УДК 004.738.5

Модель и алгоритм функционирования клиент-серверной информационной системы в условиях сетевой разведки

Максимов Р. В., Орехов Д. Н., Соколовский С. П.

Постановка задачи: *расширение возможностей и повышение результативности сетевой разведки по вскрытию клиент-серверных информационных систем актуализируют вопросы обеспечения их устойчивости к воздействиям дестабилизирующих факторов. Известные способы защиты от сетевой разведки, основывающиеся на реализации принципов пространственного обеспечения безопасности, а также формализации и внедрения множества запрещающих регламентов, основанных на обнаружении и реагировании на факт ведения сетевой разведки или совершения компьютерных атак, не способны эффективно противостоять современным средствам сетевой разведки. Реализация таких мер защиты вынуждает нарушителя далее воздействовать на клиент-серверные информационные системы и (или) менять стратегию воздействия.* ***Целью работы*** *является разработка модели и алгоритма, обеспечивающего оперативное обслуживание максимального количества запросов санкционированных клиентов клиент-серверных информационных систем с одновременным снижением качества обслуживания запросов от средств сетевой разведки.* ***Используемые методы****: решение задачи исследования процесса функционирования клиент-серверной информационной системы в условиях сетевой разведки при различных стратегиях взаимодействующих сторон, а также управления ресурсными возможностями средств сетевой разведки при установлении и поддержании сетевых соединений, заключается в представлении процесса их взаимодействия в виде марковского случайного процесса с дискретными состояниями и непрерывным временем.* ***Новизна:*** *элементами новизны представленной модели является применение математического аппарата теории марковских случайных процессов и решение уравнений Колмогорова для исследования и решения задачи динамического управления ресурсными возможностями клиент-серверной информационной системы за счет управления параметрами сетевых соединений. Новизна разработанного алгоритма заключается в применении представленной модели функционирования клиент-серверной информационной системы для управления ресурсными возможностями средств сетевой разведки при установлении и поддержании сетевых соединений.* ***Результат****: использование представленного решения по динамическому управлению ресурсными возможностями клиент-серверной информационной системы за счет управления параметрами сетевых соединений позволяет повысить результативность защиты за счет снижения вероятности обнаружения нарушителем факта использования средств защиты и идентификации их характеристик, а также увеличения длительности удержания в двухстороннем порядке соединения с нарушителем, за счет имитации канала связи с плохим качеством, а также блокирования попыток средств сетевой разведки разорвать соединение.* ***Практическая значимость****: заключается в нахождении вероятностных и временных характеристик, описывающих состояние процесса функционирования клиент-серверной информационной системы при различных стратегиях установления и поддержания параметров соединений взаимодействующими сторонами. Практическая значимость представленного алгоритма заключается в решении задачи динамической конфигурации параметров сетевых соединений клиент-серверной информационной системы, обеспечивающей дискриминацию трафика средств сетевой разведки, скрытие факта использования средств защиты и идентификации их характеристик.*

Ключевые слова: *клиент-серверная информационная система, компьютерная атака, сетевые соединения, honeypots, network tarpits, протокол, сетевая разведка.*

Библиографическая ссылка на статью:
Максимов Р. В., Орехов Д. Н., Соколовский С. П. Модель и алгоритм функционирования клиент-серверной информационной системы в условиях сетевой разведки // Системы управления, связи и безопасности. 2019. № 4. С. 50-99. DOI: 10.24411/2410-9916-2019-10403.

Reference for citation:
Maximov R. V., Orekhov D. N., Sokolovsky S. P. Model and Algorithm of Client-Server Information System Functioning in Network Intelligence Conditions. *Systems of Control, Communication and Security*, 2019, no. 4, pp. 50-99. DOI: 10.24411/2410-9916-2019-10403 (in Russian).

Актуальность

В настоящее время достаточно большое количество компьютерных атак (КА) носит разведывательный характер с целью получения злоумышленником с помощью средств сетевой разведки (СР) информации о топологии и типологии информационной системы (ИС), являющейся объектом КА, а также об используемых средствах защиты ИС. Возможности СР обусловлены открытостью архитектуры ИС и протоколов информационного обмена (семейства TCP/IP), обеспечивающих взаимодействие через организацию интерфейсов СР с элементами ИС. Именно интерфейсы позволяют осуществить взаимодействие последовательно соединенных устройств и программ полученной совокупной системы, реализующей канал утечки информации [1-5].

Ключевыми фазами взаимодействия являются программное подавление (отказ в обслуживании), контроль событий (наблюдение) и управление (перехват управления). Первая фаза отличается от третьей тем, что может иметь декларативный характер, а СР может потерять возможность реализации диалогового взаимодействия. В случае изолированности ИС средствами обеспечения безопасности информации (ОБИ) взаимодействия реализуют посредством КА и недекларированных возможностей (НДВ), обеспечивая «доставку» технических средств СР в инфраструктуру ИС (обеспечивая контакт технических средств СР с объектом защиты). Диалоговое (программное, протокольное) взаимодействие осуществляется локально от канального уровня ЭМВОС, а удаленно – от сетевого [6-9].

Программные помехи – программные (логические) возмущения, снижающие качество ИС: реальную скорость передачи данных и доступность узлов ИС посредством создания дополнительной (нештатной) нагрузки на процессы и устройства их реализующие. В частности, скорость передачи данных по каналам связи может быть снижена фрагментацией пакетов сообщений, а доступность узлов ИС – компьютерными атаками типа «отказ в обслуживании».

Программное подавление – процесс воздействия преднамеренных программных помех, осуществляемый путем организации и реализации процедурного или декларативного воздействия источника программных помех (средств СР) на элементы ИС, которым присущи НДВ, уязвимости и открытость архитектуры.

Одними из средств сетевой защиты, функционирующих с применением сетевых стратегий, направленных на создание у нарушителя иллюзий уязвимых целей или способствующих видимости более сложной (ложной) инфраструктуры, являются сетевые «приманки» (honeypots) [10-13]. Более совершенные способы введения в заблуждение включают в себя не только предоставление СР правдоподобной цели, но и такие меры как, например, удержание в двухстороннем порядке соединения с отправителем пакетов сообщений, что вызывает «истощение» ресурсов у отправителя пакетов сообщений для поддержания состояния соединения, замедляет процесс автоматического сканирования атакуемой ИС и, как результат, накладывает ограничение на используемый нарушителем вычислительный ресурс, что приводит к невозможности осуществлять нарушителем сетевой информационный обмен. Рассмотренные способы защи-

ты реализованы в виде так называемых сетевых «ловушек» (network tarpits) [14-19].

В свою очередь, нарушителями информационной безопасности также активно разрабатываются и совершенствуются средства снижения результативности сетевых «ловушек», реализующие следующие способы их компрометации: детектирование уникальных идентификаторов (демаскирующих признаков) сетевых «ловушек» и детальный анализ сетевого трафика, поступающего с сетевых «ловушек». Таким демаскирующим признаком сетевой «ловушки» является использование значения служебного поля «размер окна» TCP-пакетов сообщений по умолчанию устанавливаемого равным десяти байтам [20].

В качестве средств компрометации сетевых «ловушек», в части обнаружения факта использования всей совокупности IP-адресов, нарушителем могут применяться различные утилиты (nmap, ethereal, arping и др.), предназначенные для анализа сетевого трафика и топологии ИС.

Анализ работ [21-34], опубликованных по направлению противодействия сетевой разведке показал значительную проработанность вопросов в рассматриваемой предметной области, однако вопросы управления сетевыми соединениями со средствами сетевой разведки, а также снижения демаскирующих признаков средств сетевой защиты, все еще недостаточно раскрыты, что обуславливает актуальность проводимого исследования.

Цель динамической конфигурации параметров сетевых соединений ИС в условиях СР – оперативно обслуживать максимальное количество запросов санкционированных клиентов с одновременным снижением качества обслуживания запросов СР. Так как помехи физического уровня в работе не рассматриваются, целесообразно определить возмущающие факторы внешней среды как совокупность программных помех и программного подавления (компьютерных атак (КА) типа «отказ в обслуживании», так называемых DOS- и DDOS-атак).

Клиент-сервер – это вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между серверами (поставщиками услуг) и клиентами (заказчиками). Фактически клиент и сервер – это программное обеспечение, размещенное на ЭВМ, взаимодействующих через ИС (вычислительную сеть).

Серверное программное обеспечение (ПО) – программный компонент ИС, выполняющий сервисные (обслуживающие) функции по запросам клиентов, предоставляя им доступ к определенным ресурсам или услугам.

Для взаимодействия с клиентами сервер выделяет необходимые ресурсы межпроцессного взаимодействия и ожидает запросы на открытие соединения (или запросы на предоставляемый сервис).

В зависимости от типа такого ресурса, сервер может обслуживать процессы в пределах одной ИС или процессы на других ЭВМ через сети передачи данных. Формат запросов клиента и сервера определяется протоколом.

Серверы классифицируют по типу предоставляемых услуг следующим образом.

Универсальные серверы, предоставляющие доступ клиентов к различным услугам, в том числе:

- inetd internet (от англ. super-server daemon, демон сервисов IP), функционирующие с клиентами через перенаправленные потоки стандартного ввода-вывода в семействе протоколов TCP/IP;
- RPC (от англ. remote procedure call, удаленный вызов процедур), интеграция серверов в виде процедур, доступных для вызова удаленным пользователям через унифицированный интерфейс;
- прикладные клиент-серверные технологии Windows (DCOM, от англ. distributed component object model; OLE, от англ. object linking and embedding; Active-X, позволяющие программам выполнять операции над объектами данных, используя процедуры других программ).

Сетевые службы, обеспечивающие функционирование ИС, в том числе:

- DHCP и BOOTP, обеспечивающие инициализацию рабочих станций;
- DNS, обеспечивающие трансляцию имен в адреса и наоборот;
- AAA и Radius, обеспечивающие единую аутентификацию в сети, авторизацию, регистрацию и учет выполнения политик доступа;
- VPN, обеспечивающие туннелирование.

Информационные службы (NTP, от англ. network time protocol, протокол сетевого времени).

Файловые серверы (FTP, TFTP, SFTP).

Серверы *доступа к данным* (LDAP, от англ. lightweight directory access protocol и SQL, от англ. structured query language).

Службы *обмена сообщениями* (электронной почты, чатов, новостей).

Серверы удаленного доступа, обеспечивающие пользователя аналогом локального терминала для работы на удаленной системе (telnet, SSH и др.).

Перечисленные серверы имеют уязвимости, чем обусловливают возможности злоумышленника по реализации угроз СР. Далее в работе будут рассмотрены клиент-серверные системы без детализации функций прикладного ПО, т. е. универсальные, так как управление параметрами соединений доступно во всех таких системах, функционирующих в семействе протоколов TCP/IP.

В процессе функционирования клиент-серверной ИС инициатор соединения (клиент) формирует запросы к серверу, который обрабатывает их в условиях ограниченного вычислительного ресурса. Ограниченность вычислительного ресурса выражается в том, что сервер способен обработать ограниченное количество пакетов сообщений за единицу времени без переполнения буфера обмена или же снижения качества обслуживания заявок.

Задача сервера – своевременно обслужить максимальное количество запросов санкционированных клиентов с различными приоритетами.

Разработка модели функционирования клиент-серверной ИС необходима для описания существенных свойств процессов динамической конфигурации параметров сетевых соединений ИС, что необходимо для разработки алгоритма динамической конфигурации параметров сетевых соединений ИС в условиях СР.

Анализ объекта исследования

В общем случае ИС представляет собой совокупность ЭВМ, периферийного и коммуникационного оборудования, объединенного физическими линиями связи. Все эти элементы определяются идентификаторами, в качестве которых в наиболее распространенном семействе протоколов TCP/IP используются сетевые адреса (IP-адреса).

Для передачи информации между удаленными ИС, а также между клиентами и серверами в ИС с клиент-серверной архитектурой посредством протоколов взаимодействия устанавливают логическое соединение, под которым понимают инициализацию (передача пакета с установленным флагом SYN) запросов на обслуживание (информационных потоков) от клиента к серверу, получение параметров соединения и поддержание соединения между клиентом и сервером до его окончания.

Увеличение интенсивности поступающих информационных потоков (ИП) с одного IP-адреса или множества IP-адресов, может привести к реализации атаки типа «отказ в обслуживании» (Denial of Service, DoS) или же распределенной атаки типа «отказ в обслуживании» (Distributed Denial of Service, DDoS) соответственно [35-39].

Механизмы для управления ИП обеспечивает протокол TCP сети Интернет [40]. Управление ИП позволяет поддерживать надежность передачи по протоколу TCP путем регулировки скорости ИП между отправителем и получателем TCP-пакетов сообщений в течение определенного сеанса. Управление ИП осуществляется путем ограничения количества сегментов данных, передаваемых за один раз, а также запроса подтверждений получения до отправки следующих сегментов.

Для управления ИП протокол TCP в первую очередь определяет количество сегментов данных, которое может принять получатель TCP-пакетов сообщений (сервер). Таким образом, одним из основных параметров взаимодействия клиента и сервера является 16-битное поле «размер окна» TCP-заголовка (рис. 1) ответного (от сервера) пакета сообщений, которое показывает клиенту готовность принять для ведения информационного обмена определенный объем данных (количество байтов). После согласования отправитель TCP-пакетов сообщений (клиент) должен ограничить количество сегментов данных, отправленных получателю пакетов сообщений (серверу), в соответствии со значением поля «размер окна». Только после того как отправитель TCP-пакетов сообщений (клиент) получит подтверждение того, что сегменты данных получены, он может продолжить отправку остальных данных в этом сеансе.

Соединение осуществляется (рис. 2) по инициативе отправителя пакетов сообщений [40]. При необходимости выполнить обмен данными с получателем пакетов сообщений приложение-клиент обращается к нижележащему протоколу TCP, который в ответ на это посылает сегмент-запрос на установление соединения протоколу TCP, работающему на стороне отправителя пакетов сообщений, в числе прочего в запросе содержится флаг SYN, установленный в «1». Получив запрос, сервер выделяет определенные системные ресурсы, устанавливая начальное значение W_N поля «размер окна» (например, 25 байт) для фор-

мирования TCP-заголовка ответного пакета сообщений, объявляя отправителю пакетов сообщений о своей готовности получить небольшой, но достаточный для осуществления последующего информационного обмена объем данных, а также другие переменные соединения.

Переменные соединения, например, такие как максимальный размер сегмента (MSS, Maximum Segment Size) или заполнение заголовка TCP (Padding) [40]. После того как на стороне сервера все необходимые действия выполнены, ресурсы определены, модуль TCP посылает клиенту сегмент с флагами ACK и SYN с установленным W_N отправителю. В ответ клиент передает сегмент с флагом ACK и переходит в состояние установленного логического соединения.

Рис. 1. Структура TCP-заголовка пакета сообщений

В те периоды времени, когда ИС или ресурсы сервера (получателя TCP-пакетов сообщений) перегружены, длительность задержки может увеличиться. То есть в случае отсутствия у сервера необходимых ресурсов вычислительной мощности, сервер может приостановить информационный обмен с клиентом. Для этого сервер устанавливает значение TCP-буфера путем установления поля «размер окна» в TCP-заголовке пакета сообщений равным нулю $W_U = 0$, инициализируя тем самым механизм удержания в двухстороннем порядке соединения с отправителем пакетов сообщений, и направляет ему соответствующие пакеты сообщений. Инициативное снижение скорости передачи данных при каждом сеансе помогает уменьшить конфликт ресурсов отправителя и получателя TCP-пакетов сообщений в случае инициализации нескольких сеансов связи. Этим достигают уменьшения потерь данных и количества их повторных пересылок.

Получив пакет сообщений с $W_U = 0$, в соответствии со спецификацией протокола TCP [40], отправитель пакетов сообщений будет периодически посылать пробные однобайтовые сегменты, запрашивая получателя пакетов сообщений повторить информацию о размере окна и ожидаемом следующем байте (так называемый пробный сегмент «zero-window probe»), чтобы определить, когда он

сможет возобновить отправку данных. Сервер, реализуя механизм удержания в двухстороннем порядке соединения с отправителем пакетов сообщений, может не увеличивать окно, оставляя его равным нулю, тем самым удерживая отправителя пакетов сообщений заблокированным в продолжительном соединении на время, пока не истечет время тайм-аута [11-13, 15, 16, 18, 35].

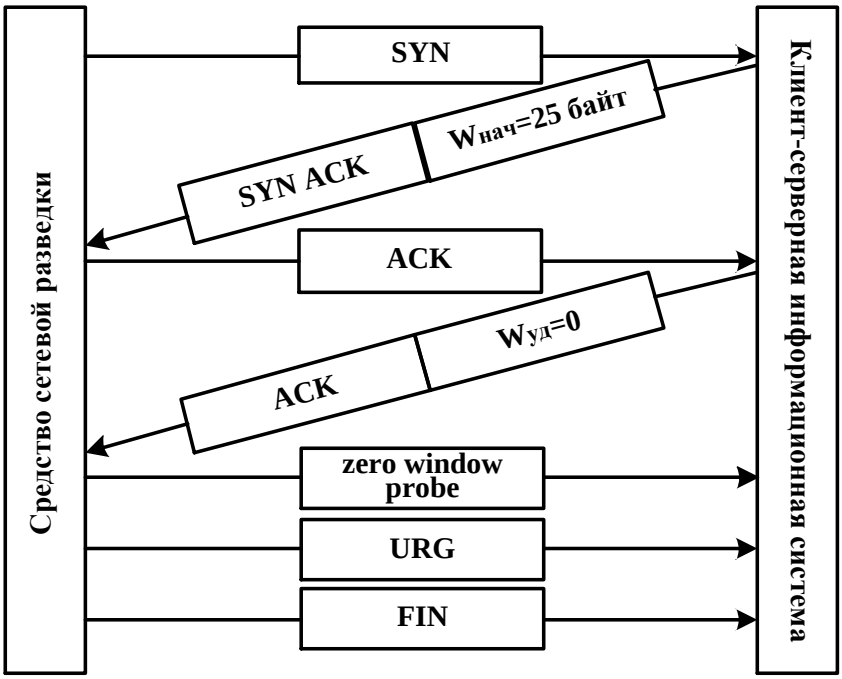

Рис. 2. Иллюстрация последовательности установки TCP-соединения и удержания соединения ИС (сервера) с отправителем пакетов сообщений (средством СР)

Если отправитель пакетов сообщений захочет разорвать соединение (отправка пакета сообщений с флагом FIN в TCP-заголовке) или выслать срочные данные (отправка пакета сообщений с флагом URG в TCP-заголовке), то получатель пакетов сообщений может игнорировать эти входящие пакеты сообщений, блокируя их. Сервер, игнорируя эти пакеты, вынуждает операционную систему клиента поддерживать ресурсы соединения до истечения состояния FIN-WAIT-1, ожидая TCP-сегмент от сервера с подтверждением о готовности закрыть соединение.

