УДК 004.942

Концепция противодействия идентификации метаструктур корпоративных информационных систем на уровне перколяционных кластеров киберпространства

Максимов Р. В., Теленьга А. П.

Постановка задачи: *метаструктуры корпоративных информационных систем (КИС) представляют собой вторичные структуры, которые формируются под влиянием различных процессов и могут быть использованы для анализа и вскрытия процессов функционирования КИС в киберпространстве с использованием компьютерной разведки (КР). Задача противодействия идентификации метаструктур информационных систем на уровне перколяционных кластеров киберпространства может быть сформулирована в виде задачи многокритериальной оптимизации с использованием метрики перколяционной адаптивности в качестве критерия оптимальности маскирования.* ***Цель работы*** *заключается в том, чтобы модифицировать граф, представляющий информационную систему, путем добавления новых вершин и рёбер к его одномодовым проекциям таким образом, чтобы повысить его перколяционную адаптивность, т.е. снизить перколяционную центральность вершин для каждой проекции, уменьшить сходство между исходным и модифицированным графом, и при этом максимально увеличить размер связной компоненты путем добавления минимального количества вершин.* ***Используемые методы:*** *решение задачи противодействия идентификации метаструктур КИС на уровне перколяционных кластеров киберпространства основано на использовании разработанной методики противодействия идентификации, включающую алгоритмы маскирования и оценки результативности методом случайного блуждания для предложенной модели КИС на уровне перколяционных кластеров киберпространства, позволяющую методом оптимизации DIRECT определить оптимальные параметры маскирования.* ***Новизна:*** *элементом новизны представленного решения является использование показателя перколяционной центральности и введение новой совокупности действий и связей между ними для определения важности вершин графа и последующем формировании модифицированного графа с учетом этих критериев, что позволяет учитывать не только топологическую информацию графа, но также его перколяционную структуру для генерации новой вершинной и реберной информации.* ***Результат:*** *использование представленного решения по противодействию идентификации метаструктур КИС на уровне перколяционных кластеров киберпространства позволяет повысить его перколяционную адаптивность, т.е. снизить перколяционную центральность вершин для каждой проекции, уменьшить сходство между исходным и модифицированным графом, и при этом максимально увеличить размер связной компоненты путем добавления минимального количества вершин. Проведенное моделирование для оценки результативности маскирования методом случайных блужданий для набора графов с 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 вершинами показало снижение вероятности обхода вершин исходного графа в маскированном графе (т.е. вскрытии исходной структуры КИС) за 100 попыток для различных коэффициентов важности критериев на 40 %.* ***Практическая значимость:*** *представленное решение предлагается реализовать в виде математического обеспечения ложных сетевых информационных объектов. Это позволит создавать более адаптивные и эффективные механизмы противодействия идентификации метаструктур КИС в киберпространстве, увеличивая сложность анализа и обнаружения потенциальных угроз злоумышленником.*

Библиографическая ссылка на статью:
Максимов Р. В., Теленьга А. П. Концепция противодействия идентификации метаструктур корпоративных информационных систем на уровне перколяционных кластеров киберпространства // Системы управления, связи и безопасности. 2024. № 4. С. 179-222. DOI: 10.24412/2410-9916-2024-4-179-222

Reference for citation:
Maximov R. V., Telenga A. P. The Concept of Countering the Identification of Corporate Information Systems Metastructures at the Level of Percolation Clusters in Cyberspace. *Systems of Control, Communication and Security*, 2024, no. 4, pp. 179-222 (in Russian). DOI: 10.24412/2410-9916-2024-4-179-222

Ключевые слова: *метаструктура, корпоративные информационные системы, маскирование, перколяция, перколяционная адаптивность, киберпространство, перплексия, t-SNE.*

Введение

В последние годы авторы, как зарубежные [1], так и отечественные [2], вводят понятие киберпространства [3] как искусственного неоднородного технологического пространства, характеризующегося следующими ключевыми свойствами:

- множественность органов управления (киберпространство представляет собой систему с множеством разноуровневых органов оперативного и технологического управления);
- волатильность создания и эксплуатации (процесс создания и эксплуатации киберпространства не предопределяется требованиями одной системы управления, а функционирует в интересах множества разнородных систем управления);
- антагонистические отношения (киберпространство может иметь антагонистические отношения между внутренними и внешними потребителями);
- зависимость свойств киберпространства от характеристик элементов и процессов его компонентов (свойства киберпространства зависят как от характеристик собственных элементов, так и от объема и свойств реализуемых процессов в интересах внутренних и внешних потребителей).

Формирование структуры корпоративных информационных систем (КИС) в киберпространстве не является одномоментным процессом. Она динамически меняется под влиянием повседневной деятельности, внештатных ситуаций или преднамеренных информационно-технических воздействий.

Характеристика КИС как «мерцающей» во времени позволяет представить её структуру не только как двух- или трехмерные конструкции (граф, матрицу связности), но и как динамически меняющуюся сеть взаимосвязей. Каждый момент времени представляет собой «срез» КИС в данном конкретном состоянии.

Таким образом, традиционное представление структуры КИС как статического образа необходимо дополнить понятием динамики и изменчивости ее формирования.

Согласно концепции Киберкомандования США [4], в киберпространстве выделяются следующие уровни:

- киберперсонализации: уровень идентичности и атрибутов сущностей;
- логической сети: структура данных и информационных потоков;
- физической сети: инфраструктура и информационные технологии физических доменов.

Таким образом, в терминах логической модели информационной системы можно выделить следующие уровни:

– инфраструктура: базовые компоненты системы, включая вычислительные мощности, сеть и хранилище данных;
– апплиструктура: приложения информационной системы и сервисы, обслуживающие их;
– инфоструктура: данные и информация, содержимое баз данных, файловых хранилищ и т.д.

Эти уровни представляют собой логическую модель КИС в киберпространстве, позволяющую анализировать и понимать взаимосвязи между различными компонентами системы. Протоколы и механизмы, которые обеспечивают интерфейс между инфраструктурой, структурой приложений и структурой данных в информационной системе формируют закон группы, которую образуют разнородные структуры, высшей логической абстракцией которой является метаструктура.

Подобно «вторичным структурам» в геологии, возникающим в горной породе под влиянием позднейших процессов, например, механического, термального или химического воздействия, можно говорить о формировании в киберпространстве метаструктур КИС, которые могут обнаруживаться как информационные следы на соответствующем уровне киберпространства:

– статистический след – обнаруживается в логической сети как сетевой трафик КИС;
– семантический след – проявляется на уровне киберперсонализации, представляя собой служебную информацию операционных систем и приложений;
– структурный след – обнаруживается в физической сети как перколяционные процессы кластеров сетей передачи данных.

Эти метаструктуры представляют собой вторичные структуры, которые формируются под влиянием различных факторов и могут быть использованы для анализа и вскрытия процессов функционирования КИС в киберпространстве с использованием компьютерной разведки (КР) [5-7].

Конфликты в киберпространстве характеризуются тем, что все его участники имеют развитые системы мониторинга и наблюдения состояния антагониста, системы информационного воздействия, а также собственные защищаемые информационно-управляющие системы.

Методология описания этапов компьютерной атаки Cyber KillChain, разработанная компанией Lockheed Matrin, первым определяет этап разведки. База знаний тактик и техник злоумышленников MITRE ATT&CK выделяет 10 групп техник разведки, среди которых поиск по открытым источникам, активное сканирование, фишинг, поиск по закрытым источникам, агрегация информации о сетях, сотрудниках и организационной структуре объекта разведки.

Разнообразие этих техник и их комбинаций, а также возможность анализа полученной информации методами глубокого анализа данных, существенно повышает вероятность вскрытия структуры КИС, а значит, и осуществления деструктивного воздействия.

Статичность, однородность и детерминированность КИС обуславливают наличие у противника ряда преимуществ в использовании временного и вычислительного ресурса для ведения КР. Эти преимущества включают:

– возможность крупномасштабных атак – после успешного проведения мелкомасштабной атаки можно провести крупномасштабную атаку с небольшими ресурсными затратами;
– высокая достоверность результатов КР – высокая достоверность результатов КР в течение длительного времени позволяет осуществлять планирование, выбор времени и технологического процесса КИС для начала компьютерных атак (КА);
– бескомпроматное применение средств КР – возможность бескомпроматного применения средств КР и реализации КА в любое удобное время;
– возможность неоднократного обнаружения уязвимостей – возможность неоднократного обнаружения и анализа уязвимостей аппаратного и программного обеспечения с последующим их тестированием на проникновение для конкретной цели.

Эти преимущества обуславливают необходимость постоянных усилий по повышению безопасности КИС и своевременному реагированию на обнаруженные уязвимости. Для нейтрализации этих преимуществ в настоящее время в состав мер защиты информации в государственных информационных системах регулятором включены:

– сокрытие архитектуры и конфигурации КИС;
– создание (эмуляция) ложных КИС или их компонентов, предназначенных для обнаружения, регистрации и анализа действий нарушителей в процессе реализации угроз безопасности информации;
– воспроизведение ложных и (или) скрытие истинных отдельных информационных технологий и (или) структурно-функциональных характеристик КИС или ее сегментов, обеспечивающее навязывание нарушителю ложного представления об истинных информационных технологиях и (или) структурно-функциональных характеристиках КИС.

Задача реализации перечисленных мер защиты информации может быть решена противодействием идентификации метаструктур КИС на разных уровнях киберпространства путём маскирования [8] – парадигмы, позволяющей избегать конфликта как столкновения средств защиты и деструктивного информационно-технического воздействия, формируя на информационном пространстве противника выгодную для обеих сторон «картину мира».

Маскирование метаструктур КИС имеет целью заставить противника действовать таким образом, как это выгодно системе защиты с целью нивелирования конфликта [9]. Тактики, техники и процедуры повышения неопределенности активно скрывают существенные элементы реальной и ложной информации (производитель и версии используемого в КИС программного и аппаратного обеспечения, информационные направления между элементами КИС, технические возможности и замысел маскирования, перечень ложных

компонентов КИС), тактики, техники и процедуры введения в заблуждение, наоборот, раскрывают некоторые несущественные элементы реальной и ложной информации (раскрытие части структурно-функциональных характеристик КИС, модификация сетевого трафика ложными пакетами), формируя у противника неоднозначную интерпретацию добытой информации.

Концепция создания более динамичной и непредсказуемой среды для противника, затрудняющей сбор информации и атакующие действия, носит название «защита движущейся цели» (Moving Target Defense, MTD). В последние годы тематика MTD привлекла внимание как зарубежных, так и российских ученых благодаря своей перспективности в противодействии современным киберугрозам.

Так, в обзоре [10] рассматриваются подходы к MTD, а также их влияние на сетевую безопасность и практическое применение защитных механизмов. Авторы [11] анализируют ключевые характеристики MTD и его стратегическое применение, акцентируя внимание на изменении классической асимметрии между атакой и защитой. В [12] была исследована эффективность MTD, используя модели безопасности для оценки изменения векторов атаки при динамическом изменении параметров системы. В исследовании [13] MTD охарактеризована как система проактивной защиты, которая изменяет поверхность атаки, повышая сложность предсказания для злоумышленников.

