: http://www.hackzone.ru/forum.

83. Форум CRACK FORUM [Электронный ресурс] // CRACK FORUM. – Режим доступа: http://www.crack-forum.ru.

84. Форум EXELAB [Электронный ресурс] // EXELAB. – Режим доступа: http://exelab.ru/f/.

85. The Anatomy of an Anonymous Attack, Hacker Intelligence Summary Report [Электронный ресурс] // Imperva. – Режим доступа: http://www.imperva.com/docs/hii_the_anatomy_of_an_anonymous_attack.pdf.

86. Кельберт, М. Я. Вероятность и статистика в примерах и задачах. Т. II: Марковские цепи как отправная точка теории случайных процессов и их приложения / М.Я. Кельберт, Ю.М. Сухов. – М.: МЦНМО, 2009. – 295 с.

87. Алгазинов, Э.К. Анализ и компьютерное моделирование информационных процессов и систем / Э. К. Алгазинов, А. А. Сирота. Под общ. ред. А. А. Сироты. – М.: Диалог-МИФИ, 2009. – 416 с.

88. Тихонов, В.И. Марковские процессы / В.И. Тихонов, М.А. Миронов. – М.: «Сов. радио», 1977. – 488 с.

89. Об электронной подписи: Федеральный закон от 6 апреля 2011 № 63-ФЗ // СЗ РФ. – 2011. – № 15.

90. Электронная цифровая подпись [Электронный ресурс] // Википедия. – Режим доступа: http://ru.wikipedia.org/wiki/Электронная цифровая подпись.

91. ООО «КРИПТО-ПРО» [Электронный ресурс]. – Режим доступа: http://www.cryptopro.ru/.

92. Удостоверяющий центр Правительства Воронежской области [Электронный ресурс]. – Режим доступа: http://uc.govvrn.ru/.

93. Федеральная служба государственной регистрации, кадастра и картографии (Росреестр) [Электронный ресурс]. – Режим доступа: https://rosreestr.ru/.

94. Министерство здравоохранения РФ [Электронный ресурс]. – Режим доступа: https://www.rosminzdrav.ru/.