Сервер (получатель пакетов сообщений) не поддерживает состояние соединения со своей стороны и свой вычислительный ресурс не расходует, что позволяет ему в полной мере реализовать функции обработки поступающих пакетов сообщений от клиентов с более высоким приоритетом.

Для подтверждения получения от клиента сегмента данных (или совокупности сегментов) сервером направляют клиенту ACK-сообщения. В процессе информационного обмена по каналам связи возможны нарушения порядка доставки сегментов и потери сегментов, что вызывает необходимость их повторной передачи от клиента к серверу.

В частности, при наличии помех в канале связи, возможно направление трех дубликатов подтверждения ACK на каждый из полученных фрагментов TCP-пакета сообщений, что представляет собой использование алгоритма контроля насыщения – алгоритма быстрого повтора (Fast Retransmit) для протокола TCP [16, 41].

Алгоритм быстрого повтора для протокола TCP основан на том, что получателю TCP-пакетов сообщений (серверу) следует незамедлительно передавать дубликат ACK при получении сегмента с нарушением порядка доставки. Это делается для того, чтобы с помощью подтверждения ACK информировать отправителя TCP-пакетов сообщений о том, что сегмент был получен с нарушением порядка и указать порядковый номер ожидаемого сегмента.

С точки зрения клиента (отправителя TCP-пакетов сообщений) дубликат ACK может быть вызван различными сбоями в сети. Во-первых, причиной может служить отбрасывание сегментов. В этом случае все сегменты после отброшенного будут порождать дубликаты ACK. Во-вторых, дубликаты ACK могут быть обусловлены нарушением порядка доставки сегментов (например, при доставке по разным маршрутам). Наконец, причиной появления дубликатов ACK может быть репликация пакетов ACK или сегментов данных в сети.

Серверу (получателю TCP-пакетов сообщений) следует незамедлительно передавать подтверждение ACK при получении сегмента, который полностью или частично заполняет пропуски в порядковых номерах. Это позволит предоставить своевременную информацию отправителю TCP-пакетов сообщений, выполняющему восстановление после потери с использованием тайм-аута повторной передачи (retransmission timeout), быстрого повтора (fast retransmit) или улучшенного алгоритма восстановления (loss recovery).

Клиенту (отправителю TCP-пакетов сообщений) следует использовать алгоритм быстрого повтора для детектирования потери и исправления ошибки с использованием входящих дубликатов ACK. Алгоритм быстрого повтора использует прибытие трех дубликатов ACK, без каких-либо промежуточных сегментов ACK, как индикацию потери сегмента. После получения трех дубликатов ACK протокол TCP выполняет повторную передачу сегмента, который считается потерянным, без ожидания завершения отсчета таймера повтора передачи, предусмотренного спецификацией протокола TCP [40].

Таким образом, стратегия сервера заключается в оптимальном распределении своего ресурса для обеспечения своевременности обработки запросов клиентов с учетом их приоритетов, чего можно достичь динамическим управлением параметрами соединения.

Постановка задачи

Приведенное описание процесса функционирования клиент-серверной ИС (вербальная модель) позволяет формализовать задачу исследования.

При формализации задачи исследования необходимо [42] используя математическую запись сформулировать суть решаемой задачи, критерий ее решения, входные и выходные данные, существенные факторы и условия задачи.

Для формальной постановки и решения задачи в работе введены обозначения, представленные в таблице 1.

Таблица 1 – Обозначения и переменные, принятые для формальной постановки и решения задачи

Обозначение	Физический смысл обозначения
$\bar{T}_D^C$	– среднее время нахождения клиента в простое в связи с занятостью сервера
K_A	– коэффициент доступности сервера или клиента
$\bar{T}_D$	– средняя длительность промежутка времени, когда клиентам ИС недоступны услуги сервера с требуемым качеством (среднее время простоя), или простоя клиента
$\bar{T}$	– среднее время общей работы сервера (клиента)
S	– клиент-серверная ИС
C	– множество входных параметров модели, параметры контроля насыщения соединения
W	– размер окна (*Window Size*)
A	– подтверждение получения сегмента данных, передают для каждого сегмента, размер которого задан параметром W (*Acknowlegment*)
P_i	– множество выходных параметров модели, значения финальных вероятностей состояний системы S
Z	– множество внутренних параметров модели
I	– множество параметров условий функционирования
Q	– показатель эффективности функционирования клиент-серверной ИС
μ	– модель клиент-серверной ИС S

Своевременность обработки можно выразить минимизацией среднего времени нахождения клиента в простое в связи с занятостью сервера, $\bar{T}_D^C \to \min$. Тогда доступность сервера и (или) клиента можно выразить через коэффициент его доступности $K_A \to \max$ (исправного действия), вычисляемый по формуле

$$K_A = \frac{\bar{T} - \bar{T}_D}{\bar{T}} \cdot 100\%, \quad (1)$$

где: $\bar{T}_D$ – средняя длительность промежутка времени, когда клиентам ИС недоступны услуги сервера с требуемым качеством (среднее время простоя), или простоя клиента; $\bar{T}$ – среднее время общей работы сервера (клиента). Обратным по смыслу коэффициентом $K_D \to \min$ является коэффициент простоя:

$$K_D = \frac{\bar{T} - \bar{T}_A}{\bar{T}} \cdot 100\%, \quad (2)$$

где: $\bar{T}_A$ – средняя длительность промежутка времени, когда клиентам ИС доступны услуги сервера с требуемым качеством (среднее время исправного действия), или исправного действия клиента.

Воздействие на сервер (клиента) ИС случайных и преднамеренных помех создает дополнительную (нештатную) нагрузку на процессы связи и устройства, их реализующие. В результате $\bar{T}_D$ – длительность промежутка времени, когда абонентам недоступны от сервера ИС услуги с требуемым качеством

(время простоя) – увеличивается, а показатель доступности сервера (клиента) ИС – уменьшается.

Моменты возможных переходов клиент-серверной ИС из состояния в состояние неопределенны, случайны и происходят под действием потоков событий, характеризующиеся их интенсивностями, являющимися важной характеристикой потоков событий и характеризующими среднее число событий, приходящееся на единицу времени. Рассмотрим переход от детерминированной постановки задачи к постановке задачи в условиях неопределенности. Тогда финальную вероятность состояния S_i клиент-серверной системы можно будет интерпретировать как среднее относительное время пребывания клиент-серверной системы в этом состоянии.

Дано:

S – клиент-серверная ИС;

C – множество входных параметров модели, параметры контроля насыщения соединения, $C \subseteq \{W, A\}$, где W = [0, 1, …, 65535] – размер окна в байтах,

A = [0, 1, 2, 3] – подтверждение получения сегмента данных, передают для каждого сегмента, размер которого задан параметром W;

P_i – множество выходных параметров модели, значения финальных вероятностей состояний системы S, $P_i = \lim\limits_{t\to\infty} P_i(t)$, где i = 1, 2,…, h , причем число состояний конечно и из каждого из них можно за конечное число шагов перейти в любое другое;

Z – множество внутренних параметров модели $Z \subseteq \{S_i, \Lambda_j\}$, где $S_i = \{S_1, \ldots, S_h\}$, $\Lambda_j = \{\lambda_1, \lambda_2, \ldots, \lambda_J\}$, перечень моделируемых состояний системы и интенсивностей потоков событий в ней описаны ниже по тексту;

I – множество параметров условий функционирования, где I – протоколы транспортного уровня семейства протоколов *TCP/IP*, поддерживаемые моделируемой системой, условие допустимости $I \subseteq \{TCP, UDP\}$;

Q – показатель эффективности функционирования клиент-серверной ИС, $Q = \lim\limits_{t\to\infty} P_D^C(t)$, $P_D^C(t) \to \min$, определяемый простоем клиента.

Найти: закономерность изменения множества P_i выходных параметров модели функционирования клиент-серверной ИС и множества Q показателей эффективности функционирования клиент-серверной ИС от множества C значений входных параметров, множества Z значений внутренних параметров, множества I значений параметров условий функционирования. На значения параметров множеств C, P_i, Z, I наложены условия их допустимости.

Тогда формальная постановка задачи на моделирование клиент-серверной ИС:

$$\mu : < S, C, Z, I> \to P_i, Q \mid C \subseteq \{W, A\}, P_i = \lim_{t\to\infty} P_i(t), I \subseteq \{TCP, UDP\},$$

а формальная постановка задачи на оптимизацию показателей эффективности клиент-серверной ИС:

$$< S, C, Z, I> \to \min P_D^C \mid P_D^C \in \{P_i\}, i = 1, 2, \ldots, h$$

для минимизации вероятности простоя клиента и сервера.

$$< S, C, Z, I> \rightarrow \max P_D^{NI} \mid P_D^{NI} \in \{P_i\},\ i = 1, 2,\ldots, h.$$

для максимизации вероятности простоя средства СР.

Потоки событий (и отказов) от клиента к серверу и от сервера к клиенту представляют собой последовательность управления параметрами соединения, приводящими к обеспечению своевременности связи с клиентами с наивысшим приоритетом, к изменению коэффициента простоя клиентов и сервера в соответствии с их приоритетами (наименьшим приоритетом, понятно, будут обладать клиенты или клиенты, пытающиеся осуществлять подключения к серверу с высокой интенсивностью заявок, способной перегрузить его).

Модель функционирования клиент-серверной информационной системы

Пусть имеется узел ИС – сервер, обеспечивающий функционирование клиент-серверной системы, в том числе и в части системы контроля (оценки) значения показателя простоя. Моделируемая система S с течением времени меняет свое состояние (переходит из одного состояния в другое). Необходимые для исследования состояния клиент-серверной ИС S_1, S_2, … можно перечислить так, как представлено в таблице 2.

Таблица 2 – Дискретные состояния клиент-серверной ИС

Состояние	Описание состояния
S_1	Клиент находится в состоянии простоя $P_D^C \rightarrow \max$, не принимает и не передает пакеты сообщений
S_2	Инициализация соединения клиентом
S_3	Оценка значения показателя простоя клиента
S_4	Установление (изменение) скорости потока данных между клиентом и сервером установлением (изменением) параметра «размер окна» W
S_5	Установление (подтверждение) соединения сервером и получение клиентом параметра «размер окна» W
S_6	Передача и прием потоков данных между клиентом и сервером
S_7	Подтверждение сервером приема частей потока данных (квитирование)

Моменты возможных переходов клиент-серверной ИС из состояния в состояние неопределенны, случайны и происходят под действием потоков событий, характеризующиеся их интенсивностями λ, представленными в таблице 3, являющимися важной характеристикой потоков событий и характеризующими среднее число событий, приходящееся на единицу времени.

Граф состояний функционирования клиент-серверной ИС представлен на рис. 3.

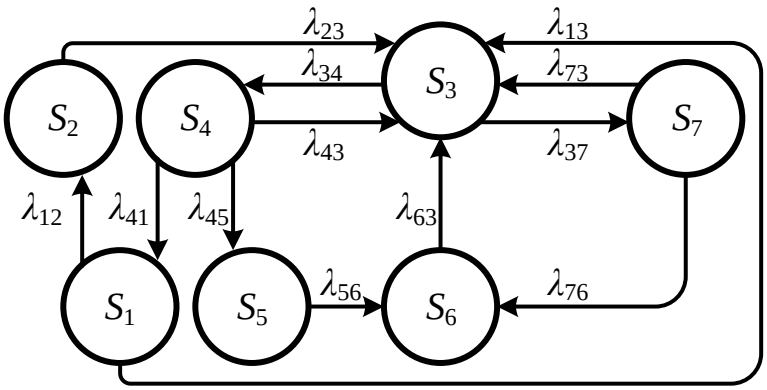

Рис.3. Граф состояний функционирования клиент-серверной ИС

Оценка эффективности процессов функционирования ИС связана с необходимостью моделирования процесса в реальном времени, что обусловливает целесообразность использования математического аппарата марковских случайных процессов, необходимое условия которого – потоки событий являются простейшими (обладают свойствами стационарности, ординарности и не имеют последействий). Таким образом, процесс функционирования клиент-серверной ИС можно представить, как марковский случайный процесс с дискретными состояниями и непрерывным временем.

Таблица 3 – Интенсивности потоков событий в клиент-серверной ИС

Интенсивность	Описание интенсивности потоков событий
λ_{12}	Заявки на инициализацию (нового) соединения клиентом
λ_{23}	Заявки на оценку значения показателя простоя после инициализации соединения клиентом
λ_{34}	Заявки на установление (изменение) скорости потока данных между клиентом и сервером установлением (изменением) параметра «размер окна» W
λ_{43}	Заявки на оценку значения показателя простоя после установления (изменения) параметра «размер окна» W
λ_{45}	Заявки на установление (подтверждение) соединения сервером и получение клиентом параметра «размер окна» W
λ_{56}	Заявки на передачу и прием потоков данных между клиентом и сервером после подтверждения сервера установки соединения (передачи им W_N и SYN ACK)
λ_{63}	Заявки на оценку значения показателя простоя в процессе передачи и приема потоков данных между клиентом и сервером
λ_{37}	Заявки на подтверждение сервером приема частей потока данных (квитирование)
λ_{73}	Заявки на оценку значения коэффициента простоя K_D после подтверждение сервером приема частей потока данных (квитирование)
λ_{76}	Заявки на передачу клиентом очередной части потока данных после подтверждения приема части потока
λ_{41}	Заявки на максимизацию значения показателя простоя клиента (параметров 0-скорости потока данных) установлением (изменением) параметра «размер окна» $W = 0$
λ_{13}	Заявки на оценку значения показателя простоя клиента

Рассмотрим сценарий перехода моделируемой системы из состояния S_i в состояние S_j под воздействием потоков событий с интенсивностями λ_{ij}.

При функционировании клиент-серверной ИС возникают объективные ограничения на производительность как ИС в целом, так и ее элементов. При возникновении клиентов с разным уровнем приоритетов ситуация усугубляется. В этой связи в таких ИС используют диспетчеризацию запросов клиентов к серверу. В случае необходимости обработку запросов пользователей с более низким приоритетом приостанавливают, не разрывая с ними соединения. Это рационально, так как повторное установление соединения вызывает повтор технологических операций, связанных с ним, что отрицательно влияет на производительность ИС.

Пусть S_1 – начальное состояние моделируемой клиент-серверной ИС, в котором она не принимает и не передает потоки данных, то есть состояние покоя, что характеризуется для клиента высоким значением показателя простоя, оценка которого осуществляется в состоянии S_3 (по заявке λ_{13}). В это состояние S_1 также целесообразно в пределе перевести клиентов, заявки, полученные от которых, способны перегрузить сервер так, чтобы у них не было ресурса для перехода в состояние S_2 инициализацией заявок λ_{12} на соединение с сервером. Клиенты могут инициализировать альтернативные заявки λ_{12} вплоть до исчерпания ресурса системы, которое наступит, если предыдущие потоки данных не будут закрыты. Если такой ресурс все еще имеется – система S переходит в состояние S_2 и инициализирует соединение с сервером по передаче пакетов сообщений с установленным флагом SYN. Аналогичное событие наступает в исследуемой системе S при появлении в ней новых (альтернативных) санкционированных клиентов или новых заявок от уже подключенных клиентов, но по другому протоколу (организация нового сокета). Тогда в моделируемой системе возникают заявки λ_{23} на оценку значения показателя простоя S_3 после инициализации соединения клиентом.

После оценки значения показателя простоя в системе S возникают заявки λ_{34} на установление (изменение) скорости потока данных между клиентом и сервером S_4 установлением (изменением) параметра «размер окна» W. Значение этого параметра выбирается в соответствии со значением показателя простоя исследуемой ИС: если инициализация соединения осуществляется клиентами с низким или обычным количеством заявок, а производительность сервера имеет ограниченный ресурс, то устанавливают значение параметра «размер окна» W в некоторое ненулевое значение W_N, например, $W_N = 20$ байт. В противном случае, то есть если инициализация соединения осуществляется клиентами с большим количеством заявок, способным перегрузить сервер, имеется возможность установить значение параметра «размер окна» W в нулевое значение, $W_U = 0$ байт. В результате этого в исследуемой системе S возникают заявки λ_{41} на максимизацию значения показателя простоя. Заявки λ_{43} на оценку значения показателя простоя после установления (изменения) параметра «размер окна» W позволяют динамически изменять (регулировать) скорость потока данных от клиентов к серверу. Если устанавливается ненулевое значение W_N, то в исследуемой системе S возникают заявки λ_{45} на установление (подтверждение) со-

единения сервером SYN ACK и получение клиентами параметра «размер окна» W после чего система переходит в состояние S_5. Это состояние, в случае наличия у клиентов (сервера) данных для передачи, вызывает возникновение заявок λ_{56} на передачу и прием потоков данных между клиентами и сервером. В результате чего исследуемая система S переходит в состояние S_6 передачи и приема потоков данных между клиентами и сервером, в процессе которого возникают заявки λ_{63} на оценку значения показателя простоя при передаче и приеме потоков данных между клиентами и сервером. В процессе передачи и приема потоков данных клиенты и сервер обмениваются квитанциями (подтверждениями) [41] – состояние S_7 по заявкам λ_{37}. Порядок квитирования также влияет на значение показателя простоя, что отражено на графе заявками λ_{73} на его оценку.

После получения очередной квитанции λ_{76} от сервера клиенты передают ему следующую часть потока данных. В том случае, если в результате воздействия на канал связи и сервер преднамеренных и (или) непреднамеренных помех какие-либо части потока данных уничтожаются в процессе передачи, или приходят от клиентов к серверу в неверном порядке, то скорость передачи данных в клиент-серверной ИС снижается. В этом случае нецелесообразно говорить о простое системы S или клиентов, поэтому соответствующей связи между состояниями S_7 и S_1 на графе состояний нет.

По полученному размеченному графу состояний клиент-серверной ИС строится математическая модель ее функционирования – дифференциальные уравнения с неизвестными функциями $p_i(t)$:

$$
\begin{cases}
\dfrac{dp_1(t)}{dt} = \lambda_{41} p_4(t) - \lambda_{12} p_1(t) - \lambda_{13} p_1(t), \\
\dfrac{dp_2(t)}{dt} = \lambda_{12} p_1(t) - \lambda_{23} p_2(t), \\
\dfrac{dp_3(t)}{dt} = \lambda_{23} p_2(t) + \lambda_{43} p_4(t) + \lambda_{63} p_6(t) + \\
+\lambda_{73} p_7(t) + \lambda_{13} p_1(t) - (\lambda_{34} + \lambda_{37}) p_3(t), \\
\dfrac{dp_4(t)}{dt} = \lambda_{34} p_3(t) - (\lambda_{41} + \lambda_{43} + \lambda_{45}) p_4(t), \\
\dfrac{dp_5(t)}{dt} = \lambda_{45} p_4(t) - \lambda_{56} p_5(t), \\
\dfrac{dp_6(t)}{dt} = \lambda_{56} p_5(t) + \lambda_{76} p_7(t) - \lambda_{63} p_6(t), \\
\dfrac{dp_7(t)}{dt} = \lambda_{37} p_3(t) - (\lambda_{73} + \lambda_{76}) p_7(t), \\
\sum_{i=1}^{7} p_i(t) = 1.
\end{cases}
\qquad (3)
$$

DOI: 10.24411/2410-9916-2019-10403
URL: https://sccs.intelgr.com/archive/2019-04/03-Maximov.pdf

Для решения дифференциальных уравнений Колмогорова задаются начальные условия. Вектор вероятностей начальных состояний марковской цепи с учетом отсутствия воздействий на клиент-серверную ИС в начальный момент времени имеет вид:

$$p(0)=|1\ 0\ 0\ 0\ 0\ 0\ 0|, \quad (4)$$

что соответствует высокому значению показателя простоя.