В работе [14] предложена концепция передачи данных в сетях с динамической рандомизацией адресного пространства для повышения устойчивости к прослушиванию трафика и распределенным атакам типа «отказ в обслуживании». Первое техническое решение по манипулированию адресным пространством было предложено в [15] и [16].

Авторами [50] предложена теоретическая модель работы системы, которая диверсифицируется за счет введения динамических переменных.

В работе [51] сделано расширение технологии MTD к общей проблеме защиты систем от исследования и разработан алгоритм построения защищенных от исследования систем с неограниченным количеством дополнительных параметров, предложены решения для защиты веб-приложений.

Авторами [17] описывается подход к повышению устойчивости от распределенного отказа в обслуживании (Distributed Denial of Service, DDoS), которая достигается за счет внедрения новой адресной политики в IP-сетях, скрывающей физическое местоположение защищаемого сервера для всех неавторизованных клиентов путем динамического отображения адреса сервера на большой набор временных адресов.

В [18] рассмотрена методика маскирования информационного обмена в IP-сетях передачи данных в ходе реализации целевой атаки организованным нарушителем, которая включает в себя блок-схемы процедур реализаций генетического и биологического алгоритмов, формирование «информационных портретов», соответствующих передаваемой по IP-сети служебной информации при реализации технологических процессов.

В [19] рассмотрены методы и средства проактивной защиты программного обеспечения информационных систем специального назначения, позволяю-

щие при наличии в информационных системах программных средств скрытого информационного воздействия обнаруживать их и обходить участки кода с их наличием, не позволяя осуществить вредоносное информационно-техническое воздействие.

В рамках исследований [20-33] были разработаны и применены различные модели, методики и научно-технические предложения для повышения эффективности маскирования и защиты информационных систем от компьютерной разведки. Направления исследований и результаты перечислены ниже.

1. Маскирование информационных направлений распределённых инфокоммуникационных систем [20]:
– использование теории таксометрического анализа, теории графов и теории оптимального управления для оценки и снижения информативности демаскирующих признаков;
– выбор оптимальных параметров маскираторов и управление их ресурсами;
– определение оптимального количества IP-адресов и частоты их смены.
2. Маскирующий обмен в интегрированных сетях связи [21]:
– применение элементов теории графов и теории алгоритмов для оптимизации структуры сети маскирующего обмена;
– использование модифицированных алгоритмов построения минимальных остовных деревьев для управления демаскирующими признаками и снижения нагрузки на сетевые информационные объекты.
3. Имитация сетевых информационных объектов и защита вычислительных сетей [22]:
– применение иерархического кластерного анализа и симплекс-метода для выбора функционально-эквивалентных структур и распределения ложного трафика.
4. Функционирование клиент-серверных информационных систем [23]:
– использование теории марковских случайных процессов и уравнений Колмогорова для динамического управления ресурсами и устранения демаскирующих признаков систем защиты информации.
5. Динамическое управление параметрами маскированных каналов связи [24, 25]:
– применение теории марковских случайных процессов для управления параметрами маскированных каналов связи от деструктивных воздействий.
6. Конфигурирование параметров передачи данных в системах файлового обмена [26, 27]:
– оптимизация параметров передачи данных на основе теории марковских случайных процессов для повышения устойчивости и максимизации вероятности удержания средств компьютерной разведки.

7. Маскирование информационного обмена в сетях передачи данных [28]:
– применение теории марковских случайных процессов для мониторинга и адаптации структуры сети, снижения нагрузки на абонентов и обеспечения своевременности информационного обмена.
8. Конфигурирование структурно-функциональных характеристик информационных систем [29]:
– вычисление оптимальных параметров в пространстве IP-адресов, доменных имен и сетевых портов для повышения эффективности проактивной защиты информационных систем.
9. Оптимизация функционирования систем электронной почты [30]:
– использование однородных полумарковских процессов для моделирования почтовых сеансов и конфигурирования параметров передачи сообщений.
10. Оптимизация многоадресных соединений [31]:
– применение теории однородных полумарковских процессов и методов многокритериальной оптимизации для конфигурирования параметров многоадресных соединений.
11. Оптимизация функционирования ложных сетевых информационных объектов [32, 33]:
– оценка вероятностно-временных характеристик процессов и конфигурирование адресации для обеспечения реалистичности функционирования ложных сетевых информационных объектов.

В то же время, функционирование КИС в киберпространстве и выделение их метасруктур как целевых объектов разведки рассматривается впервые.

Анализ представленных публикаций свидетельствует о необходимости реализации противодействия идентификации через совокупность маскирующих техник, осуществляемых сетевыми информационными объектами (СИО) для управления демаскирующими характеристиками метаструктур КИС. Это включает управление интенсивностью трафика (в том числе генерацией маскирующего сетевого трафика, статистически схожего с реальным [33]) между топологически локализованными СИО распределенной КИС, изменением сетевых протоколов взаимодействия, а также иерархическими уровнями (рангами) компонентов КИС, добавлением ложных уязвимых элементов СИО (приманок, в зарубежной литературе – honeypot) для расширения поверхности атаки, сбора информации о противнике и применяемым им программным средствам информационно-технического воздействия, осуществляемым ложными СИО (обманными системами, в зарубежной литературе – deception system).

Условия функционирования КИС требует решения следующих задач противодействия идентификации.

1. Необходимо дополнить метаструктуры КИС до метаструктур киберпространства («внешней сети» – сети связи общего пользования), иными словами, необходима *маскировка* метаструктур КИС под метаструктуры киберпространства, поскольку разнообразие уникальных цифровых отпечатков

устройств КИС – набора параметров, позволяющих однозначно идентифицировать устройство пользователя – несоизмеримо меньше, чем устройств киберпространства.

2. Необходимо «*отравить*» (насытить) метаструктуру КИС ложными данными для снижения эффективности средств КР. В этом случае применение противником, например, методов машинного обучения для ведения КР будет существенно затруднено.

3. Необходимо *имитировать* метаструктуру КИС для обеспечения успешного киберманеврирования. Сущность киберманевра заключается в искусственном расширении поверхности атаки за счёт создания ложных целей.

4. Необходима *мимикрия* метаструктур киберпространства под метаструктуры КИС с целью введения в заблуждение КР и отвлечения внимания.

При этом перечисленные выше задачи могут решаться как по отдельности, так и совместно, образуя комплекс средств маскирования.

Содержательная постановка задачи противодействия идентификации метаструктур КИС на уровне перколяционных кластеров киберпространства

На уровне физической сети киберпространства характерно проявление метаструктур КИС в виде перколяционных кластеров. В физике и химии явлением перколяции (от лат. percōlāre – просачиваться, протекать) называется явление протекания жидкостей через пористые материалы. Теория перколяции [34, 35] находит применение в описании разнообразных систем и явлений, в том числе таких, как распространение эпидемий, надежность компьютерных сетей и протекание электричества через смесь проводящих и непроводящих частиц. Совокупность элементов, по которым происходит протекание, называется перколяционным или стягивающим кластером.

Для произвольной точки инжекции перколяция происходит только вдоль остова (backbone) перколяционного кластера. Части перколяционного кластера, связанные с его остовом через единственный узел, называются обособленными ветвями или висящими узлами. Чтобы отделить обособленную ветвь от остова, достаточно удалить этот единственный узел, который называется рассекающим. Остов включает все узлы, лежащие на всех возможных траекториях случайного блуждания без самопересечений, начинающихся в узле (узлах) инжекции и заканчивающихся на границе области [34].

Обнаружение перколяционных кластеров КИС позволяет противнику наблюдать весь трафик, передаваемый в сети, тем самым решая задачу оптимального размещения средств компьютерной разведки, а также выявлять рассекающие узлы, атака на которые разрушает связность кластера и приводит сеть в неработоспособное состояние.

Теория перколяции позволяет решить задачи узлов и связей для сетей с различной как регулярной (2*D* структурой – треугольной, шестиугольной, деревьях Кейли, и т.д. и 3*D* – гексагональной, кубической и т.д.), так и случайной структурой. При решении задачи связей можно, например, определить долю

связей, которую нужно разорвать, чтобы сеть распалась на несвязанные части, или образовались кластеры блокированных узлов, а в задаче узлов, например, можно определить их долю, при которой сеть распадется на несвязанные между собой части. Доля блокированных узлов (в задаче узлов) или связей (в задаче связей), при которой исчезает проводимость между двумя произвольно выбранными узлами сети в целом, называется порогом перколяции (протекания). Кроме того, возможно измерить степень влияния соседних узлов и связей между ними, называемой затуханием инжекции, на скорость перколяции.

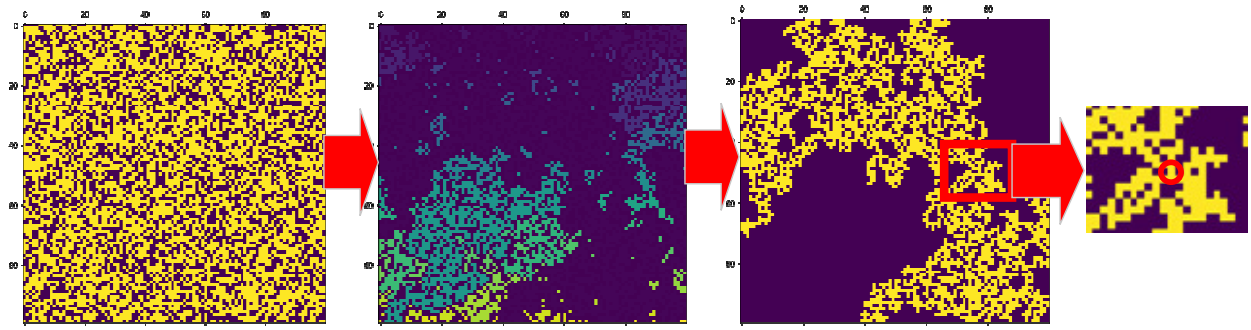

Рис. 1. Выделение перколяционных кластеров и рассекающих узлов на регулярной решетке

Задача защиты на этом уровне киберпространства заключается в насыщении перколяционного кластера ложными связями для защиты рассекающих узлов. Для решения сформулированной задачи необходимо иметь критерий значимости узлов перколяционных кластеров.

Очевидно, например, что в связном графе перколяция существует по определению. В теории графов мера важности вершин называется показателем центральности или близости к центру.

В дополнение к уже существующим мерам центральности [36], таким как центральность по степени, центральность собственного вектора, центральность по посредничеству, в 2013 году была предложена новая мера – перколяционная центральность [37], которая определяется для данного узла в данный момент времени как доля «перколированных путей», проходящих через этот узел. «Перколированный путь» – это кратчайший путь между парой узлов, в которой исходящий узел является перколированным. Перколяционная центральность вершины v в момент времени t в графе рассчитывается по формуле:

$$C^t(v) = \frac{1}{(N-2)} \sum_s \sum_r \left(\frac{\sigma_{s,r}(v)}{\sigma_{s,r}} \frac{x_s^t}{\left[\sum_i x_i^t \right] - x_v^t} \right),$$

где $\sigma_{s,r}$ – количество кратчайших путей между источником s и приемником r, $\sigma_{s,r}(v)$ – количество кратчайших путей между источником s и приемником r, проходящих через вершину v, x_i^t – состояние перколяции вершины i в момент

№4. 2024
ISSN 2410-9916

времени t, N – количество вершин, $w_{s,v}^t = \frac{x_s^t}{\left[\sum_i x_i^t\right] - x_v^t}$ – относительный вклад (вес) каждого перколяционного пути из источника s в меру перколяционной центральности $C(v)$.

В этом выражении суммирование производится по всем возможным парам вершин s и r графа, таким, что:

– $s \neq v$, исходящий узел s не может быть узлом v, так как интересуют пути, на которых v является промежуточным узлом;
– $r \neq v$, аналогично, конечный узел r не может совпадать с v;
– $s \neq r$, путь должен быть между разными узлами, т.е. петли в графе не учитываются.

Для поиска рассекающих узлов в произвольном графе можно использовать представление исходной структуры в виде регулярной решетки, представленной на рис. 1, из которой удалены те вершины, для которых перколяционность узлов исходной структуры равна 0.

На рис. 2 представлены перколяционный кластер, остов перколяционного кластера и перколяционный путь между висящими вершинами части графа произвольной структуры, представленной в виде регулярной решетки.

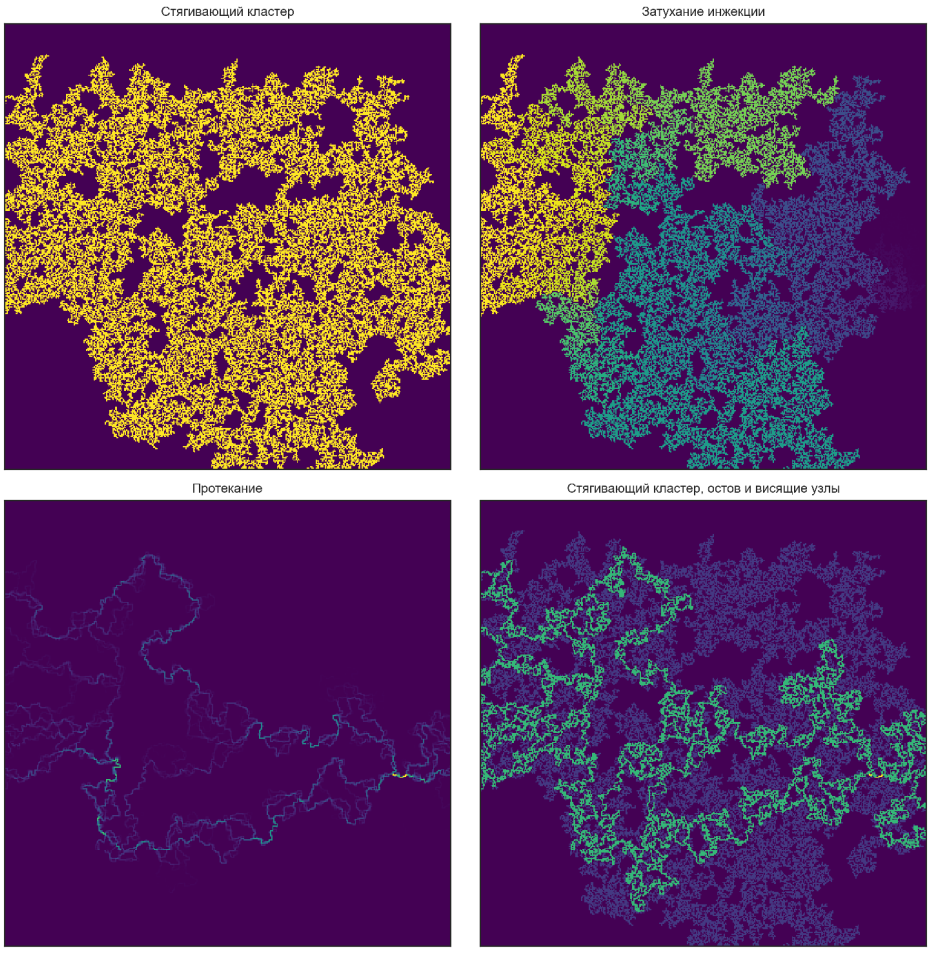

Рис. 2. Стягивающий кластер, затухание инжекции, остов и перколяционный путь части графа произвольной структуры

DOI: 10.24411/2410-9916-2024-4-179-222
URL: https://sccs.intelgr.com/archive/2024-04/07-Maximov.pdf

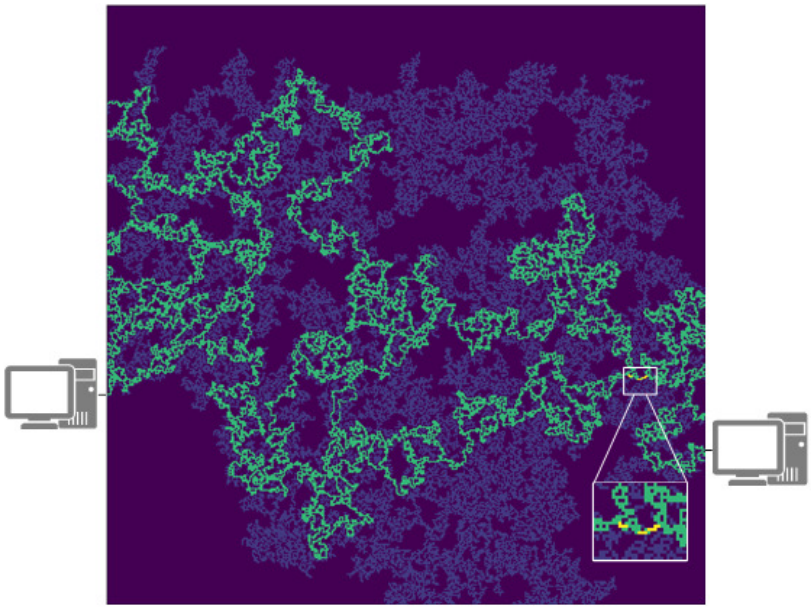

Рис. 3. Рассекающие узлы регулярной решетки

Легко сделать вывод, что через рассекающие узлы проходит наибольшее количество перколяционных путей, другими словами, признаком рассекающего узла является максимальность его перколяционной центральности (рис. 3):

$$v^{рассек} = v : C(v) = \max(C(i)), i = \overline{1, N}.$$

Модель КИС на уровне перколяционных кластеров киберпространства

Мы рассматриваем произвольную структуру КИС на уровне перколяционных кластеров киберпространства как темпоральную сеть с изменениями топологии вследствие отказов или отключений узлов, вызванных различными сбоями в работе узлов, а также корректировками ребер в результате применения стратегий адаптации маскирования.

Предполагается, что в заданной темпоральной сети, обозначаемой $G(t)$, в момент времени t узлы КИС распределены по нескольким группам задач, причем каждая группа, обозначаемая g_k, предоставляет услугу k. Начальное множество узлов и задач фиксировано, т.е. в сети не появляется новых узлов и задач. Набор всех активных групп задач может меняться с течением времени, так как некоторые группы могут выходить из строя (т.е. группа исчезает, если в ней нет ни одного члена), этот набор в момент времени t обозначается $M(t)$. Набор групп, в которые входит узел i в момент времени t, обозначается $M_i(t)$ для $i = 1,\ldots,N$, где i – идентификатор узла, а N – общее число узлов в сети. Узел i может участвовать в нескольких группах задач, но его максимальный уровень ресурса ограничен значением r_i. Для простоты мы абстрагируемся от r_i, чтобы считать максимальным уровень вычислительных и коммуникационных ресурсов, которые может использовать узел i, для всех задач. В зависимости от коли-

№4. 2024
ISSN 2410-9916

чества групп, в которых участвует узел i, уровень ресурсов узла i будет меняться и может быть выражен в терминах используемых и неиспользуемых ресурсов, обозначаемых $\left(r_i', r_i''\right)$, где $r_i = r_i' + r_i''$, а $R(t) = \{r_1, ..., r_N\}$ – набор ресурсов для всех узлов в момент времени t. Таким образом, $G(t)$ может быть представлена в виде $G(t) = \left(V(t), E(t), R(t)\right)$.

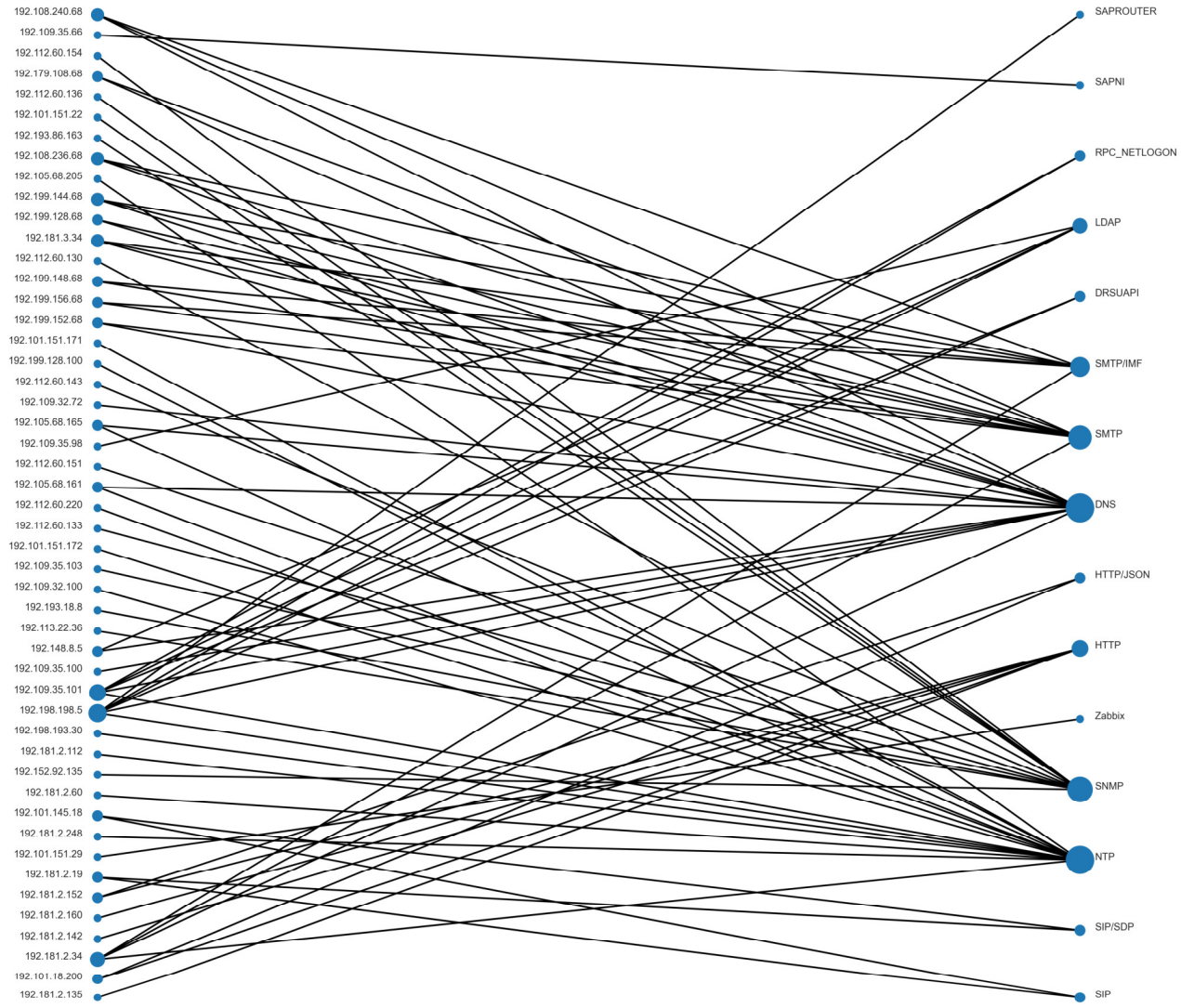

Рис. 4. Двудольный граф КИС с узлами и ассоциированными с ними группами задач

Мы можем представить эту сеть в виде двудольного графа, в котором существует два типа вершин, представляющих соответственно узлы сети и группы задач, выраженных протоколами прикладного уровня модели взаимодействия открытых систем OSI. На рис. 4 показан пример двудольного графа, в котором каждый узел может принадлежать к нескольким группам задач, а члены групп связаны друг с другом для предоставления услуг на основе сотрудничества. На рис. 5 показана одномодовая проекция двудольного графа на пространство узлов, где узлы связаны друг с другом на основе общих групп, в которые они входят, образуя клику. Однако однозначное обратное преобразова-

DOI: 10.24411/2410-9916-2024-4-179-222
URL: https://sccs.intelgr.com/archive/2024-04/07-Maximov.pdf

ние графа на рис. 5 в двудольный граф на рис. 4 выполнить невозможно, поскольку может существовать несколько групп, имеющих одинаковый набор членов, или пара узлов может иметь несколько общих групп.