Задавая численные значения интенсивностей λ, представленных в таблице 3, и переходя к непрерывному времени $t \to \infty$, решается система линейных дифференциальных уравнений (3) с постоянными коэффициентами (однородный марковский процесс). Для любого момента времени *t* сумма всех вероятностей состояний равна единице:

$$\sum_{i=1}^{n} p_i(t) = 1. \quad (5)$$

Характер выбранных значений интенсивностей определяется в соответствии со стратегиями клиентов и сервера – сторон ресурсного конфликта.

Модель функционирования клиент-серверной ИС учитывает воздействия на сервер санкционированных клиентов с различными приоритетами запросов и количеством заявок.

Использование модели предполагает поиск стратегий взаимодействия сервера и клиентов ИС, и позволит перейти к вероятностной оценке простоя $P_D^{NI} \to \max$, $P_D^C \to \min$. Учет в марковской модели времени пребывания ИС в каждом из состояний в зависимости от стратегий взаимодействующих сторон позволяет исследовать динамику функционирования клиент-серверной ИС.

В качестве исходных данных для моделирования выступают:

- система линейных дифференциальных уравнений (3);
- вектор вероятностей начальных состояний (4);
- значения интенсивностей потоков событий, представленные в таблице 3;
- нормировочное условие (5).

Недостатки методов Эйлера и других численных методов решения более высоких порядков [43] заключающиеся в необходимости вычисления на каждом шаге частных производных функции $S(t, p)$, что приводит к большой вычислительной сложности, предопределили выбор в качестве метода решения системы ЛДУ классический метод четвертого порядка – метод Рунге-Кутты с фиксированным шагом интегрирования, имеющий вид (6), где *h* – приращение, соответствующее шаговой поправке Эйлера, Δp_i – средневзвешенная величина поправок $h\eta_1^i, h\eta_2^i, h\eta_3^i, h\eta_4^i$ каждого этапа интегрирования (с весовыми коэффициентами $1/6, 2/6, 2/6, 1/6$ соответственно), то есть результат усреднения с указанными коэффициентами четырех этапных поправок.

$$\begin{cases} \eta_1^i = S\left(t_i,\, p_i\right), \\ \eta_2^i = S\left(t_i + \dfrac{h}{2},\, p_i + \dfrac{h}{2}\eta_1^i\right), \\ \eta_3^i = S\left(t_i + \dfrac{h}{2},\, p_i + \dfrac{h}{2}\eta_2^i\right), \\ \eta_4^i = S\left(t_i + h,\, p_i + h\eta_3^i\right), \\ \triangle p_i = \dfrac{h}{6}\left(\eta_1^i + 2\eta_2^i + 2\eta_3^i + \eta_4^i\right), \\ p_{i+1} = p_i + \triangle p_i. \end{cases} \quad (6)$$

Приводим систему (6) к векторному представлению – столбец D, где каждый элемент соответствует правой части определенного дифференциального уравнения в системе:

$$D\left(t,\, p\right) = \begin{pmatrix} \lambda_{41}p_4\left(t\right) - \lambda_{12}p_1\left(t\right) - \lambda_{13}p_1\left(t\right) \\ \lambda_{12}p_1\left(t\right) - \lambda_{23}p_2\left(t\right) \\ \lambda_{23}p_2\left(t\right) + \lambda_{43}p_4\left(t\right) + \lambda_{63}p_6\left(t\right) + \\ +\lambda_{73}p_7\left(t\right) + \lambda_{13}p_1\left(t\right) - \left(\lambda_{34} + \lambda_{37}\right)p_3\left(t\right) \\ \lambda_{34}p_3\left(t\right) - \left(\lambda_{41} + \lambda_{43} + \lambda_{45}\right)p_4\left(t\right) \\ \lambda_{45}p_4\left(t\right) - \lambda_{56}p_5\left(t\right) \\ \lambda_{56}p_5\left(t\right) + \lambda_{76}p_7\left(t\right) - \lambda_{63}p_6\left(t\right) \\ \lambda_{37}p_3\left(t\right) - \left(\lambda_{73} + \lambda_{76}\right)p_7\left(t\right) \end{pmatrix} \quad (7)$$

Использование известного порядка решения системы ЛДУ методом Рунге-Кутты (7) позволяет получить числовую таблицу приближенных значений p_i искомых решений $p(t)$ на некотором интервале $t \in \left[t_0,\, t_1\right]$, как показано в таблице 4.

Таблица 4 – Приближенные решения $p(t)$ на заданных интервалах времени

Этапы интегрирования, n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	t_0	$p_1(t_0)$	$p_2(t_0)$	$p_3(t_0)$	$p_4(t_0)$	$p_5(t_0)$	$p_6(t_0)$	$p_7(t_0)$
…	…	…	…	…	…	…	…	…
n	t_1	$p_1(t_1)$	$p_2(t_1)$	$p_3(t_1)$	$p_4(t_1)$	$p_5(t_1)$	$p_6(t_1)$	$p_7(t_1)$

Таким образом, получают вероятностные и временные характеристики, описывающие состояния процесса функционирования клиент-серверной ИС, которые в свою очередь составляют основу для исследования данного процесса при различных стратегиях взаимодействующих сторон, как показано в таблице 5, что позволяет оценивать состояние клиент-серверной ИС.

Таблица 5 – Значения интенсивностей событий в зависимости от стратегий функционирования клиент-серверной ИС

Признаки	Стратегии			
	C_1	C_2	C_3	C_4
Наличие очереди, λ_{41}, λ_{45}	**max**	min	**max**	min
Наличие подтверждений, λ_{76}	min	min	**max**	**max**

Оценим устойчивость модели к вариациям исходных данных, задавая граничные значения в стратегиях взаимодействующих сторон, при этом рассмотрим следующие варианты стратегий:

- C_1 – без подтверждения и с очередью, в этом случае соединение клиентов с сервером осуществляется по протоколу UDP, где надежная передача данных и подтверждение их получения, в случае необходимости, должна реализовываться пользовательским приложением, сервер получив значительное множество заявок на соединение от клиентов выстраивает их в очередь и далее последовательно обрабатывает;
- C_2 – без подтверждения и без очереди, в этом случае соединение клиентов с сервером осуществляется по протоколу UDP, где надежная передача данных и подтверждение их получения, в случае необходимости, должна реализовываться пользовательским приложением, сервер получая заявки на соединение от клиентов успевает их обработать без задержки (без необходимости создания очереди из заявок) или отбрасывает;
- C_3 – с подтверждением и с очередью, в этом случае соединение клиентов с сервером осуществляется по протоколу TCP, где сервер, получив значительное множество заявок на соединение от клиентов, выстраивает их в очередь и далее последовательно обрабатывает;
- C_4 – с подтверждением и без очереди, в этом случае клиентов с сервером осуществляется по протоколу TCP, где сервер, получая заявки на соединение от клиентов, успевает их обработать без задержки (без необходимости создания очереди из заявок) или отбрасывает.

Значения интенсивностей потоков событий задаем постоянными, как показано в таблице 6, в соответствии с выбранной стратегией взаимодействия клиентов и сервера.

Графики зависимостей вероятностей состояний процесса функционирования клиент-серверной ИС от времени $p_1(t)$, $p_2(t)$, …, $p_7(t)$ для значений интенсивностей событий соответствующие стратегии C_1, в соответствии с таблицей 7, представлены на рис. 4.

Таблица 6 – Интенсивности потоков событий для каждой из стратегий взаимодействия сервера и клиентов

Интенсивность	λ	Значения λ для каждой из стратегий			
		C_1	C_2	C_3	C_4
Заявки на инициализацию (нового) соединения клиентом	λ_{12}	2	100	2	100
Заявки на оценку значения показателя простоя после инициализации соединения клиентом	λ_{23}	100	100	100	100
Заявки на установление (изменение) скорости потока данных между клиентом и сервером установлением (изменением) параметра «размер окна» W	λ_{34}	100	2	100	2
Заявки на оценку значения показателя простоя после установления (изменения) параметра «размер окна» W	λ_{43}	100	100	100	100
Заявки на установление (подтверждение) соединения сервером и получение клиентом параметра «размер окна» W	λ_{45}	100	100	100	100
Заявки на передачу и прием потоков данных между клиентом и сервером после подтверждения сервера установки соединения (передачи им W_N и SYN ACK)	λ_{56}	100	100	100	100
Заявки на оценку значения показателя простоя в процессе передачи и приема потоков данных между клиентом и сервером	λ_{63}	100	100	100	100
Заявки на подтверждение сервером приема частей потока данных (квитирование)	λ_{37}	2	2	100	100
Заявки на оценку значения коэффициента простоя K_D после подтверждение сервером приема частей потока данных (квитирование)	λ_{73}	100	100	100	100
Заявки на передачу клиентом очередной части потока данных после подтверждения приема части потока	λ_{76}	100	100	100	100
Заявки на максимизацию значения показателя простоя клиента (параметров 0-скорости потока данных) установлением (изменением) параметра «размер окна» $W = 0$	λ_{41}	100	2	100	2
Заявки на оценку значения показателя простоя клиента	λ_{13}	100	100	100	100

Таблица 7 – Числовая таблица приближенных значений $p_i(t)$ для λ стратегии C_1

Этапы интегрирования,n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	0	1	0	0	0	0	0	0
2	$1\cdot10^{-2}$	0,951	$9{,}278\cdot10^{-3}$	0,039	$8{,}956\cdot10^{-4}$	$2{,}969\cdot10^{-5}$	$2{,}041\cdot10^{-6}$	$3{,}652\cdot10^{-5}$
3	$2\cdot10^{-2}$	0,905	0,017	0,074	$3{,}216\cdot10^{-3}$	$2{,}148\cdot10^{-4}$	$1{,}975\cdot10^{-5}$	$1{,}336\cdot10^{-4}$
…	…	…	…	…	…	…	…	…
10^3	10	0,513	0,051	0,348	0,053	0,023	$9{,}069\cdot10^{-3}$	$2{,}547\cdot10^{-3}$

DOI: 10.24411/2410-9916-2019-10403
URL: https://sccs.intelgr.com/archive/2019-04/03-Maximov.pdf

№4. 2019
ISSN 2410-9916

На интервале времени [0; 0,09] ИС находится в переходном режиме функционирования, где наблюдается всплеск значений вероятности состояния $p_2(t)$ и $p_3(t)$ что соответствует нахождению ИС в состоянии инициализации соединения клиентами и оценки значения показателя простоя клиентов.

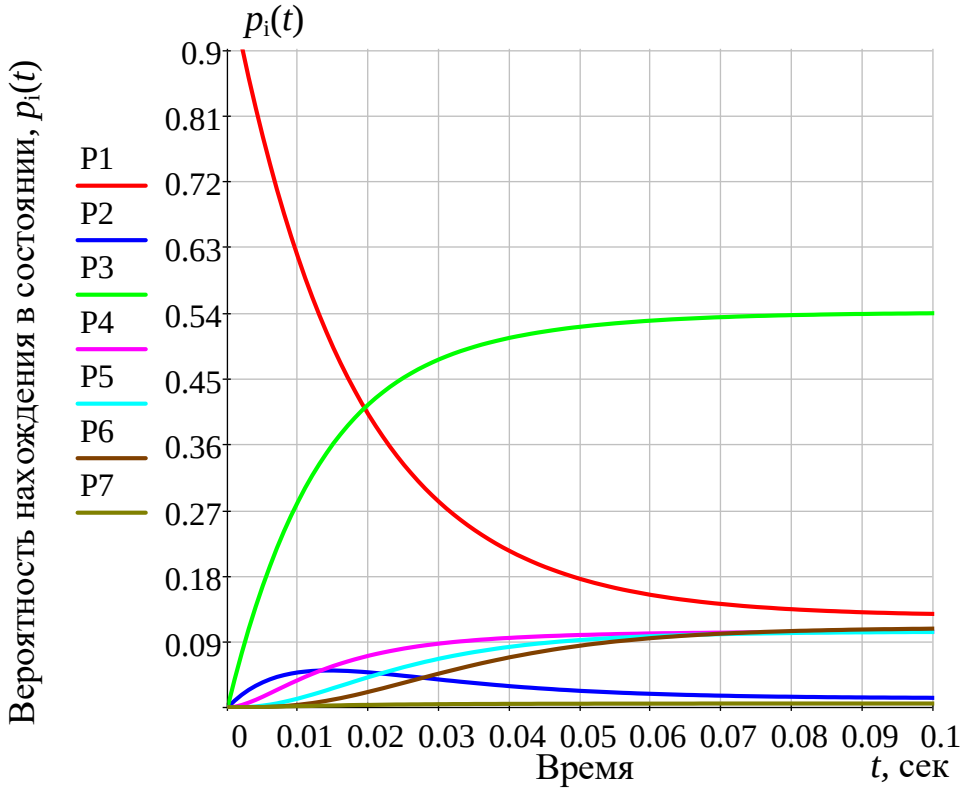

Рис. 4. Результаты расчета зависимости вероятностей состояний от времени для значений интенсивностей событий соответствующие стратегии C_1

При $t \to \infty$ в ИС устанавливается стационарный режим, когда ИС случайным образом меняет свои состояния и ее вероятности $p_1(t)$, $p_2(t)$, …, $p_7(t)$ уже не зависят от времени и равны финальным (предельным) вероятностям.

Полученные значения финальных вероятностей $p_1 = 0{,}513$, $p_2 = 0{,}051$, $p_3 = 0{,}348$, $p_4 = 0{,}053$, $p_5 = 0{,}023$, $p_6 = 9{,}069 \cdot 10^{-3}$, $p_7 = 2{,}547 \cdot 10^{-3}$ показывают, сколько времени ИС в среднем находится в каких состояниях.

Для исследования процесса функционирования и защиты ИС при перечисленных стратегиях функционирования клиент-серверной ИС, представленных в таблице 5, и соответствующих им значений интенсивностей событий производится расчет вероятностных и временных характеристик согласно вышеизложенному примеру.

Получаем числовую таблицу приближенных значений p_i на интервале $t \in [0,\ 10]$ с фиксированным шагом интегрирования 10^3, что представлено в таблице 8, для значений интенсивностей потоков событий стратегии C_2, которые приведены в таблице 6, сплайн-интерполяция значений которой показана на графиках зависимостей вероятностей состояний от времени (рис. 5).

DOI: 10.24411/2410-9916-2019-10403
URL: https://sccs.intelgr.com/archive/2019-04/03-Maximov.pdf

№4. 2019
ISSN 2410-9916

Таблица 8 – Числовая таблица приближенных значений $p_i(t)$ для λ стратегии C_2

Этапы интегрирования, n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	0	1	0	0	0	0	0	0
2	$1\cdot10^{-2}$	0,819	0,086	0,095	$4{,}508\cdot10^{-4}$	$1{,}494\cdot10^{-5}$	$3{,}407\cdot10^{-6}$	$9{,}022\cdot10^{-5}$
3	$2\cdot10^{-2}$	0,67	0,148	0,179	$1{,}628\cdot10^{-3}$	$1{,}085\cdot10^{-4}$	$2{,}721\cdot10^{-5}$	$3{,}26\cdot10^{-4}$
…	…	…	…	…	…	…	…	…
10^3	10	0,061	0,186	0,703	0,027	0,012	$6{,}442\cdot10^{-3}$	$5{,}372\cdot10^{-3}$

Приближенные значения p_i на интервале $t \in [0,10]$ с фиксированным шагом интегрирования 10^3 для значений интенсивностей потоков событий стратегии C_3, в соответствии с таблицей 6, представлены в числовой таблице 9, сплайн-интерполяция значений представлена на графиках зависимостей вероятностей состояний от времени (рис. 6).

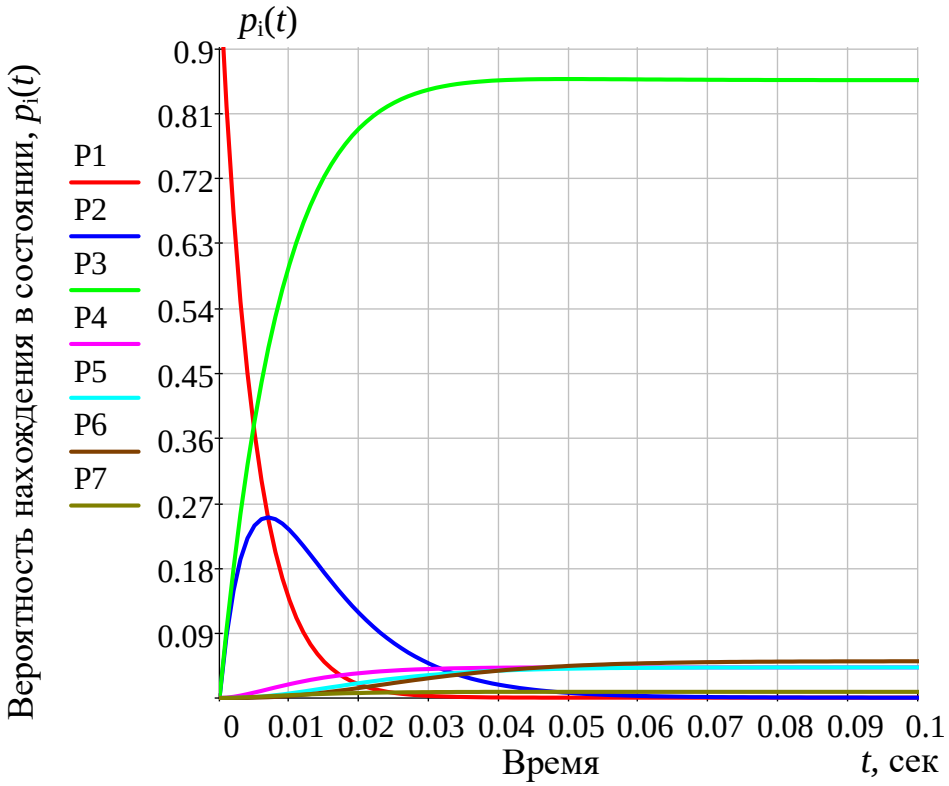

Рис. 5. Результаты расчета зависимости вероятностей состояний от времени для значений интенсивностей событий соответствующие стратегии C_2

Приближенные значения $p_i(t)$ для значений интенсивностей потоков событий стратегии C_4, в соответствии с таблицей 6, представлены в числовой таблице 10, сплайн-интерполяция значений представлена на графиках зависимостей вероятностей состояний от времени (рис. 7).