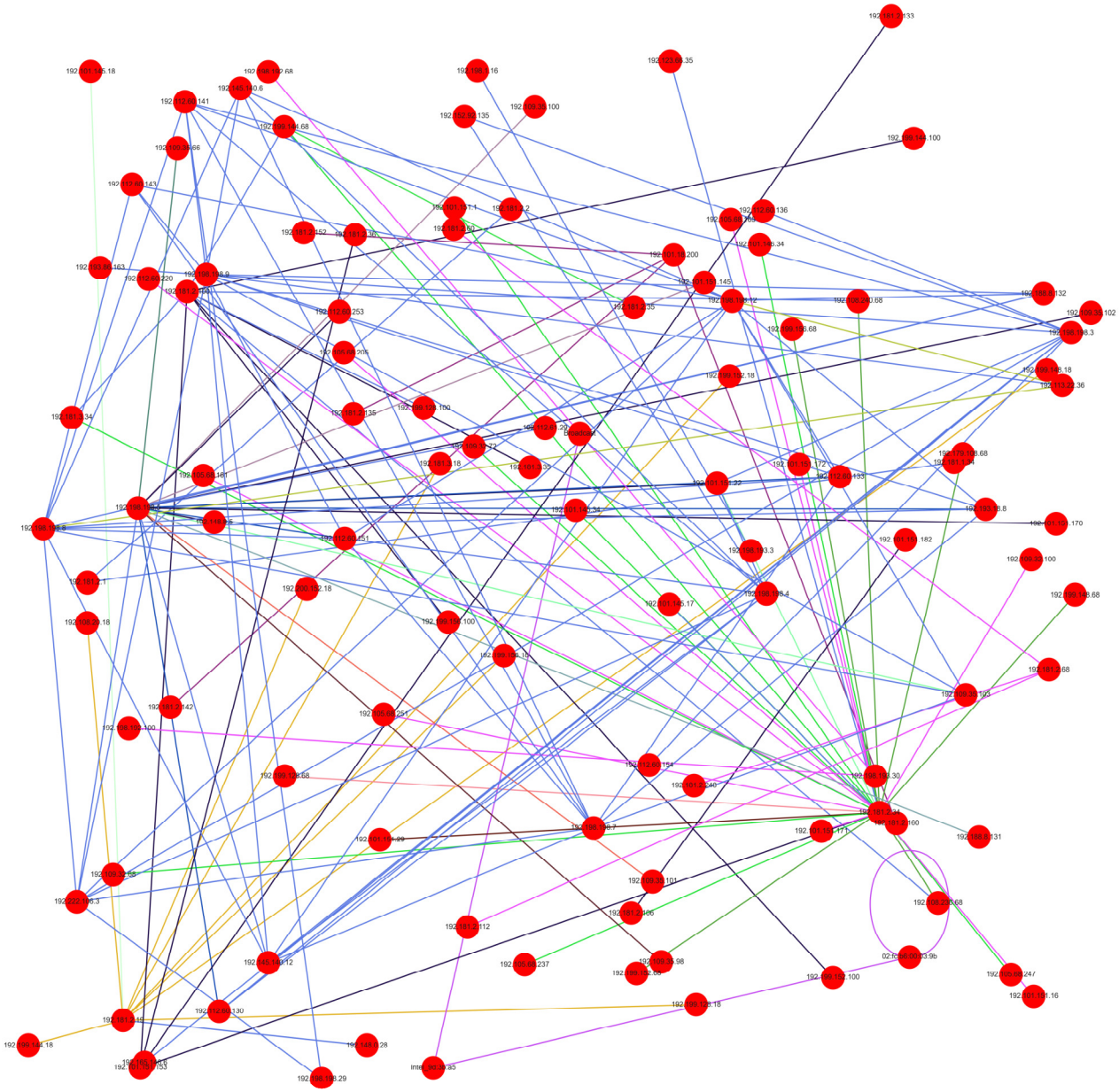

Рис. 5. Топология графа на основе общих связанных групп

Таким образом, общая стратегия маскирования может заключаться в следующем. Для исходной темпоральной сети, представленной двудольным графом $G(t) = \left(V(t), E(t), R(t)\right)$ строится множество одномодовых проекций двудольного графа на пространство узлов на основе протоколов прикладного уровня модели OSI, для каждой проекции с учётом перколяционной центральности вершин выполняется добавление ложных узлов и рёбер, затем формируется новый двудольный граф $G^*(t) = \left(V^*(t), E^*(t), R^*(t)\right)$, для которого оцениваются количество добавленных узлов, размер гигантской перколяционной компоненты добавленных вершин (т.е. максимальное количество вершин в

связном подграфе модифицированного графа) и схожесть полученной структуры с исходной [38-41].

Формализованная постановка задачи противодействия идентификации метаструктур информационных систем

В качестве метрик маскирования выступают перколяционная центральность вершин исходного графа (одномодовой проекции $G(t) = \left(V(t), E(t), R(t)\right)$ на пространство узлов), количество добавленных узлов, размер гигантской перколяционной компоненты добавленных вершин и коэффициент сходства исходного и маскированного графов.

Оценка эффективности маскирования сводится к свертке метрик, которую будем называть *перколяционной адаптивностью* графа.

Пусть $F_{\text{perc\_adapt}}(r, c_p)$ – целевая функция маскирования, зависящая от двух параметров:

- r – количества доступных ресурсов, выраженных количеством вершин, которые можно добавить в граф;
- c_p – перколяционной центральности вершин графа, к которым будут добавляться ребра от новых вершин.

Цель методики противодействия идентификации заключается в том, чтобы модифицировать граф $G(t) = \left(V(t), E(t), R(t)\right)$, представляющий информационную систему, путем добавления новых вершин и рёбер к его одномодовым проекциям таким образом, чтобы повысить его перколяционную адаптивность, т.е. снизить перколяционную центральность вершин для каждой проекции, уменьшить сходство между исходным и модифицированным графом, и при этом максимально увеличить размер связной компоненты путем добавления минимального количества вершин.

Таким образом, задача противодействия идентификации метаструктур информационных систем на уровне перколяционных кластеров киберпространства может быть сформулирована в виде задачи многокритериальной оптимизации с использованием метрики перколяционной адаптивности в качестве критерия оптимальности маскирования. Методика направлена на обеспечение высокого уровня безопасности КИС с учетом их гибкости и адаптивности к изменяющимся условиям.

Математическая постановка задачи оптимизации

В многокритериальной оптимизации важно учитывать несколько критериев одновременно, чтобы найти решение, которое удовлетворяет максимальному числу требований или целей. Преимущества взвешенной суммы критериев в этом контексте включают следующее.

Гибкость – взвешенная сумма позволяет задавать различные веса для каждого критерия в зависимости от их относительной важности. Это позволяет

учитывать предпочтения принимающего решение и адаптировать оптимизацию под конкретные потребности.

Простота интерпретации – после определения весов каждого критерия взвешенная сумма становится простой и понятной формой объединения критериев. Результаты такой оптимизации легко интерпретировать и использовать для принятия решений.

Вычислительная эффективность – взвешенная сумма обычно является вычислительно эффективной операцией, что обеспечивает быструю оценку решений и облегчает работу с моделями многокритериальной оптимизации.

Устойчивость к шуму – в случае наличия шума или неопределенности в данных использование взвешенной суммы может помочь уменьшить влияние этих факторов, так как веса могут быть настроены таким образом, чтобы учитывать эту неопределенность.

С учётом вышесказанного, целевая функция задачи маскирования метаструктур информационных систем на уровне физической сети киберпространства представляет собой свертку критериев

$$F_{\text{perc\_adapt}}(x) = w_{\text{added\_nodes}} \cdot \left(-N_{\text{added}} / r_{\max}\right) + w_{\text{max\_centrality}} \cdot \left(-C_{\max}\right) + \\ + w_{\text{giant}} \cdot G_{\text{giant}} / r_{\max} + w_{\text{jaccard}} \cdot \left(-J_{\text{jaccard}}\right) \quad (1)$$

где:

N_{added} – количество добавленных вершин, $N_{\text{added}} \to \min$;

$C_{\max}$ – максимальная перколяционная центральность вершины, к которой добавляется новое ребро, $C_{\max} \to \min$;

G_{giant} – размер максимальной связной компоненты добавленных вершин, $G_{\text{giant}} \to \max$;

J_{jaccard} – коэффициент сходства Жаккарда между исходным и маскированным графом, $J_{\text{jaccard}} \to \min$, $J_{jaccard} = \dfrac{\left|G_{orig} \cap G_{mask}\right|}{\left|G_{orig} \cup G_{mask}\right|}$, где G_{orig} – исходный граф, G_{mask} – результирующий маскированный граф;

$\left(w_{\text{added\_nodes}}, w_{\text{max\_centrality}}, w_{\text{giant}}, w_{\text{jaccard}}\right)$ – веса, определяющие важность каждого критерия, $\sum_i w_i = 1$;

$r_{\max}$ – максимальное количество вершин, которое может быть добавлено в модифицированный граф;

$x = \left(r, c_p\right)$ – параметры маскирования, включающие количество добавленных вершин r и перколяционную центральность c_p для добавления вершин.

Задача оптимизации состоит в максимизации целевой функции

$$F^{\text{norm}}_{\text{perc\_adapt}}(x) \xrightarrow[x \in \Psi^{\text{perc}}]{} \max \quad (2)$$

где:

$$F^{\text{norm}}_{\text{perc\_adapt}}(x) = \frac{F_{\text{perc\_adapt}}(x) - F^{\min}_{\text{perc\_adapt}}(x)}{F^{\max}_{\text{perc\_adapt}}(x) - F^{\min}_{\text{perc\_adapt}}(x)}, \quad (3)$$

$F^{\min}_{\text{perc\_adapt}}(x)$ – минимальное значение целевой функции,

$F^{\max}_{\text{perc\_adapt}}(x)$ – максимальное значение целевой функции,

при ограничениях

$$\Psi^{\text{perc}} = \begin{cases} 2 \le r \le r_{\max} \\ 0{,}01 \le c_p \le c_p^{\max} \\ N_{\text{added}} \le r_{\max} \\ C_{\max} \le c_p^{\max} \\ G_{\text{giant}} > 0 \\ J_{\text{jaccard}} < 1 \\ \sum_{i=1}^{4} w_i = 1 \end{cases}, \quad (4)$$

где $c_p^{\max}$ – максимальное значение перколяционной центральности вершины, к которой добавляется новое ребро.