DOI: 10.24411/2410-9916-2019-10403
URL: https://sccs.intelgr.com/archive/2019-04/03-Maximov.pdf

Таблица 9 – Числовая таблица приближенных значений $p_i(t)$ для λ стратегии C_3

Этапы интегрирования, n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	0	1	0	0	0	0	0	0
2	$1\cdot10^{-2}$	0,896	$9{,}004\cdot10^{-3}$	0,086	$4{,}112\cdot10^{-3}$	$1{,}375\cdot10^{-4}$	$1{,}458\cdot10^{-4}$	$4{,}246\cdot10^{-3}$
3	$2\cdot10^{-2}$	0,808	0,016	0,15	0,014	$9{,}425\cdot10^{-4}$	$1{,}039\cdot10^{-3}$	0,014
…	…	…	…	…	…	…	…	…
10^3	10	0,267	0,034	0,316	0,1	0,054	0,092	0,138

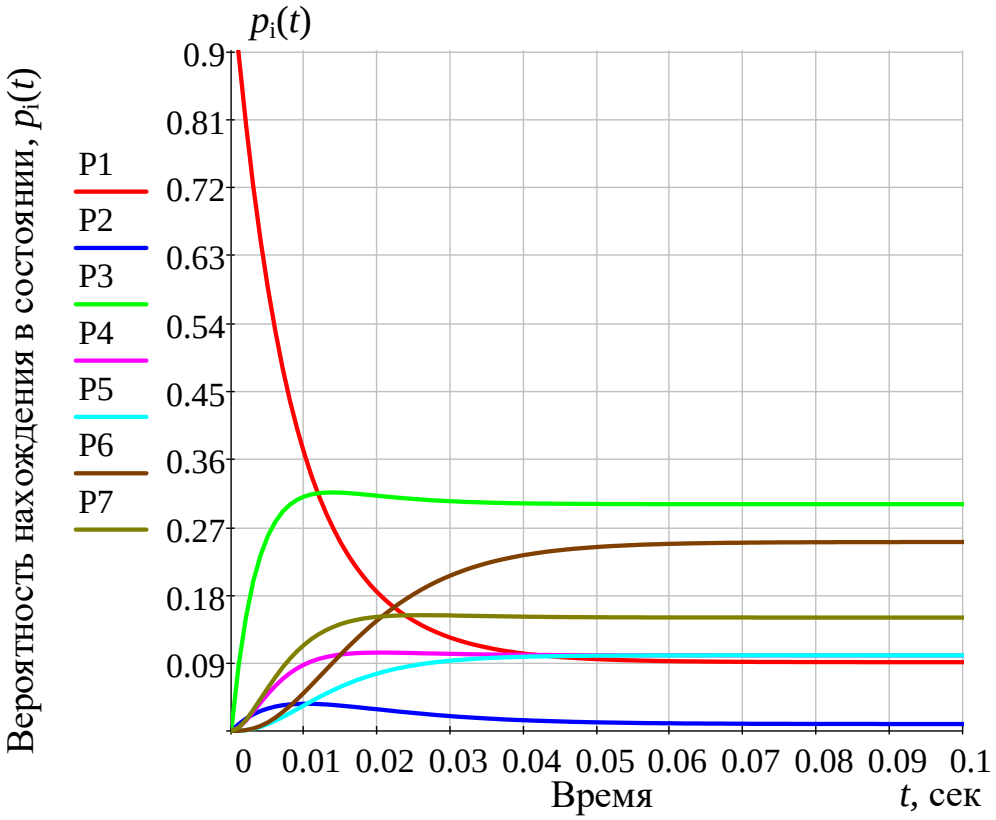

Рис. 6. Результаты расчета зависимости вероятностей состояний от времени для значений интенсивностей событий соответствующие стратегии C_3

Таблица 10 – Числовая таблица приближенных значений $p_i(t)$ для λ стратегии C_4

Этапы интегрирования, n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	0	1	0	0	0	0	0	0
2	$1\cdot10^{-2}$	0,861	0,044	0,091	$8{,}755\cdot10^{-5}$	$2{,}913\cdot10^{-6}$	$1{,}458\cdot10^{-4}$	$4{,}381\cdot10^{-3}$
3	$2\cdot10^{-2}$	0,741	0,078	0,165	$3{,}07\cdot10^{-4}$	$2{,}074\cdot10^{-5}$	$1{,}039\cdot10^{-3}$	0,015
…	…	…	…	…	…	…	…	…
10^3	10	0,122	0,124	0,465	$3{,}79\cdot10^{-3}$	$1{,}811\cdot10^{-3}$	0,092	0,191

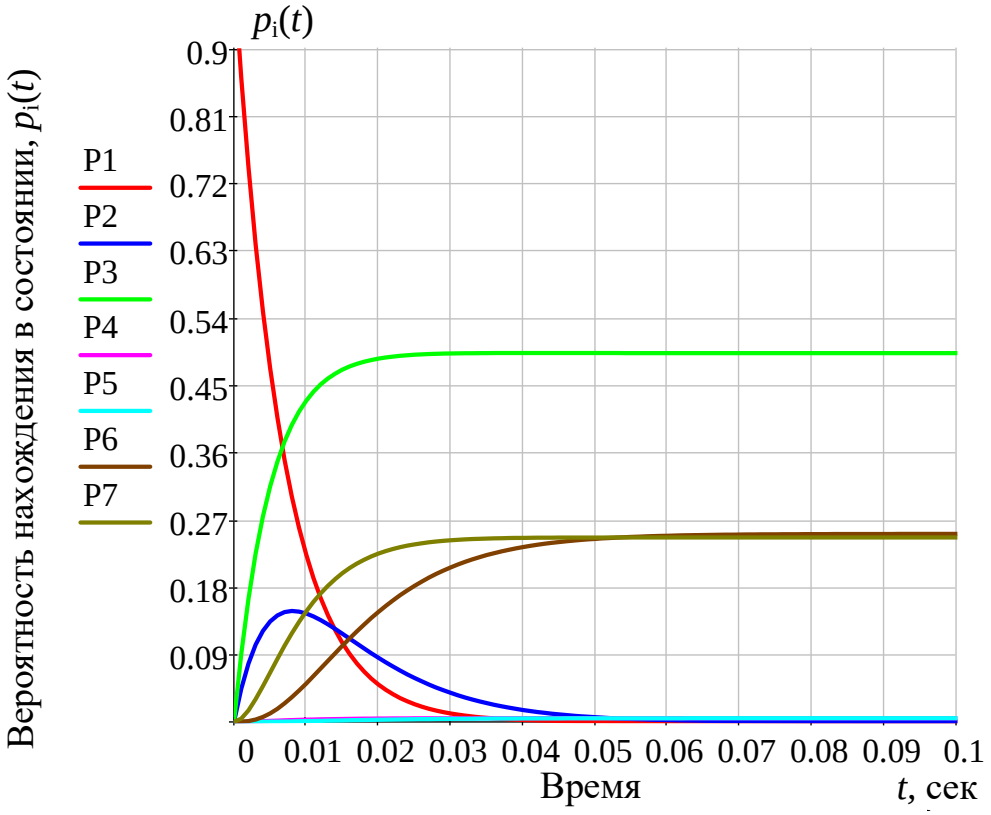

Рис. 7. Результаты расчета зависимости вероятностей состояний от времени для значений интенсивностей событий соответствующие стратегии C_4

Разработанная модель функционирования клиент-серверной ИС учитывает влияние и характер воздействия на ИС потоков событий от клиентов с низким и высоким приоритетом их обслуживания, а также с нормальным и высоким количеством заявок от клиентов к серверу, способных перегрузить его. Процесс защиты сервера от перегрузки в соответствии с данной моделью сводится к минимизации вероятности (и среднего времени) значения показателя простоя клиентов с высоким приоритетом обслуживания или высоким количеством заявок, и, следовательно, минимизации вероятности перегрузки сервера ИС $P_D^C \to \min$. Защита ИС от потоков событий от клиентов с высоким приоритетом их обслуживания или высоким количеством заявок предполагает поиск стратегий функционирования клиент-серверной ИС в зависимости от изменяющихся вариантов взаимодействия сторон из-за ограниченности ресурса сервера во времени. Модель позволяет вскрыть зависимости процесса функционирования клиент-серверной ИС от потоков воздействий, оценивать оперативность обслуживания клиентов, обоснованно выбирать алгоритмы защиты сервера от перегрузки и оптимально использовать ресурс сервера.

Увеличение интенсивностей заявок, как со стороны сервера, так и со стороны клиентов соответствует изменению стратегий взаимодействующих сторон. С увеличением λ_{12} (на рис. 8 до 1751) клиент-серверная ИС находится в затрудненном режиме работы, вероятность ее нахождения в состоянии S_1 равна

P_1=0,294, а $P_D^C \to \min$. В то же время вероятность перехода системы в состояние S_2 будет максимальной и равной P_2=0,611, для заданных значений интенсивностей из таблицы для стратегии C_3.

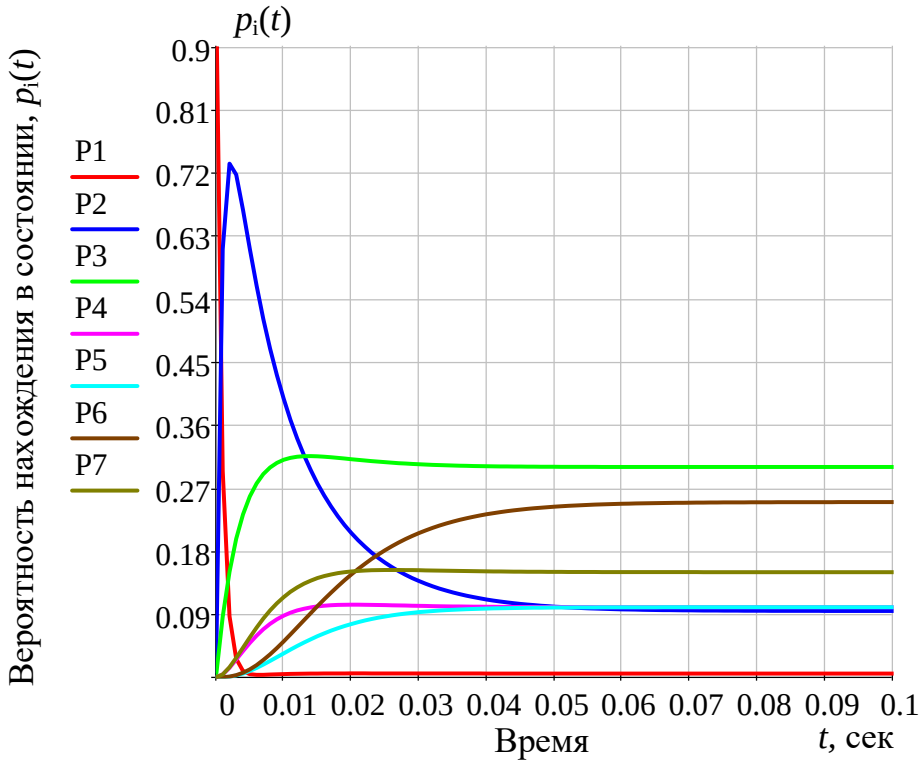

Рис. 8. Зависимости вероятностей состояний от времени для заданных значений интенсивностей событий (λ_{12}=1751)

Снизить нагрузку на данное состояние сервера возможно путем увеличения его ресурса за счет создания очереди заявок между клиентом и сервером посредством изменения скорости потока данных установлением параметра «размер окна» W, регулируя значения интенсивностей λ_{34} и λ_{41}, а также за счет подтверждения сервером приема частей потока данных (квитирования) регулируя значения интенсивностей λ_{37} и λ_{56}.

На рис. 9 представлены графики зависимостей вероятностей состояний от времени для фиксированных значений интенсивностей событий и при λ_{12}=2619. При превышении данного порогового значения (λ_{12}=2619), соответствующего состоянию системы, в котором сервер получил максимальное количество заявок, которое он может обработать без переполнения буфера обмена или же снижения качества обслуживания заявок, наступает процесс исчерпания ресурса системы за счет незавершенных соединений.

№4. 2019
ISSN 2410-9916

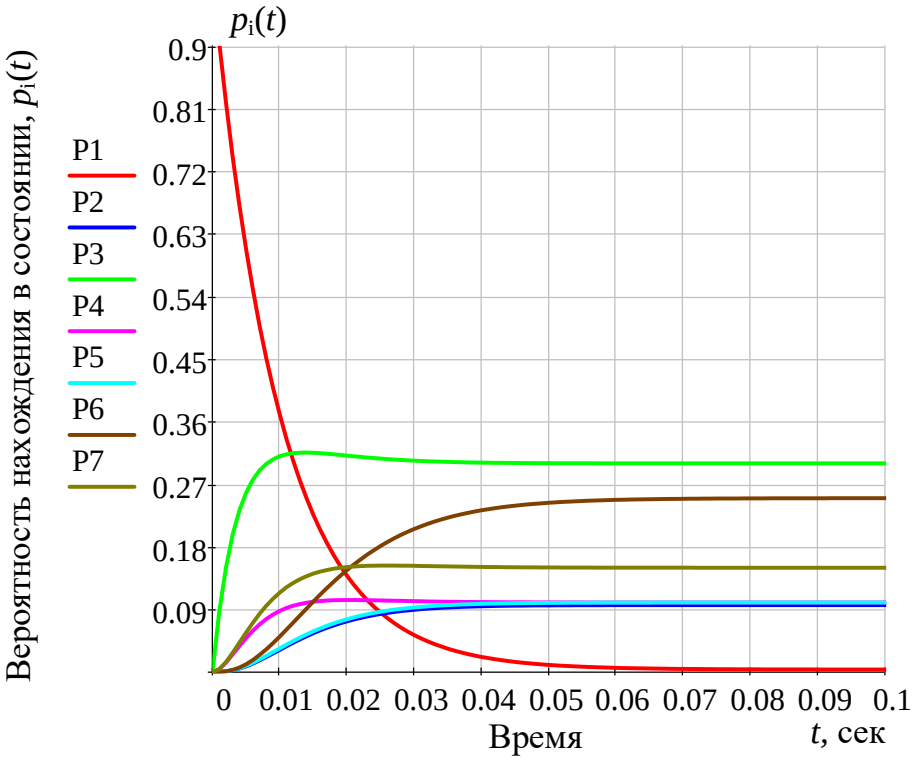

Рис. 9. Зависимости вероятностей состояний от времени для заданных значений интенсивностей событий (λ_{12}=2619)

Научная новизна модели заключается в применении математического аппарата теории Марковских случайных процессов и решении уравнений Колмогорова для исследования и решения задачи динамического управления ресурсными возможностями клиент-серверной ИС за счет управления параметрами сетевых соединений.

Практическая значимость заключается в нахождении вероятностных и временных характеристик, описывающих состояния процесса функционирования клиент-серверной ИС при различных стратегиях установления и поддержания параметров соединений взаимодействующими сторонами.

Алгоритм динамической конфигурации параметров сетевых соединений информационных систем в условиях сетевой разведки

Область применения алгоритма. Алгоритм относится к области информационной безопасности ИС и может быть использован в системах обнаружения и предупреждения атак с целью противодействия несанкционированным воздействиям в ИС, основанных на семействе коммуникационных протоколов TCP/IP.

Недостатками известных алгоритмов являются:

- относительно низкая результативность защиты ИС [44-46] от несанкционированных воздействий, признаками наличия которых являются несанкционированные информационные потоки (ИП);
- узкая область применения алгоритмов защиты [47-49];

DOI: 10.24411/2410-9916-2019-10403
URL: https://sccs.intelgr.com/archive/2019-04/03-Maximov.pdf

- относительно низкая результативность конфигурации параметров соединений [50] в условиях СР.

Низкая результативность защиты, конфигурации параметров соединений и узость области применения известных алгоритмов обусловлены:

- блокированием передачи пакетов сообщений при определении факта наличия несанкционированного ИП в ИС, что является недостаточным для их защиты, т.к. реализация указанного подхода к защите вынуждает СР далее воздействовать на ИС и (или) менять стратегию воздействия;
- возможностью перегрузки ИС с увеличением интенсивности несанкционированных ИП и сохранением заданного времени задержки отправки ответных пакетов сообщений отправителю (СР);
- снижением скорости ИП системой защиты в одностороннем порядке (т.е. только со стороны ИС) без учета возможности СР разорвать соединение;
- высокой вероятностью обнаружения СР факта использования средств защиты ИС и идентификации их характеристик, т. к. в прототипе используют значение служебного поля «размер окна» заголовка ответного TCP-пакета сообщений равное нулю, что позволяет СР идентифицировать средство защиты ИС путем изучения параметров сетевого обмена.

Назначение алгоритма – динамическая конфигурация параметров сетевых соединений ИС, обеспечивающая повышение результативности защиты ИС за счет снижения вероятности обнаружения СР факта использования средств защиты и идентификации их характеристик, достигаемой путем обеспечения реалистичности функционирования защищаемой ИС имитацией канала связи с плохим качеством и занятости ИС, удержания в двухстороннем порядке соединения с отправителем пакетов сообщений (СР), а также блокирования попыток СР разорвать соединение.

Физическая (содержательная) постановка задачи. Информационный обмен между абонентами (клиентами) и сервером ИС детализирован в модели функционирования клиент-серверной ИС. В процессе функционирования клиент-серверной ИС, СР инициирует запросы к серверу, который обрабатывает их в условиях ограниченного вычислительного ресурса. Ограниченность вычислительного ресурса выражается в том, что сервер способен обработать ограниченное количество запросов за единицу времени без переполнения буфера обмена или же снижения качества обслуживания заявок. Блокирование запросов СР приводит к компрометации системы защиты, в результате СР может менять стратегию воздействия. Бескомпроматное функционирование системы защиты, заключающееся в динамической конфигурации параметров сетевых соединений ИС со средствами СР, может привести к истощению ресурса средств СР.

Динамическая конфигурация параметров сетевых соединений в разработанном алгоритме основана на управлении потоком данных, реализуемом протоколом TCP, путем снижения скорости передачи данных между СР и получателем TCP-пакетов сообщений в течение определенного сеанса. Такая дискри-

минация трафика осуществляется путем ограничения количества сегментов данных, передаваемых за один раз, а также передачей фиктивных подтверждений до отправки следующих сегментов.

Ограничение количества сегментов данных, передаваемых за один раз, достигают управлением значения поля «размер окна», а передачей фиктивных подтверждений до отправки следующих сегментов достигают многократного дублирования передаваемых от СР сегментов.

На показатель скорости передачи полезной информации по каналу связи влияют и процессы фрагментации пакетов сообщений, которые заключаются в разбиении и упаковке исходного сформированного пакета сообщений в новые пакеты. При этом служебная информация пакета сообщений многократно дублируется, снижая значение этого показателя.