Для решения поставленной задачи можно применять различные методы многокритериальной оптимизации и методы оптимизации с ограничениями. Некоторые из наиболее часто используемых перечислены далее.

Методы градиентного спуска с ограничениями: методы градиентного спуска, такие как метод штрафных функций или метод проекции градиента, могут быть адаптированы для решения задач с ограничениями. Они могут использоваться для поиска локального оптимального решения.

Генетические алгоритмы являются эффективным методом для решения многокритериальных задач оптимизации. Они могут использоваться для поиска набора парето-оптимальных решений.

Методы эволюционной оптимизации: эволюционные стратегии, генетическое программирование и другие методы эволюционной оптимизации могут быть применены для поиска оптимальных решений в многокритериальных задачах.

Методы оптимизации с использованием алгоритмов интеллектуального поиска: методы, такие как алгоритмы роя частиц, алгоритмы оптимизации колонии муравьев, могут быть использованы для решения сложных задач оптимизации с ограничениями.

Методы оптимизации на основе метаэвристик: метаэвристические методы, такие как имитации отжига, поиск с запретами, могут быть эффективны для решения задач оптимизации с ограничениями и множеством критериев.

Методы оптимизации на основе машинного обучения: некоторые методы машинного обучения, такие как алгоритмы усиления и алгоритмы оптимизации с подкреплением, могут быть адаптированы для решения задач оптимизации с ограничениями и многокритериальных задач.

Выбор конкретного метода оптимизации зависит от доступных ресурсов и требуемой точности решения.

Алгоритм противодействия идентификации метаструктур информационных систем на уровне перколяционных кластеров

Для реализации алгоритма противодействия идентификации метаструктур необходимы следующие исходные данные.

Исходный граф одномодовой проекции двудольного графа. Граф должен содержать информацию о всех узлах и связях внутри проекции.

Порог перколяционной центральности (c_p): пороговое значение центральности, используемое для определения рабочих узлов. Узлы с центральностью выше или равной этому значению будут участвовать в маскировании.

Количество доступных ресурсов (r): количество новых узлов, которые могут быть добавлены в граф для создания дополнительных связей и усложнения структуры.

Описание шагов алгоритма.

Шаги 3-8. Получение списка вершин для маскирования – если граф G имеет менее трех вершин, то рабочими узлами (переменная *working_nodes*) становятся все вершины графа. В противном случае, рассчитывается перколяционная центральность (переменная *percolation_centralities*) каждой вершины (*node*) графа G и только те вершины, у которых центральность больше или равна заданной пороговой величине c_p, становятся рабочими узлами, т.е. помещаются в массив *working_nodes*.

Шаги 9-11. Генерация новых узлов и рёбер – если рабочие узлы отсутствуют, то новые узлы (переменная *new_nodes*) генерируются путём добавления к каждому существующему узлу нового номера.

Шаг 12. Обновление списка вершин для маскирования – в список рабочих узлов (переменная *working_nodes*) добавляются новые узлы (переменная *new_nodes*).

Шаг 13. Выбор источника – случайно выбирается набор из r вершин из рабочих узлов (переменная *working_nodes*) в качестве источника (переменная *source_nodes*).

Шаги 14-17. Определение ближайших соседей (переменная *neighbors*) – определяются все вершины графа, связанные с источником и имеющие перколяционную центральность (переменная *percolation_centralities*), не равную нулю.

Шаги 18-20. Добавление рёбер – добавляется ребро (переменная *edge*) между источником и каждым из ближайших соседей.

Шаг 21. Удаление петель – удаляются все петли в графе, включая те, которые были добавлены на предыдущем шаге.

Шаг 22. Удаление узлов без рёбер – удаляются все вершины графа, не имеющие никаких рёбер.

Шаг 23. Возврат модифицированного графа – возвращается модифицированный граф с новой структурой и убранными петлями и узлами без рёбер.

Псевдокод алгоритма представлен на рис. 6.

Algorithm 1 Маскирование графа

```
Вход: Граф G, параметры маскирования (r, c_p)
Выход: Модифицированный граф G'
G' ← G.copy()
if |G'| < 3 then
    work_nodes ← {все вершины G'}
else
    percolation_centralities ← count_percolation_centrality(G')
    work_nodes ← {node | percolation_centralities[node] ≥ c_p}
if work_nodes = ∅ then
    work_nodes ← generate_new_nodes(G', r)
new_nodes ← generate_new_nodes(G', r)
work_nodes ← work_nodes ∪ new_nodes
source_nodes ← random_choice(work_nodes, r)
edges ← {}
for source ∈ source_nodes do
    neighbors ← {node | (node ∈ G'.neighbors(source)) ∧ (percolation_centralities[node] ≠ 0)}
    for neighbor ∈ neighbors do
        if source ≠ neighbor then
            edges ← edges ∪ {(source, neighbor)}
G'.add_edges_from(edges)
G'.remove_selfloops()
nodes_to_remove ← {n | G'.degree(n) = 0}
G'.remove_nodes_from(nodes_to_remove)

return G'
```

Рис. 6. Псевдокод алгоритма маскирования графа

Анализ временной сложности алгоритма противодействия идентификации метаструктур информационных систем на уровне перколяционных кластеров

Для анализа временной сложности алгоритма, представленного на рис. 6 необходимо оценить верхнюю асимптотическую сложность, используя общепринятый подход О-нотации.

1. Строка 3: Создание копии графа G. Время копирования графа зависит от количества вершин $|V|$ и рёбер $|E|$, что даёт временную сложность $O(|V|+|E|)$.

2. Строки 4–7: Проверка размера графа и вычисление перколяционной центральности.

Если $|V|<3$, операция тривиальна и имеет сложность $O(1)$.

Если $|V|\geq 3$, необходимо вычислить перколяционную центральность для каждой вершины графа, что является сложной операцией. В зависимости от алгоритма, вычисление перколяционной центральности может иметь сложность

$O(|V|^2 + |V||E|)$ в худшем случае, что характерно для многих реализаций алгоритмов центральности.

3. Строки 8–10: Если множество рабочих узлов пусто, генерируются новые узлы с помощью функции *generate_new_nodes*. Время выполнения этой функции зависит от параметра r, так как новые узлы добавляются в зависимости от его значения. Предположим, что сложность генерации новых узлов $O(r)$.

4. Строки 11–13: Объединение рабочих узлов с новыми узлами и случайный выбор исходных узлов. Операция объединения и случайного выбора имеет сложность $O(r)$, так как количество узлов ограничено значением параметра r.

5. Строки 14–21: Этот блок кода содержит два вложенных цикла:

– внешний цикл перебирает исходные узлы из множества $source\_nodes$, их количество ограничено r, что даёт сложность $O(r)$;

– внутренний цикл перебирает соседей каждого исходного узла. В худшем случае количество соседей может быть $O(|V|)$, что приводит к сложности $O(r \cdot |V|)$ для внутреннего цикла.

6. Строка 22: Добавление рёбер в граф. Эта операция имеет сложность $O(|E|)$, так как добавление всех рёбер может занять время, пропорциональное количеству рёбер.

7. Строка 23: Удаление петель (*self-loops*). В худшем случае требуется проверка каждого ребра, что приводит к сложности $O(|E|)$.

8. Строки 24–25: Удаление вершин с нулевой степенью. Для каждой вершины графа проверяется её степень, что имеет сложность $O(|V|)$.

Итоговая временная сложность:

основные временные затраты связаны с вычислением перколяционной центральности $O(|V|^2 + |V||E|)$ и вложенными циклами при работе с рёбрами и узлами $O(r \cdot |V|)$;

сложность копирования графа и других операций с рёбрами и вершинами также добавляет значимость к общей сложности.

Таким образом, временную сложность алгоритма в худшем случае можно выразить как:

$$O(|V|^2 + |V||E| + r \cdot |V|),$$

где:

$|V|$ – количество вершин в графе,

$|E|$ – количество рёбер в графе,

r – параметр, контролирующий количество узлов для работы и добавления рёбер.

Выбор метода оптимизации

Для выбора наиболее эффективного метода оптимизации было проведено сравнение следующих методов: двойной отжиг (dual_annealing) [42, 43], дифференциальная эволюция (differential_evolution) [44], глобальная оптимизация симплициальной гомологии (shgo) [45], DIRECT (direct) [46]. Сравнение эффективности алгоритмов производилось по критерию точности и временной сложности.

Псевдокод алгоритма сравнения методов оптимизации представлен на рис. 7.

Algorithm 2 Сравнение методов оптимизации

Вход: Граф G, список алгоритмов $algorithms$, предельные значения r_{max}, c_p^{max}, весовые коэффициенты w
Выход: Имя метода оптимизации $algorithm\_info["name"]$, оптимальные параметры r, c_p, оптимальное значение $optimal\_value$ целевой функции для $algorithm\_info["name"]$, время выполнения расчётов $elapsed\_time$
$algorithms \leftarrow$ [
{ "name": "dual_annealing "algorithm": dual_annealing },
{ "name": "differential_evolution "algorithm": differential_evolution },
{ "name": "shgo "algorithm": shgo },
{ "name": "direct "algorithm": direct }
]
for $algorithm\_info \in algorithms$ do
    $start\_time \leftarrow$ текущее время
    $r \leftarrow algorithm\_info["algorithm"](\text{create\_optimization\_problem}(G, w), bounds = [(1, r_{max}), (0.01, c_p^{max})])$
    $elapsed\_time \leftarrow$ текущее время - $start\_time$
    $optimal\_value \leftarrow -r, c_p$
    if $'best\_optimal\_value' \in locals()$ then
        $accuracy\_diff \leftarrow best\_optimal\_value - optimal\_value$
    $best\_optimal\_value \leftarrow optimal\_value$

return $algorithm\_info["name"]$, r, c_p, $optimal\_value$, $elapsed\_time$

Рис. 7. – Псевдокод алгоритма сравнения методов оптимизации

Алгоритм сравнения методов оптимизации получает на вход:

исходный граф G – список вершин и соединяющих их рёбер;

список $algorithms$ – перечень методов оптимизации, которые будут сравниваться, каждый метод представлен в виде словаря с его именем и самой функцией метода;

r_{max} и c_p^{max} – границы поиска оптимальных параметров;

w – параметры, влияющие на важность различных компонентов целевой функции.

В алгоритме используются локальные переменные:

$algorithm\_info$ – объект, содержащий информацию о текущем методе оптимизации (его название и реализация);

$elapsed\_time$ – время, затраченное алгоритмом на выполнение своей задачи, что позволяет сравнить эффективность методов по скорости;

optimal _value – оптимальное значение целевой функции, найденное алгоритмом;

best _optimal _value – лучшее из всех найденных значений целевой функции, используемое для сравнения алгоритмов;

accuracy _diff – разница между текущим и лучшим значением целевой функции, показывающая, насколько новый метод лучше или хуже предыдущего.

На выходе алгоритма получается имя метода, оптимальное значение целевой функции, значения оптимальных параметров целевой функции r и c_p, время поиска оптимального решения.

Этапы алгоритма:

1. Инициализация: подготавливается список алгоритмов, а также их параметры, включая границы поиска для оптимизируемых переменных.

2. Цикл по алгоритмам:

– каждый метод оптимизации запускается с задачей оптимизации, созданной на основе графа G и весов w;

– алгоритм получает параметры r и c_p, оптимизируя целевую функцию;

– вычисляется время выполнения алгоритма;

– записывается значение целевой функции, полученное алгоритмом.

3. Сравнение результатов:

– значение целевой функции сравнивается с предыдущими результатами для определения наилучшего метода;

– если текущий алгоритм нашёл лучшее значение, оно сохраняется.

4. Вывод результатов: возвращаются имя лучшего алгоритма, его оптимальные параметры, найденное значение целевой функции и время выполнения.

Результаты сравнения методов оптимизации на тестовых данных графа из 68 вершин и 85 ребер представлены в таблице 1.

Из таблицы 1 видно, что наилучшие результаты показал метод DIRECT.

DIviding RECTangles (DIRECT) – это детерминированный алгоритм глобальной оптимизации, способный минимизировать функцию «черного ящика», переменные которой подчиняются ограничениям нижней и верхней границы, путем выборки потенциальных решений в пространстве поиска [46]. Алгоритм начинается с нормализации пространства поиска до n-мерного единичного гиперкуба. Он производит выборку функции в центре этого гиперкуба и еще в 2n (n – количество переменных) точках, по 2 в каждом направлении координат. Используя эти значения функции, DIRECT затем делит область на гиперпрямоугольники, каждый из которых имеет ровно одну точку выборки в качестве центра. На каждой итерации DIRECT выбирает, используя параметр *eps*, который по умолчанию равен 10^{-4}, некоторые из существующих гиперпрямоугольников для дальнейшего разделения. Этот процесс деления продолжается до тех пор, пока либо не будет превышено максимальное количество итераций или максимальных разрешенных оценок функции, либо до тех пор, пока гиперпрямоугольник, содержащий найденное минимальное значение, не станет достаточно маленьким. Если указано значение *f_min*, оптимизация прекратится, как

только это значение функции будет достигнуто в пределах относительного допуска. По умолчанию используется локально смещенный вариант DIRECT (первоначально называвшийся DIRECT_L) [47]. Это делает поиск более локально смещенным и более эффективным для случаев, когда имеется всего несколько локальных минимумов.

Таблица 1 – Сравнение методов оптимизации

Название метода	Преимущества	Недостатки	Значение целевой функции	Время поиска оптимального решения
dual_annealing	Способен избегать локальных минимумов за счет использования обобщенной энтропии.	Могут потребоваться длительные вычисления для сходимости.	-0,172288	37,2712 с.
differential_evolution	Эффективен для задач с непрерывными параметрами и сложными ландшафтами целевых функций.	Может застрять в локальных минимумах при недостаточной диверсификации.	-0,172288	19,3581 с.
shgo	Гарантированная глобальная оптимизация для задач с ограничениями.	Высокие вычислительные затраты.	-0,379703	5,6986 с.
direct	Баланс между глобальным и локальным поиском.	Эффективность зависит от сложности целевой функции.	-0,172288	15,6280 с.

Метод оптимизации DIRECT обладает несколькими преимуществами перед другими методами оптимизации, такими как генетические алгоритмы, методы градиентного спуска или эволюционные стратегии. Некоторые из ключевых преимуществ DIRECT заключаются в следующем.

Отсутствие требования градиента. В отличие от методов градиентного спуска, которые требуют доступа к градиенту целевой функции, DIRECT не требует вычисления градиента. Это особенно полезно в случаях, когда функция может быть шумной, разрывной или недифференцируемой.

Глобальная оптимизация. DIRECT является методом глобальной оптимизации, что означает, что он стремится найти глобальный оптимум, а не застревать в локальных оптимумах, как это иногда происходит с методами градиентного спуска.

Эффективность в высокомерных пространствах. DIRECT может быть эффективным в пространствах с высокой размерностью, где методы, основанные на градиентном спуске, могут столкнуться с проблемой «проклятия размерности».

Простота настройки. DIRECT обычно имеет меньше гиперпараметров для настройки по сравнению с другими методами оптимизации, такими как генетические алгоритмы или эволюционные стратегии. Это делает его более простым в использовании и менее зависимым от предварительной настройки.

Эффективность в случае разреженных данных. В случае работы с разреженными данными, где функции могут быть несглаженными или иметь разры-

вы, DIRECT может быть предпочтительным выбором из-за его способности эффективно искать оптимум в таких условиях.

Алгоритм оценки результативности противодействия идентификации метастурктуры информационной системы на уровне перколяционных кластеров киберпространства

В основе оценки результативности противодействия идентификации метастурктуры информационной системы на уровне перколяционных кластеров киберпространства лежит предположение, что злоумышленник не знает топологии разведываемого объекта, поэтому использует метод случайного блуждания как стратегии разведки.

Стратегия сканирования сети по методу случайного блуждания является одной из техник, используемых для обнаружения узлов и ресурсов в сети. Этот метод отличается своей простотой и эффективностью в условиях, когда отсутствует информация о топологии сети или когда необходимо избежать обнаружения стандартных сканеров.

Основная идея метода случайного блуждания основывается на принципе случайного выбора следующего узла для сканирования. В отличие от систематического или последовательного подхода, здесь используется случайный выбор IP-адресов или узлов в сети, что затрудняет детектирование сканирования и делает процесс более скрытным.

Алгоритм случайного блуждания включает следующие шаги.

Начальная точка: начало сканирования происходит из одного или нескольких начальных узлов, которые могут быть выбраны случайным образом или заданы вручную.

Генерация случайных адресов: из текущего узла выбирается случайный IP-адрес или узел в сети для следующего шага. Этот выбор может осуществляться с использованием генераторов случайных чисел.

Переход к следующему узлу: происходит переход к выбранному узлу и выполнение определенных действий, таких как попытка подключения, сбор информации или обнаружение открытых портов.

Повторение процесса: процесс повторяется заданное количество раз или до достижения определенных условий завершения, например, сканирование всей сети или истечение времени.

Преимущества алгоритма случайного блуждания заключаются: в высокой бескомпроматности, поскольку выбор узлов происходит случайным образом, поведение сканера напоминает обычный сетевой трафик, что затрудняет его обнаружение и блокировку средствами сетевой безопасности; простоте реализации, поскольку он не требует сложных вычислительных ресурсов; гибкости, поскольку метод подходит для различных типов сетей, включая большие и плохо документированные сети.

Недостатки алгоритма случайного блуждания заключаются в относительно низкой эффективности, так как могут быть не только пропущены важные уз-

лы (участки) сети, но и без дополнительных механизмов возможно многократное посещение одних и тех же узлов.

Мерой результативности противодействия идентификации метастурктуры информационной системы на уровне перколяционных кластеров киберпространства выбрана вероятность обхода вершин исходного графа в маскированном графе (т.е. вскрытии исходной структуры КИС) методом случайного блуждания.

Псевдокод алгоритма оценки результативности противодействия идентификации метастурктуры информационной системы на уровне перколяционных кластеров киберпространства представлен на рис. 8. Пояснения по физическому смыслу используемых в алгоритме переменных даны далее по тексту.

Algorithm 3 Оценка результативности маскирования по методу случайного блуждания

```
1: Вход: Оригинальный граф G_orig, маскированный граф G_mask, количество попыток обхода num_attempts
2: Выход: Вероятность обхода вершин графа G_orig в графе G_mask за num_attempts попыток
3: orig_length ← |V(G_orig)|
4: prob ← 0 # Заполняем нулями
5: for attempt ∈ range(num_attempts) do
6:     start_node ← random_choice(V(G_mask))
7:     current_node ← start_node
8:     visited_nodes ← ∅
9:     while current_node ∈ V(G_orig) do
10:        visited_nodes ← visited_nodes ∪ {current_node}
11:        neighbours ← list(N_G(current_node, G_mask)) # Список соседей в маскированном графе
12:        if neighbours = ∅ then
13:            Break
14:        next_node ← random_choice(neighbours)
15:        if next_node ∉ visited_nodes then
16:            current_node ← next_node
17:        else
18:            Break
19:    prob[attempt] ← |visited_nodes|/orig_length

20: return E(prob)
```

Рис. 8. Псевдокод алгоритма оценки результативности противодействия идентификации метаструктур КИС методом случайного блуждания

Анализ временной сложности алгоритма оценки результативности противодействия идентификации метаструктур КИС методом случайного блуждания

Оценка в О-нотации верхней асимптотической сложности алгоритма, представленного на рис. 8, представлена ниже.

1. Строка 3: Инициализация переменной $orig\_length$, что выполняется за $O(1)$, поскольку это простая операция определения размера множества вершин графа G_{orig}.

2. Строка 4: Инициализация массива $prob$, который хранит результаты для каждой попытки. Это занимает $O(num\_attempts)$, поскольку массив заполняется нулями для каждого обхода.

3. Цикл по попыткам (строки 5–20): Основной цикл выполняется $num\_attempts$ раз. Внутри этого цикла:

Строка 6: Выбор случайной вершины в графе G_{mask} происходит за $O(1)$, если граф представлен списком вершин.

Строка 8: Инициализация пустого множества $visited\_nodes$ занимает $O(1)$.

Цикл *while* (строки 9–19): Этот цикл продолжается до тех пор, пока текущая вершина принадлежит $V(G_{orig})$. В худшем случае этот цикл может выполняться за $O(|V(G_{orig})|)$, так как в каждой итерации добавляется одна уникальная вершина в множество посещённых узлов.

Внутри цикла:

Строка 11: Добавление вершины в множество $visited\_nodes$ занимает $O(1)$.

Строка 12: Получение списка соседей текущей вершины в графе G_{mask} выполняется за $O(\deg(v))$, где $\deg(v)$ – степень вершины v. В худшем случае эта операция имеет сложность $O(|V(G_{mask})|)$.

Строка 14: Проверка, есть ли соседи у вершины, выполняется за $O(1)$.

Строка 16: Выбор следующей вершины среди соседей выполняется за $O(1)$, если список соседей уже сформирован.

Строка 17: Проверка, посещалась ли вершина ранее, занимает $O(1)$, если используем эффективную структуру данных, такую как хеш-таблица.

Общая временная сложность одного выполнения цикла *while* зависит от максимальной степени вершин и длины обхода, что можно выразить как $O(|V(G_{orig})|\cdot|V(G_{mask})|)$ в худшем случае (при каждом шаге обходятся все соседи).

4. Строка 20: Обновление массива вероятностей после каждой попытки обхода выполняется за $O(1)$.

5. Строка 23: Вычисление среднего значения вероятностей по всем попыткам обхода занимает $O(num\_attempts)$, так как нужно просуммировать все значения в массиве $prob$.