В процессе СР, средствами СР инициируют отправку пакетов сообщений по протоколу TCP, который в ответ на это посылает сегмент-запрос на установление соединения протоколу TCP (в запросе содержится флаг SYN, установленный в «1»). Получив от СР запрос, ИС выделяет определенные системные ресурсы, устанавливая начальное значение W_N поля «размер окна» (например, 25 байт) для формирования TCP-заголовка ответного пакета сообщений, объявляя СР (флагами ACK и SYN) о своей готовности получить небольшой, но достаточный для осуществления последующего информационного обмена объем данных, а также другие (такие как MSS, Maximum Segment Size и Padding) переменные соединения. После этого СР посылает сегмент с флагом ACK и переходит в состояние установленного логического соединения.

Если ИС (система защиты) в процессе описанного выше ответа, т. е. на первом этапе взаимодействия, установит поля «размер окна» в TCP-заголовке пакета сообщений равным нулю $W_U = 0$, инициализируя тем самым механизм удержания в двухстороннем порядке соединения с отправителем пакетов сообщений, то работа системы защиты будет скомпрометирована, а ресурсы СР не потрачены. Получив пакет сообщений с $W_U = 0$, в соответствии со спецификацией протокола TCP [40], СР сообщений будет периодически посылать однобайтовые сегменты, запрашивая у ИС информацию о размере окна и ожидаемом следующем байте (так называемый пробный сегмент «zero-window probe», чтобы определить, когда он сможет возобновить отправку данных. Алгоритмы-прототипы либо дублируют $W_U = 0$, либо устанавливают его фиксированное значение (например, равным 10), что приводит к возможности средствами СР осуществлять мониторинг ИС и компрометацию систем защиты в автоматическом режиме.

Для снижения вероятности идентификации характеристик средств защиты в разработанном алгоритме необходимо снизить информативность демаскирующего признака средства защиты, заключающегося в использовании значения служебного поля «размер окна» TCP-пакетов сообщений, по умолчанию устанавливаемого равным десяти байтам. Для этого применяют рандомизацию значения служебного поля «размер окна».

Попытки СР разорвать соединение с неудовлетворительным качеством (отправка пакета сообщений с флагом FIN в TCP-заголовке) или выслать срочн-

ные данные (отправка пакета сообщений с флагом URG в TCP-заголовке) игнорируют. ИС тем самым операционную систему СР поддерживать ресурсы соединения до истечения состояния FIN-WAIT-1 (продолжительность которого, установленная по умолчанию для операционной системы Linux 60 с представлена на рис. 10), ожидая TCP-сегмент от получателя пакетов сообщений с подтверждением о готовности закрыть соединение. Но он этот сегмент не получит, так как получатель пакетов сообщений его не высылает.

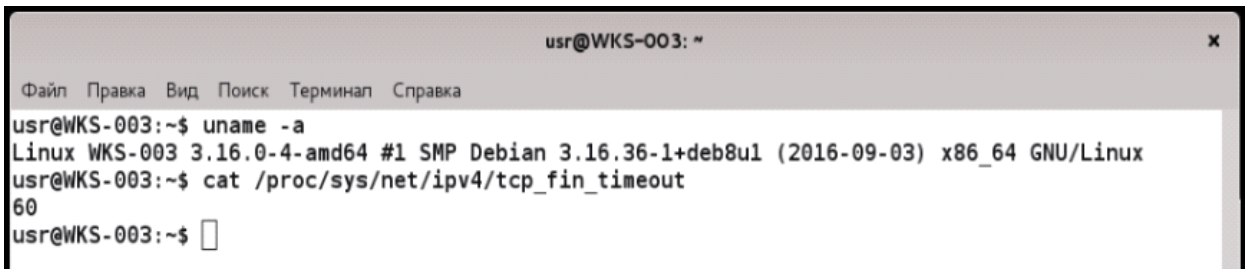

Рис. 10. Экранная копия командной строки со значением переменной, содержащей время тайм-аута соединения

В течение всего соединения ресурсы программного интерфейса для обмена данными между процессами СР расходуются для поддержания состояния соединения, что накладывает ограничение на используемый СР вычислительный ресурс. Это ограничение заключается в том, что СР не может использовать часть вычислительного ресурса, выделенного под удерживаемый в продолжительном соединении поток и в случае попытки инициализации им другого соединения к серверу, за счет параллельного использования информационных потоков, приведет в пределе к истощению ресурсов СР (отказу в обслуживании). В то же время ИС (сервер, получатель пакетов сообщений) не поддерживает состояние соединения со своей стороны и свой вычислительный ресурс не расходует, что позволяет ему в полной мере реализовать функции обработки поступающих пакетов сообщений от санкционированных клиентов.

Рассмотрим передачу фиктивных подтверждений (от ИС к СР) получения сегментов (от СР к ИС), а также их многократное дублирование. Для исчерпания ресурса средств СР необходимо имитировать нарушение порядка доставки сегментов и потери сегментов, что вызывает необходимость их повторной передачи от СР к ИС. В частности, возможно направление трех дубликатов подтверждения ACK на каждый из полученных фрагментов TCP-пакета сообщений, что представляет собой использование алгоритма контроля насыщения – алгоритма быстрого повтора (Fast Retransmit) для протокола TCP [41]. С точки зрения СР дубликат ACK может быть вызван различными сбоями в сети: причиной может служить отбрасывание сегментов (в этом случае все сегменты после отброшенного будут порождать дубликаты ACK), нарушение порядка доставки сегментов (например, при доставке по разным путям) или репликация пакетов ACK или сегментов данных в сети. После получения трех дубликатов ACK протокол TCP (на стороне СР) выполняет повторную передачу сегмента без ожидания за-

вершения отсчета таймера повтора передачи, предусмотренного спецификацией протокола TCP [40].

Перечисленные параметры конфигурации сетевых соединений позволяют повысить результативность защиты ИС от СР за счет удержания соединения со средством СР, что вызывает «истощение» ресурсов у средства СР для поддержания состояния соединения, замедляет процесс автоматического сканирования атакуемой ИС и, как результат, накладывает ограничение на используемый СР вычислительный ресурс, что приводит к невозможности осуществлять СР сетевой информационный обмен.

Таким образом, возникает ряд *противоречий*:

- между результативностью защиты ИС от СР и возможностями СР по определению структуры ИС, идентификации характеристик средств защиты, имеющих демаскирующие признаки, и их компрометации;
- между наличием необходимости по управлению конфигурацией ИС и отсутствием алгоритмов динамической конфигурации параметров сетевых соединений ИС в условиях СР.

На устранение указанных противоречий и направлен разработанный алгоритм.

Ограничения и допущения. Информация о санкционированности и несанкционированности клиента, устанавливающего соединение с ИС, считается достоверной за счет применения комплекса средств защиты ИС. Для получения численных оценок процесса защиты от СР используется разработанная приведенная выше модель функционирования клиент-серверной ИС. Конфигурация параметров сетевых соединений заключается в управлении значениями соответствующих параметров протокола TCP: Window Size и Acknowlegment.

Показатели и критерии. Показателем эффективности динамической конфигурации параметров сетевых соединений является максимизация вероятности простоя СР $P_D^{NI} \to \max$:

$$< S, C, Z, I> \to \max P_D^{NI} \mid P_D^{NI} \in \{P_i\},\ i = 1, 2,\ldots, h.$$

Теоретической основой алгоритма являются теории систем управления, вероятности, массового обслуживания, исследования операций.

Исходные данные. В качестве основных исходных данных в алгоритме выступают:

- множество внутренних параметров алгоритма $Z \subseteq \{S_i, \Lambda_j\}$, где $S_i = \{S_1, \ldots, S_k\}$, $\Lambda_j = \{\lambda_1, \lambda_2, \ldots, \lambda_J\}$ перечень моделируемых состояний системы и интенсивностей потоков событий в ней описаны ниже по тексту;
- объем массива памяти S_P (байт) для хранения k фрагментов, принятых от отправителя TCP-пакетов сообщений, где $k = 1, 2, \ldots, F$, а F – общее количество принятых фрагментов TCP-пакета сообщений;
- объем массива памяти G_P (байт) для хранения m сформированных ответных фрагментов TCP-пакета сообщений, где $m = 1, 2, \ldots, M$, а M – общее количество сформированных ответных фрагментов TCP-пакета сообщений;

- значение счетчика l количества сформированных ответных фрагментов TCP-пакета сообщений, хранящихся в массиве памяти G_P;
- объем массива памяти AG (байт) для хранения матрицы соответствия k-му принятому фрагменту TCP-пакета сообщений из массива S_P m-го сформированного фрагмента TCP-пакета сообщений из массива памяти G_P;
- длительность интервала времени t_{zad}, в течение которого отправителю TCP-пакетов сообщений будут направлены три дубликата подтверждения ACK.

Для достижения цели алгоритма осуществляют следующую *последовательность действий* (на рис. 11 представлена блок-схема последовательности действий, реализующих алгоритм динамической конфигурации параметров сетевых соединений ИС в условиях СР).

Задают исходные данные (см. блок 1 на рис. 11). Применение случайных значений служебного поля «размер окна» достигают тем, что задают (см. блок 1 на рис. 11) массив памяти S_P для хранения k фрагментов, принятых от отправителя TCP-пакетов сообщений, где $k = 1, 2, \ldots, F$, а F – общее количество принятых фрагментов TCP-пакета сообщений и массив памяти G_P для хранения m сформированных ответных фрагментов TCP-пакета сообщений, где $m = 1, 2, \ldots, M$, а M – общее количество сформированных ответных фрагментов *TCP*-пакета сообщений.

Также предварительно задают массив памяти AG для хранения матрицы соответствия k-му принятому фрагменту TCP-пакета сообщений из массива S_P m-го сформированного фрагмента TCP-пакета сообщений из массива памяти G_P и интервал времени t_{zad}, в течение которого отправителю TCP-пакетов сообщений будут направлены три дубликата подтверждения ACK. Для того, чтобы задать массив памяти, необходимо статически или динамически выделить объем оперативной памяти (в байтах). Использование t_{zad} применяется для увеличения общего времени удержания соединения с нарушителем.

Направление трех дубликатов подтверждения ACK отправителю TCP-пакетов сообщений в алгоритме применяется при однозначной идентификации факта ведения СР средствами защиты информации (в частности – при обращении средств СР к свободным IP-адресам, т. е. при сканировании ИС) и представляет собой использование в целях защиты стандартного алгоритма контроля насыщения – алгоритма быстрого повтора (Fast Retransmit) для протокола TCP для формирования у СР ложного представления о плохом качестве канала связи.

В дополнение к описанным выше причинам направления дубликатов ACK, получателю TCP-пакетов сообщений следует незамедлительно передавать подтверждение ACK при получении сегмента, который полностью или частично заполняет пропуски в порядковых номерах. Это позволит предоставить своевременную информацию отправителю TCP-пакетов сообщений, выполняющему восстановление после потери с использованием тайм-аута повторной передачи (retransmission timeout), быстрого повтора (fast retransmit) или улучшенного алгоритма восстановления (loss recovery).

Средству СР (отправителю TCP-пакетов сообщений) следует использовать алгоритм быстрого повтора для детектирования потери и исправления ошибки с использованием входящих дубликатов ACK. Алгоритм быстрого повтора использует прибытие трех дубликатов ACK, без каких-либо промежуточных сегментов ACK, как индикацию потери сегмента.

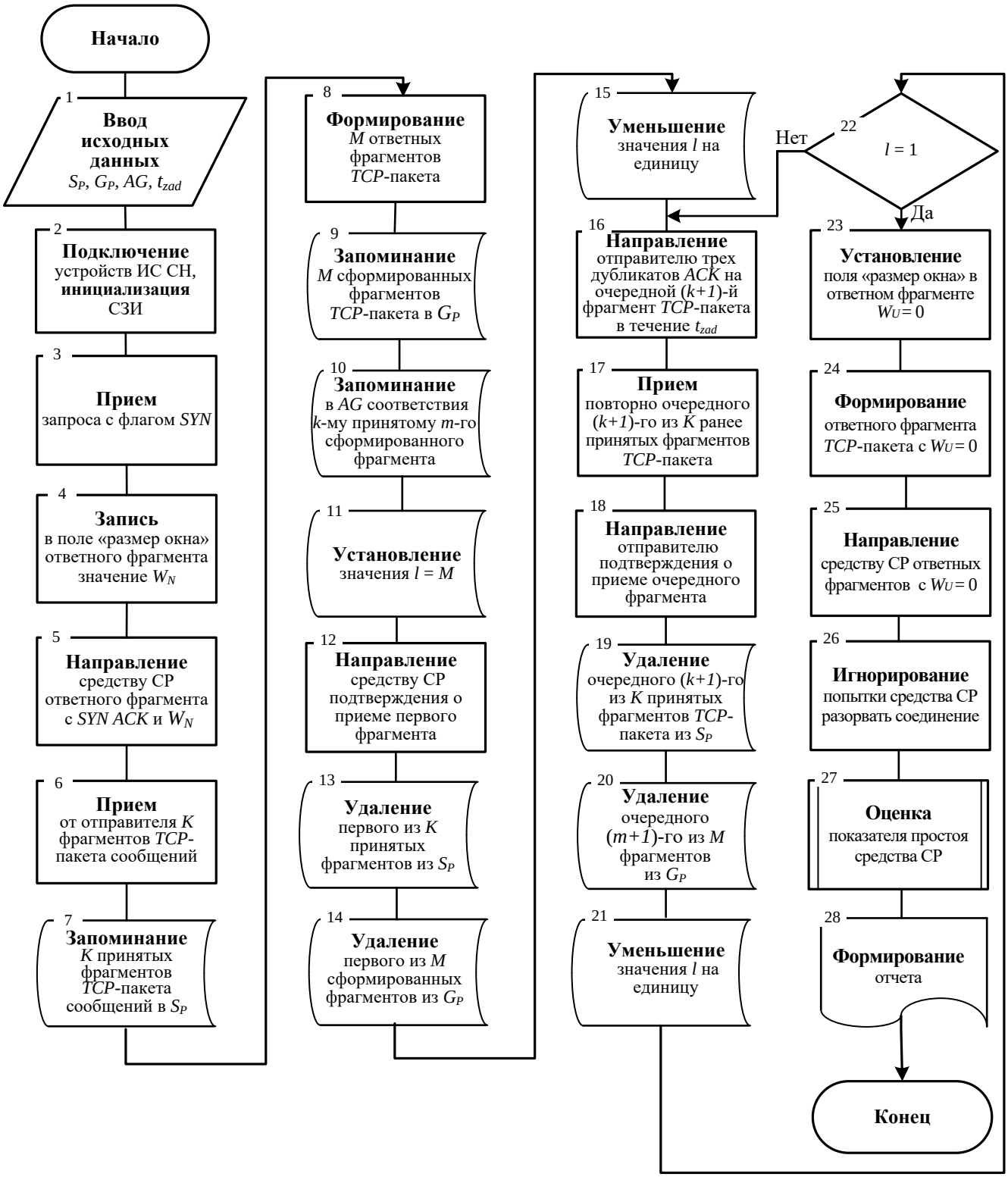

Рис. 11. Блок-схема последовательности действий, реализующая алгоритм динамической конфигурации параметров сетевых соединений ИС в условиях СР

После получения трех дубликатов ACK протокол TCP выполняет повторную передачу сегмента, который считается потерянным, без ожидания завершения отсчета таймера повтора передачи, предусмотренного спецификацией протокола TCP [40].

После задания исходных данных подключают сетевые устройства ИС (см. блок 2 на рис. 11) и инициализируют систему защиты.

При обнаружении средствами защиты информации процессов СР устанавливают TCP соединение со средством СР (т. е. отправителем пакетов сообщений), для чего принимают (см. блок 3 на рис. 11) фрагмент-запрос на установление соединения с установленным флагом SYN.

Предварительно в ИС (получатель пакетов сообщений) выделены системные ресурсы для информационного обмена. Для извещения средства СР о готовности принимать данные ограниченными блоками (фрагментами) ИС устанавливает начальное значение W_N поля «размер окна» при формировании TCP-заголовка ответного пакета сообщений, объявляя средству СР о своей готовности получить небольшой, но достаточный для осуществления последующего информационного обмена объем данных, а также другие переменные соединения. Переменные соединения [40], например, такие как максимальный размер сегмента (MSS, Maximum Segment Size) или заполнение заголовка TCP (Padding).

После этого записывают (см. блок 4 на рис. 2) в поле «размер окна» ответного фрагмента заданное случайным образом значение W_N и направляют (см. блок 5 на рис. 11) средству СР ответный фрагмент с установленными флагами SYN ACK и установленным значением поля «размер окна» W_N. Значение W_N служебного поля «размер окна» для формирования ответного фрагмента TCP-пакета сообщений выбирают в пределах от 11 до 30 байт. В ответ средство СР посылает сегмент с флагом ACK и переходит в состояние установленного логического соединения.

Принимают ИС (см. блок 6 на рис. 11) от средств СР TCP-пакет сообщений, состоящий из F фрагментов. Количество фрагментов F зависит от значения поля «размер окна» W_N, т.е. пакет данных, который необходимо передать отправителю TCP-пакетов сообщений будет приниматься получателем TCP-пакетов сообщений фрагментами не более, чем значение W_N.

Далее запоминают (см. блок 7 на рис. 11) F принятых от средств СР фрагментов TCP-пакета сообщений в массиве памяти S_P и формируют (см. блок 8 на рис. 11) M ответных фрагментов TCP-пакета сообщений. Затем запоминают (см. блок 9 на рис. 11) M сформированных ответных фрагментов TCP-пакета сообщений в массиве памяти G_P, а также запоминают (см. блок 10 на рис. 11) в массиве памяти AG соответствующий k-му принятому фрагменту TCP-пакета сообщений из массива S_P m-го сформированного ответного фрагмента TCP-пакета сообщений из массива памяти G_P. Запоминание в массиве памяти AG соответствия k-му принятому фрагменту TCP-пакета сообщений из массива S_P m-го сформированного ответного фрагмента TCP-пакета сообщений из массива памяти G_P осуществляют путем записи в ячейку $|k, m|$ массива памяти AG логической единицы. Двумерный массив памяти в результате содержит простую

матрицу, содержащую нули и единицы. Единица в ячейке матрицы означает соответствие *i*-го IP-адреса сетевого устройства *j*-му MAC-адресу.

После этого устанавливают (см. блок 11 на рис. 11) значение счетчика *l* количества сформированных ответных фрагментов TCP-пакета сообщений, хранящихся в массиве памяти G_P равным *M* и направляют (см. блок 12 на рис. 11) средству СР подтверждение о приеме первого фрагмента из *F* принятых фрагментов TCP-пакета сообщений.

Первый пакет принимают сразу в целях маскирования сетевой «ловушки», в случае применения для ее поиска в сети специальных программных средств, например, таких как [50-54] по нулевому или малому значению параметра «размер окна» заголовка TCP-пакета сообщений.

Далее удаляют (см. блок 13 на рис. 11) первый из *F* принятых TCP-сегментов от отправителя TCP-пакетов сообщений из массива памяти S_P и удаляют (см. блок 14 на рис. 11) первый из *M* сформированных ответных фрагментов TCP-пакета сообщений из массива памяти G_P. После этого уменьшают на единицу значение счетчика *l* количества сформированных *M* ответных фрагментов TCP-пакета сообщений (см. блок 15 на рис. 11).