Основной вклад во временную сложность даёт цикл по количеству попыток обхода $num\_attempts$, где на каждой итерации цикл *while* может выполняться за $O(|V(G_{orig})|\cdot|V(G_{mask})|)$ в худшем случае. Таким образом, общая временная сложность алгоритма составляет:

$$O(num\_attempts\cdot|V(G_{orig})|\cdot|V(G_{mask})|),$$

где:

$|V(G_{orig})|$ – количество вершин в оригинальном графе,

$\left|V(G_{mask})\right|$ – количество вершин в маскированном графе,

$num\_attempts$ – количество попыток обхода.

Методика противодействия идентификации метаструктуры информационной системы на уровне перколяционных кластеров киберпространства

Методика противодействия идентификации метаструктуры информационной системы на уровне перколяционных кластеров киберпространства состоит из следующих шагов.

1. Определение метрик маскирования: в качестве основных метрик для оценки эффективности маскирования используются перколяционная центральность вершин исходного графа, количество добавленных узлов, размер гигантской связной компоненты добавленных вершин и коэффициент сходства между исходным и маскированным графами (коэффициент Жаккарда).

2. Определение целевой функции: целью методики является повышение перколяционной адаптивности графа, то есть уменьшение перколяционной центральности вершин для каждой проекции, снижение коэффициента сходства между исходным и маскированными графами, при этом повышая размер гигантской связной компоненты путем добавления минимального количества вершин.

3. Выбор весовых коэффициентов целевой функции.

4. Построение графов: для исходной информационной системы с использованием одномодовых проекций для каждого протокола прикладного уровня в киберпространстве создаётся набор графов для маскирования.

5. Определение алгоритма маскирования: Реализуется алгоритм маскирования на основе модифицированного алгоритма Гирван-Ньюмена, который удаляет минимальные мостики между проекциями с учетом перколяционной адаптивности.

6. Маскирование графа: алгоритм маскирования применяется к графам, полученным на шаге 4, и проводится оптимизация маскирования с использованием целевой функции (1) и ограничений (4).

7. Оценка эффективности маскирования: проводятся статистические анализы по метрикам перколяционной адаптивности, количеству добавленных узлов и размеру гигантской связной компоненты для сравнения исходного графа и маскированного графа.

8. Итерации: в случае недостаточной эффективности маскирования процесс повторяется, достигнув оптимального результата по всем метрикам.

9. Оценка результативности противодействия идентификации метастурктур информационной системы: проводятся тесты на обнаружение и определение происхождения узлов в маскированном графе с использованием метода случайного блуждания.

Таким образом, методика противодействия идентификации метаструктуры информационной системы на уровне перколяционных кластеров киберпространства позволяет обеспечить высокую степень анонимности и безопасности

данных путем оптимизации перколяционной адаптивности, минимизации коэффициента сходства между исходным и маскированными графами и увеличения размера гигантской связной компоненты.

Методика противодействия идентификации метаструктур информационных систем на уровне перколяционных кластеров киберпространства имеет несколько следующих отличий от уже известных подходов ([48, 49]).

1. Использование графовой структуры для представления сетевой топологии и применение маскирования на уровне перколяционных кластеров. Такой подход позволяет создавать более сложные и эффективные механизмы противодействия идентификации метаструктур, увеличивая сложность анализа и обнаружения потенциальных угроз.

2. Использование перколяционных центральностей. Методика использует оценку перколяционных центральностей для каждого узла в графе, что позволяет учитывать влияние добавленных узлов на структуру графа.

3. Многокритериальная оптимизация. Целевая функция маскирования является суммой метрик перколяционной адаптивности графа, что позволяет учитывать несколько критериев оптимальности.

4. Адаптивность к изменению информационной системы. Методика ориентирована на адаптацию метаструктур информационных систем к изменениям в информации и структурах графа.

Результаты применения методики противодействия идентификации метаструктуры информационной системы на уровне перколяционных кластеров киберпространства

Рассмотрим результаты применения методики для КИС, структура которой представлена на рис. 4 и рис. 5.

Условия применения методики предусматривали несколько ситуаций, определяемых весовыми коэффициентами $\left(w_{\text{added\_nodes}}, w_{\text{max\_centrality}}, w_{\text{giant}}, w_{\text{jaccard}}\right)$ целевой функции:

Таблица 2 – Условия применения методики

Ситуация	Набор значений коэффициентов	Описание ситуации
*S*1	(0,25; 0,25; 0,25; 0,25)	Равнозначность критериев оптимальности противодействия идентификации метаструктур КИС
*S*2	(0,4; 0,2; 0,2; 0,2)	Приоритет критерия количества добавленных вершин маскированного графа
*S*3	(0,2; 0,4; 0,2; 0,2)	Приоритет критерия маскимальной перколяционной центральности вершин маскированного графа
*S*4	(0,2; 0,2; 0,4; 0,2)	Приоритет критерия размера гигантской связной компоненты добавленных вершин маскированного графа
*S*5	(0,2; 0,2; 0,2; 0,4)	Приоритет критерия коэффициента сходства Жаккарда исходного и маскированного графов

№4. 2024
ISSN 2410-9916

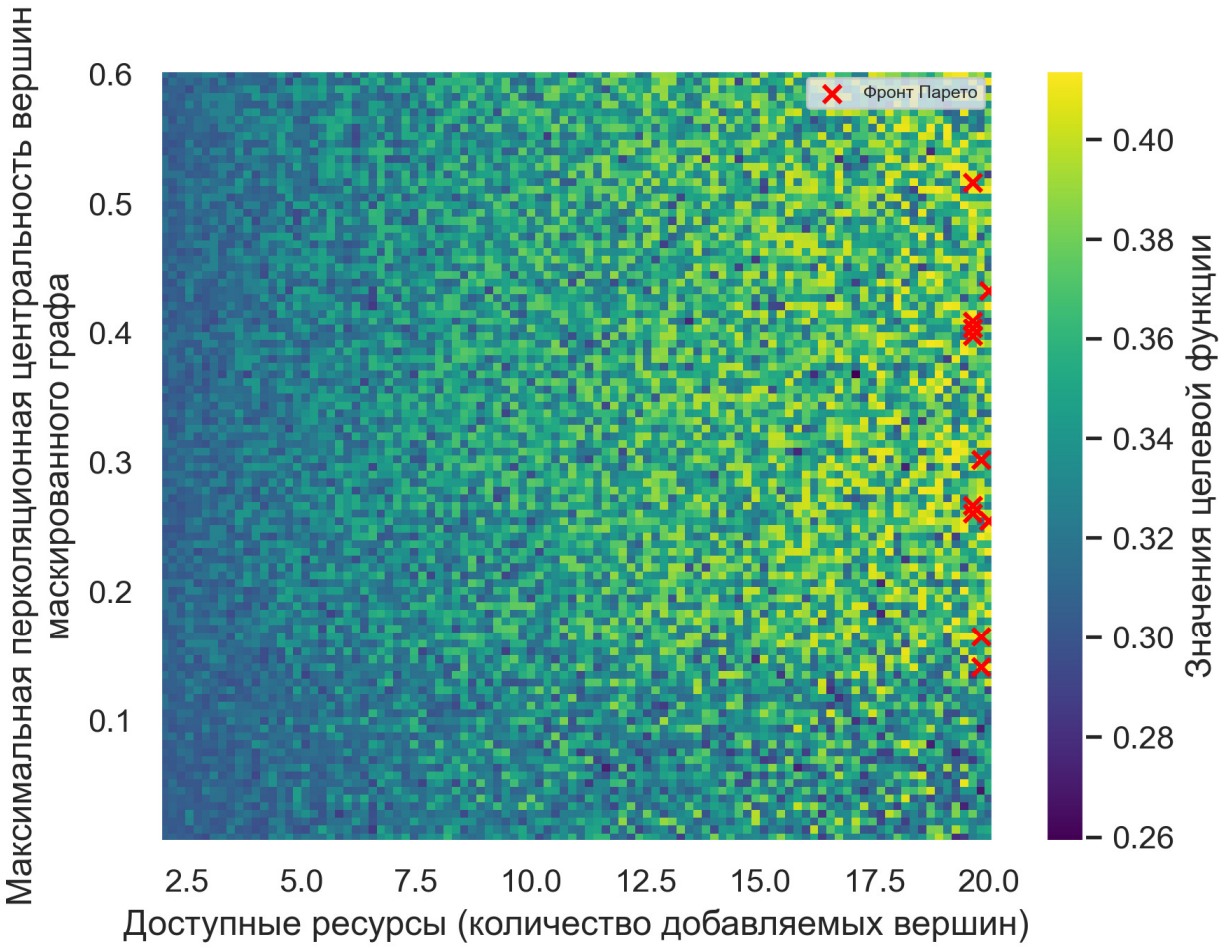

Рис. 9. Зависимость целевой функции от параметров маскирования

Результирующие маскированные графы и их гистограммы показателей относительных частот перколяционных центральностей для графа протокола *DNS* (рис. 10-11) исходной КИС представлены на рис. 12-21 (красные вершины – исходный граф, зелёные вершины – добавленные).

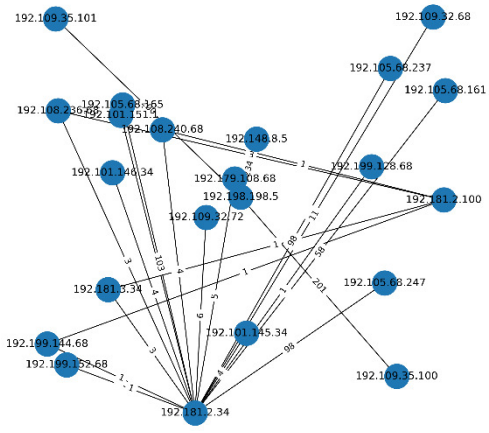

Рис. 10. Граф протокола DNS исходной сети КИС

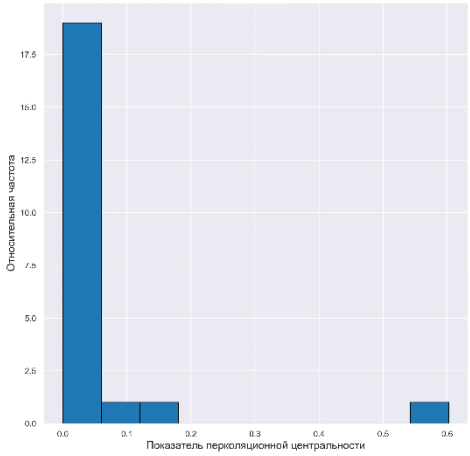

Рис. 11. Гистограмма показателей перколяционной центральности вершин графа для протокола *DNS*