В целях имитации потери отправленных фрагментов TCP-пакета сообщений, либо же нарушения порядка доставки фрагментов или репликации пакетов ACK или фрагментов данных в сети [41] направляют (см. блок 16 на рис. 11) средству СР три дубликата подтверждения ACK, говорящих о получении очередного из *K* полученных фрагментов TCP-пакета сообщений, например, с нарушением порядка доставки в течение t_{zad}. Значение t_{zad} интервала времени, в течение которого отправителю TCP-пакетов сообщений будут направлены три дубликата подтверждения ACK, выбирают в пределах от 5 до 9 с.

После этого принимают (см. блок 17 на рис. 11) повторно от отправителя сообщений очередной (*k+1*)-й из ранее принятых *F* фрагментов TCP-пакета сообщений и направляют (см. блок 18 на рис. 11) отправителю сообщений подтверждение об успешном получении очередного (*k+1*)-го из *F* ранее принятых фрагментов TCP-пакета сообщений.

Затем удаляют (см. блок 19 на рис. 11) очередной (*k+1*)-й из *F* принятых фрагментов TCP-пакета сообщений от отправителя *TCP*-пакетов сообщений из массива памяти S_P и удаляют (см. блок 20 на рис. 11) очередной (*m+1*)-й из *M* сформированных TCP-сегментов из массива памяти G_P.

После этого уменьшают (см. блок 21 на рис. 11) на единицу значение счетчика *l* количества *M* сформированных ответных фрагментов TCP-пакета сообщений, и так до тех пор, пока отправителем TCP-пакетов сообщений не будет повторно передан предпоследний (*F-1*)-й из принятых *F* фрагментов TCP-пакета сообщений, т.е. до тех пор, пока значение счетчика *l* количества *M* сформированных ответных фрагментов TCP-пакета сообщений (см. блок 22 на рис. 11) не станет равным единице $l = 1$. Это говорит о том, что в массиве памяти остался только один необработанный фрагмент и в случае его обработки, отправитель *TCP*-пакета сообщений может просто завершить соединение.

И для того, чтобы этого избежать, т.е. реализовать максимальное время контролируемого взаимодействия с нарушителем после повторного приема

предпоследнего (F-1)-го из F фрагментов TCP-пакета сообщений устанавливают (см. блок 23 на рис. 11) поле «размер окна» ответного фрагмента $W_U = 0$ и формируют (см. блок 24 на рис. 11) ответный фрагмент TCP-пакета сообщений с $W_U = 0$, направляют (см. блок 25 на рис. 11) отправителю TCP-пакетов сообщений ответный фрагмент с $W_U = 0$.

Получив пакет сообщений с $W_U = 0$, в соответствии со спецификацией протокола TCP [40], средство СР, как отправитель пакетов сообщений, будет периодически посылать пробные однобайтовые сегменты «zero window probe», запрашивая у ИС повтор информацию о значении поля «размер окна», чтобы определить, когда он сможет возобновить отправку данных. ИС, реализуя механизм удержания в двухстороннем порядке соединения с отправителем пакетов сообщений, не увеличивает окно, оставляя его равным нулю, тем самым удерживая отправителя пакетов сообщений заблокированным в продолжительном соединении на время, пока не истечет время тайм-аута (FIN-WAIT-1), определяемое предустановками операционной системы отправителя пакетов сообщений.

Если отправитель пакетов сообщений предпримет попытки разорвать удерживаемое соединение (отправкой пакета сообщений с флагом FIN = 1 в TCP-заголовке) или выслать срочные данные (отправка пакета сообщений с флагом URG = 1 в TCP-заголовке), то для блокирования попыток разорвать соединение со стороны средства СР игнорируют (см. блок 26 на рис. 11) все входящие фрагменты до тех пор, пока не истечет тайм-аут соединения.

После оценивания показателя простоя (см. блок 27 на рис. 11) средства СР формируют (см. блок 28 на рис. 11) отчет.

Для оценивания простоя средства СР в разработанном алгоритме используется модель функционирования клиент-серверной ИС с интерпретацией дискретных состояний S и интенсивностей потоков событий в ИС в условиях СР, приведенной в таблицах 11 и 12 соответственно.

Таблица 11 – Дискретные состояния ИС в условиях СР

Состояние	Описание состояния
S_1	Средство СР находится в состоянии простоя $P_D^{NI} \to \max$, и не передает пакеты сообщений к ИС (передача существенно затруднена)
S_2	Инициализация соединения средства СР (см. блок 3 на рис.11)
S_3	Оценка значения показателя простоя средства СР (см. блок 27 на рис.11)
S_4	Установление (изменение) скорости потока данных между средством СР и ИС установлением (изменением) параметра «размер окна» W (см. блоки 4, 23, 24 на рис.11)
S_5	Установление (подтверждение) соединения ИС и получение средством СР параметра «размер окна» W (см. блоки 5, 25 на рис.11)
S_6	Передача и прием потоков данных между средством СР и ИС (см. блок 6 на рис.11)
S_7	Подтверждение ИС приема частей потока данных от средства СР (см. блок 16 на рис.11)

Таблица 12 – Интенсивности потоков событий в ИС в условиях СР

Интенсивность	Описание интенсивности потоков событий
λ_{12}	Заявки на инициализацию (нового) соединения средством СР (см. блок 3 на рис.11)
λ_{23}	Заявки на оценку значения показателя простоя средства СР после инициализации им соединения (соединитель 3 на рис.11)
λ_{34}	Заявки на установление (изменение) скорости потока данных между средством СР и ИС установлением (изменением) параметра «размер окна» *W* (см. блоки 4, 23, 24 на рис.11)
λ_{43}	Заявки на оценку значения показателя простоя средства СР после установления (изменения) параметра «размер окна» *W* (соединитель 4, линия от блока 24 к блоку 27 на рис.11)
λ_{45}	Заявки на установление (подтверждение от ИС) соединения и получение средством СР параметра «размер окна» *W* (соединитель 5, линия от блока 25 к блоку 27 на рис.11)
λ_{56}	Заявки на передачу и прием потоков данных между средством СР и ИС после подтверждения от ИС установления соединения (передачи им W_N и *SYN ACK*) (см. блок 6 на рис.11)
λ_{63}	Заявки на оценку значения показателя простоя средства СР в процессе передачи и приема потоков данных между средством СР и ИС (соединитель 6 на рис.11)
λ_{37}	Заявки на подтверждение ИС приема частей потока данных (см. блок 16 на рис.11)
λ_{73}	Заявки на оценку значения показателя простоя средства СР после подтверждение ИС приема частей потока данных (соединитель 16 на рис.11)
λ_{76}	Заявки на передачу средством СР очередной части потока данных после подтверждения приема части потока (см. блок 6 на рис.11)
λ_{41}	Заявки на максимизацию значения показателя простоя средства СР $P_D^{NI} \to \max$ (параметров 0-скорости потока данных) установлением параметра «размер окна» $W = 0$ (см. блок 25 на рис.11)
λ_{13}	Заявки на контроль значения показателя простоя средства СР (линия от блока 26 к блоку 27 на рис.11)

Оценим устойчивость модели к вариациям исходных данных, задавая граничные значения в стратегиях противодействующих сторон, при этом рассмотрим следующие варианты стратегий:

- C_5 – без подтверждения и с очередью, в этом случае соединение средства СР с сервером осуществляется по протоколу TCP, сервер, получив значительное количество заявок на соединение от средства СР, выстраивает их в очередь, установлением параметра «размер окна» *W* равным нулю, подтверждение сервером приема частей потока данных (квитирования) отсутствует;
- C_6 – с подтверждением и с очередью, в этом случае соединение средства СР с сервером осуществляется по протоколу TCP, сервер, получив значительное количество заявок на соединение от средства СР, имитирует канал связи с плохим качеством за счет направления зло-

№4. 2019
ISSN 2410-9916

умышленнику дубликатов подтверждений АСК о потере или получении с нарушением порядка сегментов, затем выстраивает заявки от средства СР в очередь, установлением параметра «размер окна» W равным нулю.

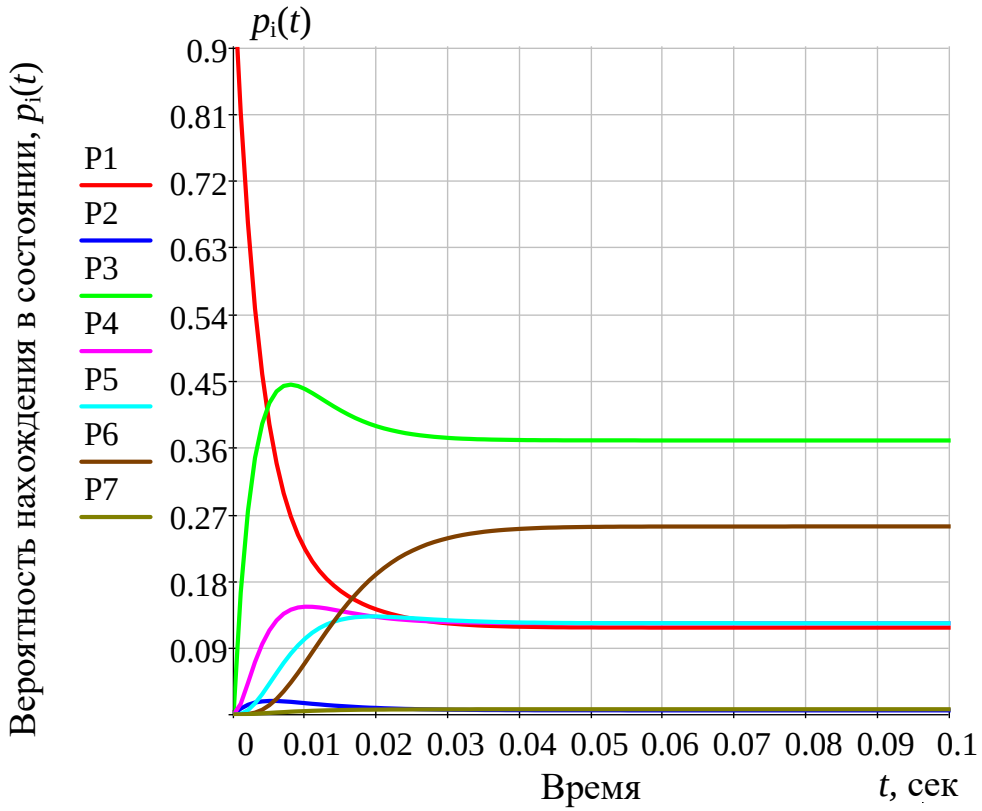

Рис. 12. Результаты расчета зависимости вероятностей состояний от времени для значений интенсивностей событий соответствующие стратегии C_5

Таблица 13 – Числовая таблица приближенных значений $p_i(t)$ для λ стратегии C_5

Этапы интегрирования, n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	0	1	0	0	0	0	0	0
2	$1\cdot10^{-2}$	0,819	$8{,}186\cdot10^{-4}$	0,165	0,015	$9{,}327\cdot10^{-4}$	$7{,}233\cdot10^{-5}$	$1{,}699\cdot10^{-4}$
3	$2\cdot10^{-2}$	0,675	$1{,}342\cdot10^{-3}$	0,274	0,042	$5{,}907\cdot10^{-3}$	$6{,}943\cdot10^{-4}$	$5{,}796\cdot10^{-4}$
…	…	…	…	…	…	…	…	…
10^3	10	0,185	$1{,}29\cdot10^{-3}$	0,415	0,141	0,125	0,126	$6{,}022\cdot10^{-3}$

DOI: 10.24411/2410-9916-2019-10403
URL: https://sccs.intelgr.com/archive/2019-04/03-Maximov.pdf

Таблица 14 – Интенсивности потоков событий для стратегий противодействия сервера ИС и средства СР

Описание интенсивности потоков событий	λ	Значения λ для стратегий C	
		C_5	C_6
Заявки на инициализацию (нового) соединения клиентом	λ_{12}	2	2
Заявки на оценку значения коэффициента простоя K_D после инициализации соединения клиентом	λ_{23}	200	200
Заявки на установление (изменение) скорости потока данных между клиентом и сервером установлением (изменением) параметра «размер окна» W	λ_{34}	200	200
Заявки на оценку значения коэффициента простоя K_D после установления (изменения) параметра «размер окна» W	λ_{43}	200	200
Заявки на установление (подтверждение) соединения сервером и получение клиентом параметра «размер окна» W	λ_{45}	200	200
Заявки на передачу и прием потоков данных между клиентом и сервером после подтверждения сервера установки соединения (передачи им W_N и SYN ACK)	λ_{56}	200	200
Заявки на оценку значения коэффициента простоя K_D в процессе передачи и приема потоков данных между клиентом и сервером	λ_{63}	100	100
Заявки на подтверждение сервером приема частей потока данных (квитирование)	λ_{37}	2	200
Заявки на оценку значения коэффициента простоя K_D после подтверждение сервером приема частей потока данных (квитирование)	λ_{73}	2	200
Заявки на передачу клиентом очередной части потока данных после подтверждения приема части потока	λ_{76}	100	100
Заявки на максимизацию значения коэффициента простоя клиента $K_D \rightarrow \max$ (параметров 0-скорости потока данных) установлением (изменением) параметра «размер окна» W	λ_{41}	200	200
Заявки на оценку значения коэффициента простоя K_D	λ_{13}	200	200

В процессе удержания соединения ресурсы отправителя пакетов сообщений расходуются для поддержания состояния соединения, что накладывает ограничение на используемый нарушителем вычислительный ресурс. В большинстве практических случаев «истощение» ресурсов у отправителя приводит к невозможности осуществлять отправителем пакетов сообщений сетевой информационный обмен.

В то же время получатель пакетов сообщений не поддерживает состояние соединения со своей стороны и свой вычислительный ресурс не расходует, что дает ему выигрыш во времени для адаптации системы защиты, заключающейся в перестройке ее параметров и (или) структуры, дает возможность продолжать обрабатывать пакеты сообщений, без переполнения буфера обмена и снижения качества обслуживания.

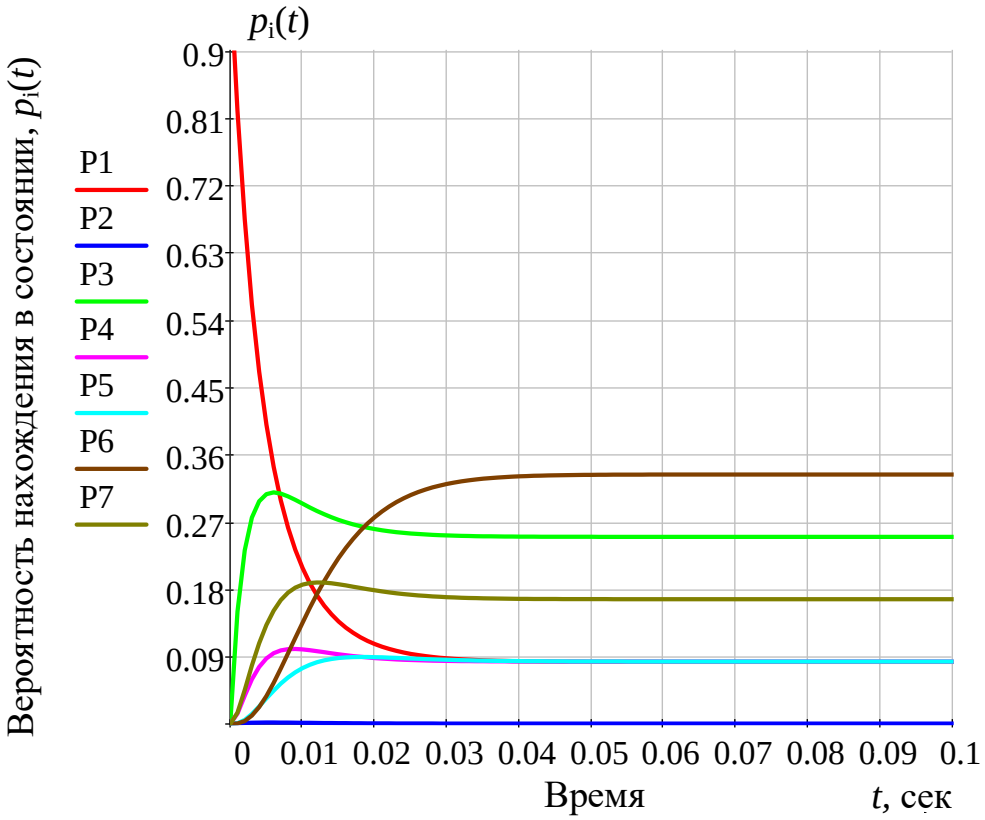

Рис. 13. Результаты расчета зависимости вероятностей состояний от времени для значений интенсивностей событий соответствующие стратегии C_6

Таблица 15 – Числовая таблица приближенных значений $p_i(t)$ для λ стратегии C_6

Этапы интегрирования, n	Точка интервала интегрирования, $[t_0, t_1]$	$p(t)$						
		$p_1(t)$	$p_2(t)$	$p_3(t)$	$p_4(t)$	$p_5(t)$	$p_6(t)$	$p_7(t)$
1	0	1	0	0	0	0	0	0
2	$1\cdot10^{-2}$	0,819	$8{,}186\cdot10^{-4}$	0,15	0,014	$8{,}667\cdot10^{-4}$	$5{,}667\cdot10^{-4}$	0,015
3	$2\cdot10^{-3}$	0,674	$1{,}342\cdot10^{-3}$	0,232	0,038	$5{,}374\cdot10^{-3}$	$3{,}886\cdot10^{-3}$	0,045
…	…	…	…	…	…	…	…	…
10^3	10	0,146	$1{,}165\cdot10^{-3}$	0,276	0,094	0,087	0,207	0,188

Свойства разработанного алгоритма. Свойства алгоритма целесообразно оценить по выполнению наиболее трудоемкой операции, которой в разработанном алгоритме является решение системы дифференциальных уравнений.

Детерминированность алгоритма. Алгоритм имеет постоянство структуры вычислительного процесса, выдает уникальный и предопределенный результат для заданных входных данных λ_{ij}.

Точность алгоритма. Примененный метод Рунге-Кутты имеет четвертый порядок точности, что означает, что ошибка на одном шаге имеет порядок $O(h^5)$, а суммарная ошибка на конечном интервале интегрирования имеет порядок $O(h^4)$.

Устойчивость алгоритма. Свойство алгоритма не увеличивать или увеличивать в незначительной степени погрешности, допущенной в начальных данных или допускаемой при вычислениях. Метод Рунге-Кутты 4-го порядка имеет интервал абсолютной устойчивости (– 2,78; 0) [55].

Производительность и масштабируемость. Отношение числа расчетов при последовательном алгоритме к числу расчетов при параллельном равняется числу используемых процессорных элементов, т. е. скорость расчетов при параллелизме увеличивается, но и число пересылок данных между процессорами увеличивается, что снижает производительность.

Научная новизна алгоритма заключается в применении модели функционирования клиент-серверной ИС, основанной на математическом аппарате теории Марковских случайных процессов и решении уравнений Колмогорова, для управления ресурсными возможностями СР при установлении и поддержании сетевых соединений.

Практическая значимость заключается в решении задачи динамической конфигурации параметров сетевых соединений ИС, обеспечивающей дискриминацию трафика СР, скрытие факта использования средств защиты и идентификации их характеристик.

Выводы

Разработанная модель позволяет определять вероятностные и временные характеристики, описывающие состояния процесса функционирования клиент-серверной ИС при различных стратегиях установления и поддержания параметров соединений взаимодействующими сторонами, что позволяет оценивать состояние клиент-серверной ИС.

Разработанный алгоритм позволяет повысить результативность защиты, по сравнению с аналогами, за счет снижения вероятности обнаружения нарушителем факта использования средств защиты и идентификации их характеристик с применением известных средств СР, достигаемой путем имитации канала связи с плохим качеством, удержания в двухстороннем порядке соединения с отправителем пакетов сообщений, а также блокирования попыток отправителя разорвать соединение.

Наиболее близким аналогом по своей технической сущности к представленному научно-методическому аппарату является [10, 14, 50], где обеспечивается повышение защищенности ИС от несанкционированных воздействий за счет удержания в двухстороннем порядке соединения с отправителем пакетов сообщений, обеспечивая тем самым увеличение дискомфорта у нарушителя и выигрыш по времени, необходимый для реализации ответных мер.

Однако недостатками указанных аналогов является относительно низкая результативность защиты, обусловленная высокой вероятностью обнаружения нарушителем факта использования средств защиты информационной системы и идентификации их характеристик. Это связано с тем, что в аналогах используются первичное значение W_N служебного поля «размер окна» заголовка ответного *TCP*-пакета сообщений равное нулю, что позволяет злоумышленнику идентифицировать средство защиты информационной системы путем изучения

параметров сетевого обмена с применением специального программного обеспечения [56].

Указанные недостатки могут быть решены применением указанного научно-методического аппарата.

Литература

1. Шерстобитов Р. С., Шарифуллин С. Р., Максимов Р. В. Маскирование интегрированных сетей связи ведомственного назначения // Системы управления, связи и безопасности. 2018. № 4. С. 136-175. URL: http://sccs.intelgr.com/archive/2018-04/08-Sherstobitov.pdf (дата обращения 14.10.2019).

2. Искольный Б. Б., Максимов Р. В., Шарифуллин С. Р. Оценка живучести распределенных информационно-телекоммуникационных сетей // Вопросы кибербезопасности. 2017. № 5 (24). С. 72-82. doi: 10.21681/2311-3456-2017-5-72-82.

3. Максимов Р. В., Андриенко А. А., Куликов О. Е., Костырев А. Л., Павловский А. В., Лебедев А. Ю. Способ контроля информационных потоков в цифровых сетях связи // Патент на изобретение RU 2267154, опубл. 27.12.2005, бюл. № 36, 16 с.

4. Ворончихин И. С., Иванов И. И., Максимов Р. В., Соколовский С. П. Маскирование структуры распределенных информационных систем в киберпространстве // Вопросы кибербезопасности. 2019. № 6 (34). С. 92–101. DOI: 10.21681/2311-3456-2019-6-92-101.

5. Голуб Б. В., Кузнецов Е. М., Максимов Р. В. Методика оценки живучести распределенных информационных систем // Вестник Самарского государственного университета. 2014. № 7 (118). С. 221-232.

6. Максимов Р. В., Соколовский С. П., Шарифуллин С. Р., Чернолес В. П. Инновационные информационные технологии в контексте обеспечения национальной безопасности государства // Инновации. 2018. № 3 (233). С. 28-35.

7. Максимов Р. В., Кожевников Д. А., Колбасова Г. С., Самохин В. Ф., Чернолес В. П. Патентная безопасность как составляющая информационной безопасности в сфере науки и техники России // Инновации. 2006. №11. С. 41-47.

8. Давыдов А. Е., Максимов Р. В., Савицкий О. К. Защита и безопасность ведомственных интегрированных инфокоммуникационных систем. – М.: ОАО «Воентелеком», 2015. – 520 с.

9. Максимов Р. В., Савинов Е. А. Оценка живучести распределенных интегрированных информационных систем // Информационные технологии и нанотехнологии (ИТНТ-2016): Материалы Международной конференции и молодежной школы. – Самара: Самарский государственный аэрокосмический университет имени академика С.П. Королева (национальный исследовательский университет), Институт систем обработки изображений РАН, 2016. С. 431-438.

10. Provos N., Holz T. Virtual Honeypots: From Botnet Tracking to Intrusion Detection. – Addison Wesley, 2007. – 480 p.

11. Максимов Р. В., Орехов Д. Н., Проскуряков И. С., Соколовский С. П. Способ защиты вычислительных сетей // Патент на изобретение RU 2649789, опубл. 04.04.2018, бюл. № 10, 25 с.

12. Максимов Р. В., Орехов Д. Н., Соколовский С. П., Крупенин А. В., Гаврилов А. Л., Катунцев С. Л., Медведев А. Н. Способ защиты вычислительных сетей // Патент на изобретение RU 2682432, опубл. 19.03.2019, бюл. № 8, 23 с.

13. Максимов Р. В., Орехов Д. Н., Соколовский С. П., Гаврилов А. Л., Катунцев С. Л., Проскуряков И. С., Прокопенко А. В. Способ защиты вычислительных сетей // Патент на изобретение RU 2682432, опубл. 14.02.2019, бюл. № 5, 22 с.

14. Andres S., Kenyon B., Birkolz E. Security Sage's Guide to Hardening the Network Infrastructure. – Sungress Publ., 2004. – 608 p.

15. Максимов Р. В., Орехов Д. Н., Соколовский С. П., Гаврилов А. Л., Катунцев С. Л., Маленков Е. С., Платов Н. Е., Шаманов А. И. Способ защиты вычислительных сетей // Патент на изобретение RU 2690749, опубл. 05.06.2019, бюл. № 16, 25 с.

16. Максимов Р. В., Орехов Д. Н., Соколовский С. П., Барабанов В. В., Ефремов А. А., Ворончихин И. С. Способ защиты вычислительных сетей // Патент на изобретение RU 2696330, опубл. 01.08.2019, бюл. № 22, 30 с.

17. Выговский Л. С., Максимов Р. В. Модель преднамеренных деструктивных воздействий на информационную инфраструктуру интегрированных систем связи // Научно-технические ведомости СПбГПУ. 2009. № 1 (73). С. 181-187.

18. Максимов Р. В., Орехов Д. Н., Соколовский С. П., Гаврилов А. Л., Катунцев С. Л., Пряхин В. П., Тимашенко Д. В., Тимашенко В. К. Способ защиты вычислительных сетей // Патент на изобретение RU 2686023, опубл. 23.04.2019, бюл. № 12, 26 с.

19. Liston T. LaBrea: «sticky» Honeypot and IDS. [Электронный ресурс]. URL: http://labrea.sourceforge.net/labrea-info.html (дата обращения: 03.09.2019)

20. Alt L., Beverly R., Dainotti A. Uncovering network tarpits with degreaser // In Proceedings of the 30th Annual Computer Security Applications Conference (ACSAC '14). ACM, New York, NY, USA, 2014. Pp. 156-165. DOI: 10.1145/2664243.2664285.

21. Макаренко С. И., Михайлов Р. Л. Оценка устойчивости сети связи в условиях воздействия на неё дестабилизирующих факторов // Радиотехнические и телекоммуникационные системы. 2013. № 1. С. 69-79.

22. Гречишников Е. В., Горелик С. П., Белов А. С. Способ управления защищенностью сетей связи в условиях деструктивных программных воздействий // Телекоммуникации. 2014. № 3. С. 18-22.

23. Язов Ю. К., Сердечный А. Л., Шаров И. А. Методический подход к оцениванию эффективности ложных информационных систем // Вопросы кибербезопасности. 2014. № 1 (2). С. 55-60.

24. Пахомова А. С., Пахомов А. П., Разинкин К. А. К вопросу о разработке структурной модели угрозы компьютерной разведки // Информация и безопасность. 2013. Том 16. № 1. С. 115-118.

25. Макаренко С. И. Динамическая модель системы связи в условиях функционально-разноуровневого информационного конфликта наблюдения и подавления // Системы управления, связи и безопасности. 2015. № 3. С. 122-185. – URL: http://sccs.intelgr.com/archive/2015-03/07-Makarenko.pdf (дата обращения 17.02.2019).

26. Макаренко С. И. Информационное оружие в технической сфере: терминология, классификация, примеры // Системы связи, управления и безопасности. 2016. № 3. С. 292–376. – URL: http://sccs.intelgr.com/archive/2016-03/11-Makarenko.pdf (дата обращения: 18.04.2019).

27. Бухарин В. В., Карайчев С. Ю., Пикалов Е. Д. Способ защиты от деструктивных программных воздействий в мультисервисных сетях связи // Вопросы кибербезопасности. 2016. № 3 (16). С. 18-24.

28. Максимов Р. В., Павловский А. В., Стародубцев Ю. И. Защита информации от технических средств разведки в системах связи и автоматизации. – СПб.: ВАС, 2007. – 88 с.

29. Вандич А. П., Яичкин М. А., Карганов В. В., Привалов А. А., Скуднева Е. В. К вопросу об организации информационного обмена для повышения защищенности сети передачи данных от технической компьютерной разведки Труды ЦНИИС. Санкт-Петербургский филиал. 2017. Т. 1. № 4. С. 72-78.

30. Привалов А. А., Скуднева Е. В., Вандич А. П., Яичкин М. А. Метод повышения структурной скрытности сетей передачи данных оперативно технологического назначения ОАО «РЖД» // ТРУДЫ ЦНИИС. Санкт-Петербургский филиал. Том 2(3). – 2016. – С. 65-74.

31. Бухарин В. В., Кирьянов А. В., Стародубцев Ю. И. Способ защиты вычислительных сетей // Информационные системы и технологии. 2012. № 4 (72). С. 116-121.

32. Сердечный А. Л., Шаров И. А., Сигитов В. Н. Подход к моделированию процесса компьютерной разведки в информационных системах с изменяющимися составом и структурой // REDS: Телекоммуникационные устройства и системы. 2015. Т. 5. № 4. С. 439-443.

33. Евглевская Н. В., Привалов А. А., Скуднева Е. В. Марковская модель конфликта автоматизированных систем обработки информации и управления с системой деструктивных воздействий нарушителя // Известия Петербургского университета путей сообщения. 2015. № 1 (42). С. 78-84.

34. Стародубцев Ю. И., Бегаев А. Н., Козачок А. В. Способ управления доступом к информационным ресурсам мультисервисных сетей различных уровней конфиденциальности // Вопросы кибербезопасности. 2016. № 3 (16). С. 13-17.

35. Максимов Р. В., Соколовский С. П., Орехов Д. Н. Особенности детектирования и способы маскирования демаскирующих признаков средств проактивной защиты вычислительных сетей // Радиолокация, навигация, связь:

Сборник трудов XXIV Международной научно-технической конференции. Том 2. – Воронеж: ООО «Вэлборн», 2018. С. 169-179.

36. Выговский Л. С., Максимов Р. В. Модель преднамеренных деструктивных воздействий на информационную инфраструктуру интегрированных систем связи // Научно-технические ведомости СПбГПУ. 2008. № 3 (60). С. 166-173.

37. Maximov R. V., Krupenin A. V., Sharifullin S. R., Sokolovsky S. P. Innovative development of tools and technologies to ensure the Russian information security and core protective guidelines // Вопросы кибербезопасности. 2019. № 1 (29). С. 10-17. DOI: 10.21681/2311-3456-2019-1-10-17.

38. Максимов Р. В., Берест П. А., Богачев К. Г., Выговский Л. С., Игнатенко А. В., Кожевников Д. А., Краснов В. А., Кузнецов В. Е. Способ сравнительной оценки структур информационно-вычислительной сети // Патент на изобретение RU 2408928, опубл. 10.01.2011, Бюл. № 1, 16 с.

39. Максимов Р. В., Кожевников Д. А., Павловский А. В., Юрьев Д. Ю. Способ выбора безопасного маршрута в сети связи (варианты) // Патент на изобретение RU 2331158, опубл. 10.08.2008, Бюл. № 22, 34 с.

40. RFC 793. Transmission Control Protocol. DARPA internet program protocol specification. 1981. – URL: https://tools.ietf.org/html/rfc793 (дата обращения 04.09.2019).

41. RFC 5681. TCP Congestion Control. 2009. – URL: https://tools.ietf.org/html/rfc5681 (дата обращения 04.09.2019).

42. Макаренко С. И. Справочник научных терминов и обозначений. – СПб.: Наукоемкие технологии, 2019. – 254 с.

43. Вержбицкий В. М. Основы численных методов: учебник для вузов. – М.: Высшая. Школа, 2002. – 840 с.

44. Максимов Р. В., Андриенко А. А., Кожевников Д. А., Колбасова Г. С., Павловский А. В., Стародубцев Ю. И. Способ (варианты) и устройство (варианты) защиты канала связи вычислительной сети // Патент на изобретение RU 2306599, опубл. 20.09.2007, Бюл. № 26, 56 с.

45. Максимов Р. В., Кожевников Д. А., Павловский А. В. Способ защиты вычислительной сети (варианты) // Патент на изобретение RU 2325694, опубл. 27.05.2008, Бюл. № 15, 106 с.

46. Максимов Р. В., Ветошкин И. С., Дрозд Ю. А., Ефимов А. А., Игнатенко А. В., Кожевников Д. А., Краснов В. А., Кузнецов В. Е. Способ защиты вычислительной сети с выделенным сервером // Патент на изобретение RU 2449361, опубл. 10.02.2011, Бюл. № 4, 16 с.

47. Максимов Р. В., Выговский Л. С., Заргаров И. А., Кожевников Д. А., Павловский А. В., Стародубцев Ю. И., Худайназаров Ю. К., Юров И. А. Способ (варианты) защиты вычислительных сетей // Патент на изобретение RU 2307392, опубл. 27.09.2007, Бюл. № 27, 22 с.

48. Максимов Р. В., Андриенко А. А., Куликов О. Е., Костырев А. Л. Способ обнаружения удаленных атак на автоматизированные системы управления // Патент на изобретение RU 2264649, опубл. 20.11.2005, Бюл. № 32, 15 с.

49. Максимов Р. В., Голуб Б. В., Горячая А. В., Кожевников Д. А., Лыков Н. Ю., Тихонов С. С. Способ маскирования структуры сети связи // Патент на изобретение RU 2622842, опубл. 20.06.2017, юл. № 17, 21 с.

50. Grimes R. A. Honeypots for Windows. – Apress, 2005. – 424 p.

51. Maximov R. V., Sokolovsky S. P., Gavrilov A. L. Hiding computer network proactive security tools unmasking features // Selected Papers of the VIII All-Russian Conference with International Participation «Secure Information Technologies» (BIT 2017). (Moscow, 6-7 December 2017) – Moscow, Bauman Moscow Technical University Publ., 2017. P. 88-92.

52. Максимов Р. В., Искольный Б. Б., Лазарев А. А., Лыков Н. Ю., Хорев Г. А., Шарифуллин С. Р. Способ сравнительной оценки структур сетей связи // Патент на изобретение RU 2626099, опубл. 21.07.2017, бюл. № 21, 19 с.

53. Iskolnyy B. B., Maximov R. V., Sharifullin S. R. Survivability Assessment of Distributed Information and Telecommunication Networks // Selected Papers of the VIII All-Russian Conference with International Participation «Secure Information Technologies» (BIT 2017). (Moscow, 6-7 December 2017) – Moscow, Bauman Moscow Technical University, 2017. P. 59-65.

54. Maximov R. V., Ivanov I. I., Sharifullin S. R. Network Topology Masking in Distributed Information Systems // Selected Papers of the VIII All-Russian Conference with International Participation «Secure Information Technologies» (BIT 2017). (Moscow, 6-7 December 2017) – Moscow, Bauman Moscow Technical University , 2017. P. 83-87.

55. Заусаев А. Ф. Разностные методы решения обыкновенных дифференциальных уравнений: учебное пособие. – Самара: Самарский государственный технический университет, 2010. – 100 с.

56. Sokolovsky S. P., Telenga A. P., Voronchikhin I. S. Moving target defense for securing Distributed Information Systems // Информатика: проблемы, методология, технологии: сборник материалов XIX международной научно-методической конференции. – Воронеж: Издательство «Научно-исследовательские публикации» (ООО «Вэлборн»), 2019. С. 639-643.

References

1. Sherstobitov R. S., Sharifullin S. R., Maximov R. V. Masking of departmental-purpose integrated communication networks. *Systems of Control, Communication and Security*, 2018, no. 4, pp. 136-175. Available at: http://sccs.intelgr.com/archive/2018-04/08-Sherstobitov.pdf (accessed 14 October 2019) (in Russian).

2. Iskolnyy B. B., Maximov R. V., Sharifullin S. R. Evaluation of the Survivability of Integrated Information-Telecommunication Networks. *Voprosy kiberbezopasnosti*, 2017, no. 5 (24), pp. 72-82. doi: 10.21681/2311-3456-2017-5-72-82 (in Russian).

3. Maximov R. V., Andrienko A. A., Kulikov O. E., Kostyrev A. L., Pavlovskij A. V., Lebedev A. YU. *Sposob kontrolya informacionnyh potokov v cifrovyh setyah svyazi* [Method of Control of Information Flows in Digital Communication Networks]. Patent Russia, no. 2267154, 27.12.2005.

4. Voronchikhin I. S., Ivanov I. I., Maximov R. V., Sokolovsky S. P. Masking of Distributed Information Systems Structure In Cyber Space. *Voprosy kiberbezopasnosti*, 2019, no. 6 (34), pp. 92–101. doi: 10.21681/2311-3456-2019-6-92-101. (in Russian).

5. Golub B. V., Kuznetsov E. M., Maximov R.V. Method For Estimation Of Vitality Of Allocated Information Systems. *Vestnik of Samara state University*, 2014, no. 7(118), pp.221-232 (in Russian).

6. Maximov R. V., Sokolovsky S. P., Sharifullin S. R., Chernoles V. P. Innovative Information Technologies in the Context of National Security. *Innovations*, 2018, no. 3 (233), pp. 28-35 (in Russian).

7. Maximov R. V., Kozhevnikov D. A., Kolbasova G. S., Samokhin V. F., Chernoles V. P. Patentnaia bezopasnost' kak sostavliaiushchaia informatsionnoi bezopasnosti v sfere nauki i tekhniki Rossii [Patent security as a component of information security in the field of science and technology in Russia]. *Innovations*, 2006, no. 11, pp. 41-47.

8. Davydov A. E., Maximov R. V., Savickij O. K. *Zashchita i bezopasnost' vedomstvennyh integrirovannyh infokommunikacionnyh sistem* [The Protection and Security of Departmental Integrated Information and Communication Systems]. Moscow, «Voentelekom» Publ., 2015. 520 p. (in Russian).

9. Maximov R. V., Savinov E. A. Otsenka zhivuchesti raspredelennykh integrirovannykh informatsionnykh system [Information Distributed Integrated Systems Survivability Assessment]. *Information Technology and Nanotechnology (ITNT-2016). Proceedings of the International conference and young scientists school*. Samara State Aerospace University, Russian Academy of Sciences Image Processing Systems Institute Ipsi Ras 2016, pp. 431-438 (in Russian).

10. Provos N., Holz T. *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*. Addison Wesley, 2007. 480 p.

11. Maximov R. V., Orekhov D. N., Proskuryakov I. S., Sokolovsky S. P. *Sposob zashchity vychislitel'nyh setej* [Method of Protection of Computer Networks]. Patent Russia, no. 2649789, 04.04.2018.

12. Maximov R. V., Orekhov D. N., Sokolovsky S. P., Krupenin A. V., Gavrilov A. L., Katuncev S. L., Medvedev A. N. *Sposob zashchity vychislitel'nyh setej* [Method of Protection of Computer Networks]. Patent Russia, no. 2682432, 19.03.2019.

13. Maximov R. V., Orekhov D. N., Sokolovsky S. P., Gavrilov A. L., Katuncev S. L., Proskuryakov I. S., Prokopenko A. V. *Sposob zashchity vychislitel'nyh setej* [Method of Protection of Computer Networks]. Patent Russia, no. 2682432, 14.02.2019.

14. Andres S., Kenyon B., Birkolz E. *Security Sage's Guide to Hardening the Network Infrastructure*. Sungress Publ., 2004. 608 p.

15. Maximov R. V., Orekhov D. N., Sokolovsky S. P., Gavrilov A. L., Katuncev S. L., Malenkov E. S., Platov N. E., Shamanov A. I. *Sposob zashchity vychislitel'nyh setej* [Method of Protection of Computer Networks]. Patent Russia, no. 2690749, 05.06.2019.

16. Maximov R. V., Orekhov D. N., Sokolovsky S. P., Barabanov V. V., Efremov A. A., Voronchikhin I. S. *Sposob zashchity vychislitel'nyh setej* [Method of Protection of Computer Networks]. Patent Russia, no. 2696330, 01.08.2019.

17. Vygovskij L. S., Maximov R. V. Model' prednamerennyh destruktivnyh vozdejstvij na informacionnuyu infrastrukturu integrirovannyh sistem svyazi [Model of Intentional Destructive Impacts on the Information Infrastructure of Integrated Communication Systems]. *St. Petersburg state Polytechnic University State University Scientific bulletin*, 2009, vol. 1, no. 73, pp. 181-187 (in Russian).

18. Maximov R. V., Orekhov D. N., Sokolovsky S. P., Gavrilov A. L., Katuncev S. L., Pryakhin V. P., Timashenko D. V., Timashenko D. K. *Sposob zashchity vychislitel'nyh setej* [Method of Protection of Computer Networks]. Patent Russia, no. 2686023, 23.04.2019.

19. Liston T. LaBrea: «sticky» Honeypot and IDS. Available at: http://labrea.sourceforge.net/labrea-info.html (accessed 01 June 2019).

20. Alt L., Beverly R., Dainotti A. Uncovering network tarpits with degreaser. *In Proceedings of the 30th Annual Computer Security Applications Conference (ACSAC '14)*. ACM, New York, NY, USA, 2014, pp. 156-165. DOI: 10.1145/2664243.2664285.

21. Makarenko S. I., Mikhailov R. L. Estimating Communication Network Stability under the Conditions of Destabilizing Factors Affecting it. *Radio and telecommunication systems*, 2013, no. 4, pp. 69–79 (in Russian).

22. Grecihnikov E. V., Gorelik S. P., Belov A. S. Sposob upravleniya zashchishchennost'yu setey svyazi v usloviyakh destruktivnykh programmnykh vozdeystviy [A method of controlling the security of communication networks in terms of destructive program influences]. *Telecommunications*, 2014, no. 3, pp. 18-22 (in Russian).

23. Yazov Yu. K., Serdechnyy A. L., Sharov I. A. Methodical approach for estimation of efficiency of honeypot system. *Voprosy kiberbezopasnosti*, 2014, vol. 2, no. 1, pp. 55-60 (in Russian).

24. Pakhomova A. S., Pakhomov A. P., Razinkin K. A. K voprosu o razrabotke strukturnoi modeli ugrozy komp'iuternoi razvedki [To the problem of the development of a structural model of computer intelligence]. *Informatsiia i bezopasnost'*, 2013, vol. 16, no. 1, pp. 115-118 (in Russian).

25. Makarenko S. I. Dynamic Model of Communication System in Conditions the Functional Multilevel Information Conflict of Monitoring and Suppression. *Systems of Control, Communication and Security*, 2015, no. 3, pp. 122-185. Available at: http://sccs.intelgr.com/archive/2015-03/07-Makarenko.pdf (accessed 17 February 2019) (in Russian).

26. Makarenko S. I. Information Weapons in the Technical Sphere: Terminology, Classification, Examples. *Systems of Control, Communication and Security*, 2016. no. 3, pp. 292–376. Available at: http://sccs.intelgr.com/archive/2016-03/11-Makarenko.pdf (accessed 18 April 2019) (in Russian).

27. Buharin V. V., Karaichev S. Yu., Pikalov E. D. Protection Method From Destructive Software Effects In Multiservice Networks. *Voprosy kiberbezopasnosti*, 2016, vol. 3, no. 16, pp. 18-24 (in Russian).

28. Maksimov R. V., Pavlovskiy A. V., Starodubcev Yu. I. *Zaschita informacii ot tchnischteskich sredstv razvedki v sistemach svyazi i avtomatizacii* [Information Security from Technical Means of Investigation in Communication Systems and Automation]. Saint-Petersburg, Military Academy of the Signal Corps, 2007. 88 p. (in Russian).

29. Vandich A. P., Yayichkin M. A., Karganov V. V., Privalov A. A, Skudneva E. V. On Question of Informational Exchange for Sequrity Improvement Against the Technical Computer Intelligence Data Networks, Saint-Petersburg, *Trudy Tsentral'nogo Nauchno-Issledovatel'skogo Instituta Sviazi*, 2017, vol. 1, no 4, pp. 72-78. (in Russian).

30. Privalov A. A., Skudneva E. V., Vandich A. P., Yaichkin M. A. Method of Structural Secrecy Increasing of Russian Railways Data Networks for Operational Processes. Saint-Petersburg, *Trudy Tsentral'nogo Nauchno-Issledovatel'skogo Instituta Sviazi*, 2016, vol. 2, no 3, pp. 65-74 (in Russian).

31. Buharin V. V., Kiryanov A. V., Starodubcev Yu. I. Method for Protecting Computer Networks. *Information Systems and Technologies*, 2012, vol. 4, no. 72, pp. 116-121 (in Russian).

32. Serdechnyy A. L., Sharov I. A., Sigitov V. N. Podhod k modelirovaniyu processa komp'yuternoj razvedki v informacionnyh sistemah s izmenyayushchimisya sostavom i strukturoj [Approach to Modeling Process of Cyberespionage in Information System with Changing Composition and Structure]. *REDS: Radio-Electronic Devices And Systems*, 2015, vol. 5, no. 4, pp. 439-443 (in Russian).

33. Evgrlevskaya N. V., Privalov A. A., Skudneva E. V. Markov Model of Conflict of Automated Information Processing and Management Systems with the System of Destructive Effects of an Offender. *Proceedings of Petersburg Transport University*, 2015, vol. 1, no. 42, pp. 78-84 (in Russian).

34. Starodubtsev Y. I., Begaev A. N., Kozachok A. V. Proposals for Access Control to Information Resources of Computer Networks Different Levels of Privacy. *Voprosy kiberbezopasnosti*, 2016, vol. 3, no. 16, pp. 13-17 (in Russian).

35. Maximov R. V., Sokolovsky S. P., Orekhov D. N. Ways of Detecting and Hiding Computer Network Proactive Security Tools Unmasking Features. *Sbornik trudov XXIV Mezhdunarodnoi nauchno-tekhnicheskoi konferentsii (RLNC 2018)* [Radar, navigation, communication (RLNC 2018)], Voronezh, 2018, pp. 169-179 (in Russian).

36. Vygovskij L. S., Maximov R. V. The Model of the Intentional Destructive Impact on the Integrated Communication Systems Infrastructure. *St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems*, 2008, vol. 3 no. 60, pp. 166-173 (in Russian).

37. Maximov R. V., Krupenin A. V., Sharifullin S. R., Sokolovsky S. P. Innovative Development of Tools and Technologies to Ensure the Russian Information Security and Core Protective Guidelines. *Voprosy kyberbezopasnosty*, 2019, vol. 1, no. 29, pp. 10-17, DOI: 10.21681/2311-3456-2019-1-10-17 (in Russian).

38. Maximov R. V., Berest P. A., Bogachev K. G., Vygovskij L. S., Ignatenko A. V., Kozhevnikov D. A., Krasnov V. A., Kuznecov V. E. *Sposob*

sravnitel'noj ocenki struktur informacionno-vychislitel'noj seti [Method of Comparative Evaluation of Information and Computer Network Structures]. Patent Russia, no. 2408928, 10.01.2011.

39. Maximov R. V., Kozhevnikov D. A., Pavlovskij A. V., Ur'ev D. U. *Sposob vybora bezopasnogo marshruta v seti svyazi (varianty)* [How to Select a Secure Route in the Communication Network (Options)]. Patent Russia, no. 2331158, 10.08.2008.

40. Internet Standard: RFC 793. Transmission Control Protocol. DARPA internet program protocol specification, 1981. Available at: https://tools.ietf.org/html/rfc793 (accessed 4 September 2019).

41. DRAFT STANDARD: RFC 5681. TCP Congestion Control. 2009. Available at: https://tools.ietf.org/html/rfc5681 (accessed 4 September 2019).

42. Makarenko S. I. *Spravochnik nauchnykh terminov i oboznachenii* [Handbook of Scientific Terms and Notation]. Saint Petersburg, Naukoemkie tekhnologii Publ, 2019. 254 p (in Russian).

43. Verzhbitskii V. M. *Osnovy chislennykh metodov* [Fundamentals of Numerical Methods]. Moscow, Vysshaya Shkola Publ, 2002. 840 p (in Russian).

44. Maximov R. V., Andrienko A. A., Kozhevnikov D. A., Kolbasova G. S., Pavlovskij A. V., Starodubcev U. I. *Sposob (varianty) i ustrojstvo (varianty) zashchity kanala svyazi vychislitel'noj seti* [Method (Options) and Device (Options) to Protect the Communication Channel of the Computer Network]. Patent Russia, no. 2306599, 20.09.2007.

45. Maximov R. V., Kozhevnikov D. A., Pavlovskij A. V. *Sposob zashchity vychislitel'noj seti (varianty)* [Method of Computer Network Protection (Options)]. Patent Russia, no. 2325694, 27.05.2008.

46. Maximov R. V., Vetoshkin I. S., Drozd YU. A., Efimov A. A., Ignatenko A. V., Kozhevnikov D. A., Krasnov V. A., Kuznecov V. E. *Sposob zashchity vychislitel'noj seti s vydelennym serverom* [Method of Protection of a Computer Network with a Dedicated Server]. Patent Russia, no. 2449361, 10.02.2011.

47. Maximov R. V., Vygovskij L. S., Zargarov I. A., Kozhevnikov D. A., Pavlovskij A. V., Starodubcev Yu. I., Hudajnazarov Yu. K., Yurov I. A. *Sposob (varianty) zashchity vychislitel'nyh setej* [Method (Options) of Protection of Computer Networks]. Patent Russia, no. 2307392, 27.09.2007.

48. Maximov R. V., Andrienko A. A., Kulikov O. E., Kostyrev A. L. *Sposob obnaruzheniya udalennyh atak na avtomatizirovannye sistemy upravleniya* [Method for Detecting Remote Attacks on Automated Control Systems]. Patent Russia, no. 2264649, 20.11.2005.

49. Maximov R. V., Golub B. V., Goryachaya A. V., Kozhevnikov D. A., Lykov N. U., Tihonov S. S. *Sposob maskirovaniya struktury seti svyazi* [A Method of Masking a Structure of a Communication Network]. Patent Russia, no. 2622842, 20.06.2017.

50. Grimes R. A. *Honeypots for Windows*. Apress, 2005, 424 p.

51. Maximov R. V., Sokolovsky S. P., Gavrilov L. A. Hiding computer network proactive security tools unmasking features. *Selected Papers of the VIII All-*

Russian Conference with International Participation «Secure Information Technologies» (BIT 2017). Moscow, Bauman Moscow Technical University, 2017, pp. 88-92 (in Russian).

52. Maximov R. V., Iskol'nyj B. B., Lazarev A. A., Lykov N. YU., Horev G. A., Sharifullin S. R. *Sposob sravnitel'noj ocenki struktur setej svyazi* [Method of Comparative Evaluation of Communication Network Structures]. Patent Russia, no. 2626099, 21.07.2017.

53. Iskolnyy B. B., Maximov R. V., Sharifullin S. R. Survivability Assessment of Distributed Information and Telecommunication Networks. *Selected Papers of the VIII All-Russian Conference with International Participation «Secure Information Technologies» (BIT 2017)*. Moscow, Bauman Moscow Technical University, 2017, pp. 59-65 (in Russian).

54. Maximov R. V., Ivanov I. I., Sharifullin S. R. Network Topology Masking in Distributed Information Systems. *Selected Papers of the VIII all-Russian Conference with International Participation «Secure Information Technologies»*. Moscow, Bauman Moscow Technical University, 2017, pp. 83-87 (in Russian).

55. Zausaev A. F. *Raznostnye metody resheniia obyknovennykh differentsial'nykh uravnenii. Uchebnoe posobie* [Difference Methods for Solving Ordinary Differential Equations]. Samara, Samara State Technical University, 2010. 100 p (in Russian).

56. Sokolovsky S. P., Telenga A. P., Voronchikhin I. S. Moving target defense for securing Distributed Information Systems. *Informatika. Problemy, metodologiia, tekhnologii. Sbornik materialov XIX mezhdunarodnoi nauchno-metodicheskoi konferentsii* [Informatics. Problems, methodology, technology]. *Voronezh*, 2019, pp. 9-643 (in Russian).

Статья поступила 10 октября 2019 г.

Информация об авторах

Максимов Роман Викторович – доктор технических наук, профессор. Профессор кафедры. Краснодарское высшее военное училище им. генерала армии С.М. Штеменко. Область научных интересов: обеспечение информационной безопасности; синтез и системный анализ систем защиты информации критически важных объектов; маскирование информационных ресурсов интегрированных ведомственных сетей связи. E–mail: rvmaxim@yandex.ru

Орехов Дмитрий Николаевич – соискатель ученой степени кандидата технических наук. Адъюнкт. Краснодарское высшее военное училище им. генерала армии С.М. Штеменко. Область научных интересов: обеспечение информационной безопасности; синтез и системный анализ систем защиты информации критически важных объектов; маскирование информационных ресурсов интегрированных ведомственных сетей связи. E–mail: orexov777@mail.ru

Соколовский Сергей Петрович – кандидат технических наук, доцент, докторант. Краснодарское высшее военное училище им. генерала армии

С.М. Штеменко. Область научных интересов: обеспечение информационной безопасности; синтез и системный анализ систем защиты информации критически важных объектов; маскирование информационных ресурсов интегрированных ведомственных сетей связи. E–mail: ssp.vrn@mail.ru
Адрес: 350063, Россия, г. Краснодар, улица Красина, д. 4.

Model and Algorithm of Client-Server Information System Functioning in Network Intelligence Conditions

R. V. Maximov, D. N. Orekhov, S. P. Sokolovsky

Purpose. *Capabilities enhancing and effectiveness improving of network intelligence to break client-server information systems actualize questions of systems resistance to destructive actions. Known methods of protection from network intelligence, based on the implementation of the principles of spatial security as well as on the formalization and integration of different prohibiting rules, which used computer attacks facts and network intelligence activity detection and reaction facts, can't be effective against modern network intelligence tools. Implementation of such protection methods allows the intruder to continue the impacts on client-server information systems and (or) to change strategy of influence.* ***Purpose of the work i****s to develop model and algorithm which provides both: operational service of the maximal amount of the authorized client-server information systems client requests and reduction of the quality of service of network intelligence tools requests.* ***Methods.*** *To solve the problems of client-server information system functioning in network intelligence conditions with different interacting forces strategies and controlling network intelligence tools resource capabilities, when the network connections are established and supported, the Markovian processes with discrete states and continuous time are used as models of network connection processes.* ***The novelty of the paper*** *is the use of the Markovian processes and the Kolmogorov equations solution to explore and to control the client-server information system resource capabilities by the network connection parameters operating. The novelty of the developed algorithm is the possibility of the client-server information system functioning model application in network intelligence tools resource capabilities controlling process when the network connections are established and supported.* ***Results.*** *The use of the presented solution for client-server information system resource capabilities dynamic control by the network connection parameters operating allows to increase the system security effectiveness. It is so, because of the security means detection probability reduction: the intruder can't detect moments when the security means are used and so he can't identify the characteristics of the means. It happens because of the two-way intruder-user connection holding time increasing, when the bad quality channel is imitated and also because of the intruder's network intelligence tools connection disconnect attempts blocking. Finding probability and time characteristics which describe client-server information system functioning process is the* ***practical relevance*** *of the work. Such process runs with different establishing and supporting strategies of connection parameters of interacting forces. The practical significance of the presented algorithm is to solve the problem of dynamic configuration of the network connections parameters of the client-server information system, which provides network intelligence traffic discrimination, concealment of the security means use and characteristics identification of such means.*

Key words: *client-server information system, computer attack, network connection, honeypots, network tarpits, protocol, network intelligence.*

Information about Authors

Roman Viktorovich Maximov – Dr. habil. of Engineering Sciences, Professor. Professor a Department. Krasnodar Higher Military School named after General of the Army S.M. Shtemenko. Field of research: information security; synthesis and system analysis of information security systems of critical objects; masking and

simulation of information resources of integrated departmental communication networks. E–mail: rvmaxim@yandex.ru

Dmitriy Nikolaevich Orekhov – applicant for academic degree of candidate of technical sciences. Post graduate student. Krasnodar Higher Military School named after General of the Army S.M. Shtemenko. Research interests: information security; synthesis and system analysis of information security systems of critical objects; masking and simulation of information resources of integrated departmental communication networks. E–mail: orexov777@mail.ru

Sergey Petrovich Sokolovsky – PhD. of Engineering Sciences, Associate Professor. Doctoral Candidate. Krasnodar Higher Military School named after General of the Army S.M. Shtemenko. Research interests: information security; synthesis and system analysis of information security systems of critical objects; masking and simulation of information resources of integrated departmental communication networks. E–mail: ssp.vrn@mail.ru

Address: 350063, Russia, Krasnodar, Krasina street, 4.