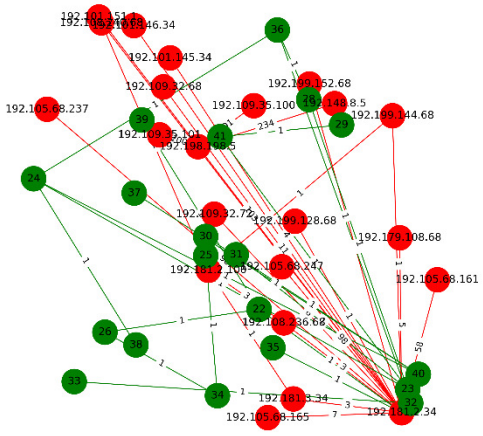

Рис. 12. Маскированный граф протокола DNS для ситуации *S*1

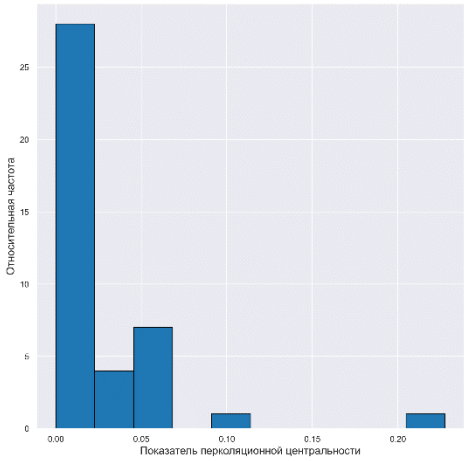

Рис. 13. Гистограмма показателей перколяционной центральности вершин маскированного графа для протокола DNS для ситуации *S*1

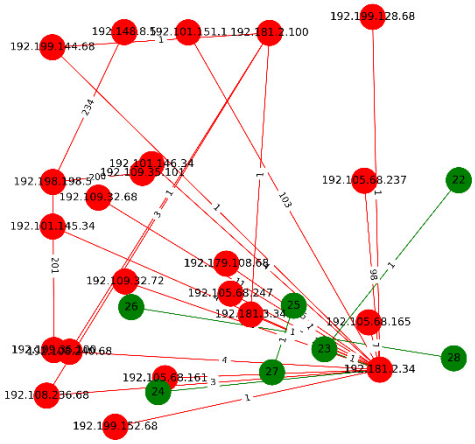

Рис. 14. Маскированный граф протокола DNS для ситуации *S*2

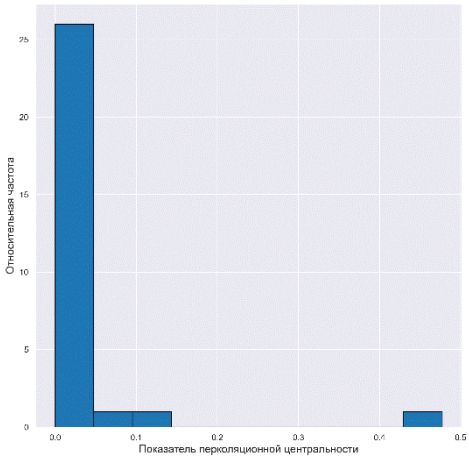

Рис. 15. Гистограмма показателей перколяционной центральности вершин маскированного графа для протокола DNS для ситуации *S*2

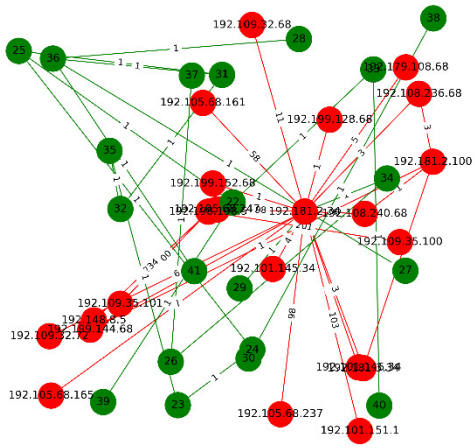

Рис. 16. Маскированный граф протокола DNS для ситуации *S*3

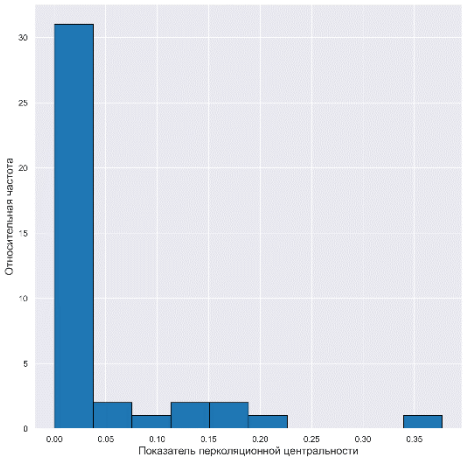

Рис. 17. Гистограмма показателей перколяционной центральности вершин маскированного графа для протокола DNS для ситуации *S*3

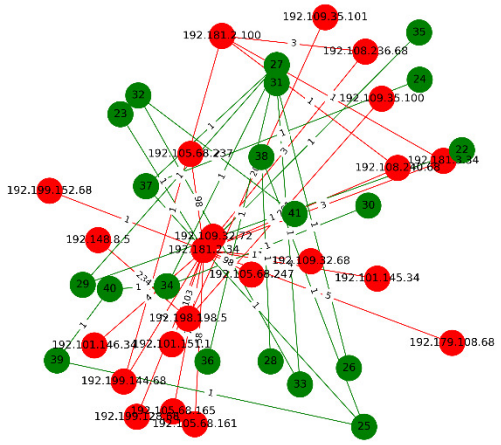

Рис. 18. Маскированный граф протокола DNS для ситуации *S*4

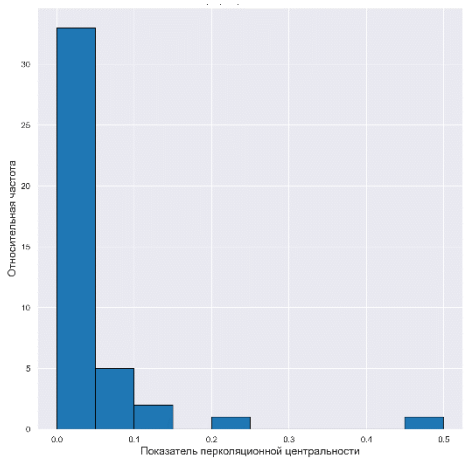

Рис. 19. Гистограмма показателей перколяционной центральности вершин маскированного графа для протокола DNS для ситуации *S*4

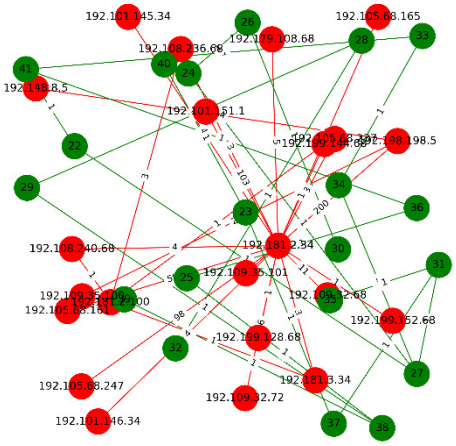

Рис. 20. Маскированный граф проекции протокола DNS для ситуации *S*5

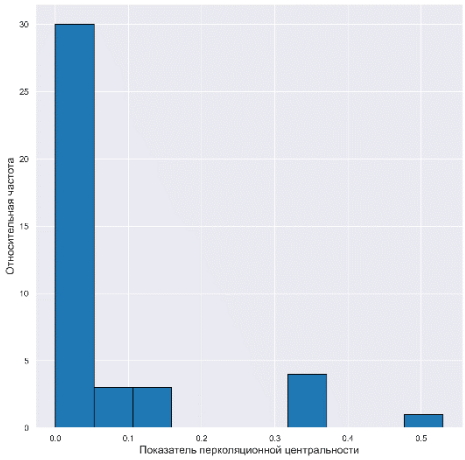

Рис. 21. Гистограмма показателей перколяционной центральности вершин маскированного графа для протокола DNS для ситуации *S*5

Зависимость частных критериев и расстояние от оптимальных значений параметров до идеальной точки при различных значениях важности в критериальном пространстве представлена на рис. 22-25.

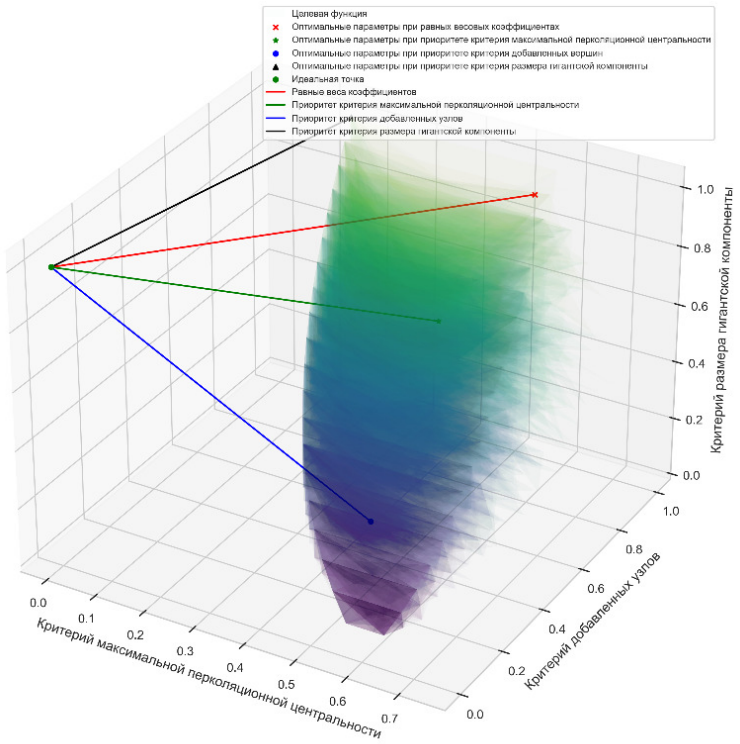

Рис. 22. Достижимое множество пространства частных критериев перколяционной центральности, количества добавленных узлов и размера гигантской компоненты

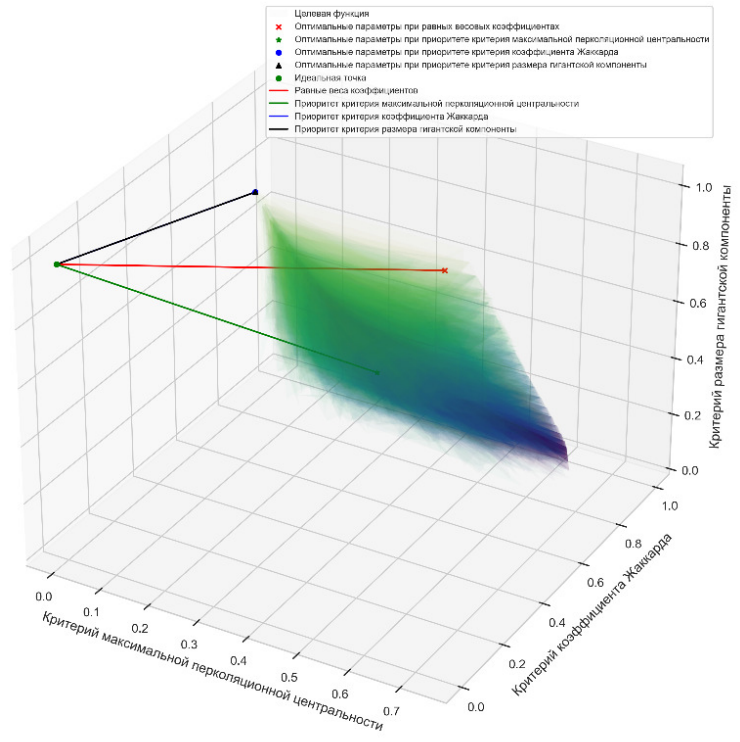

Рис. 23. Достижимое множество пространства частных критериев перколяционной центральности, коэффициента Жаккарда и размера гигантской компоненты

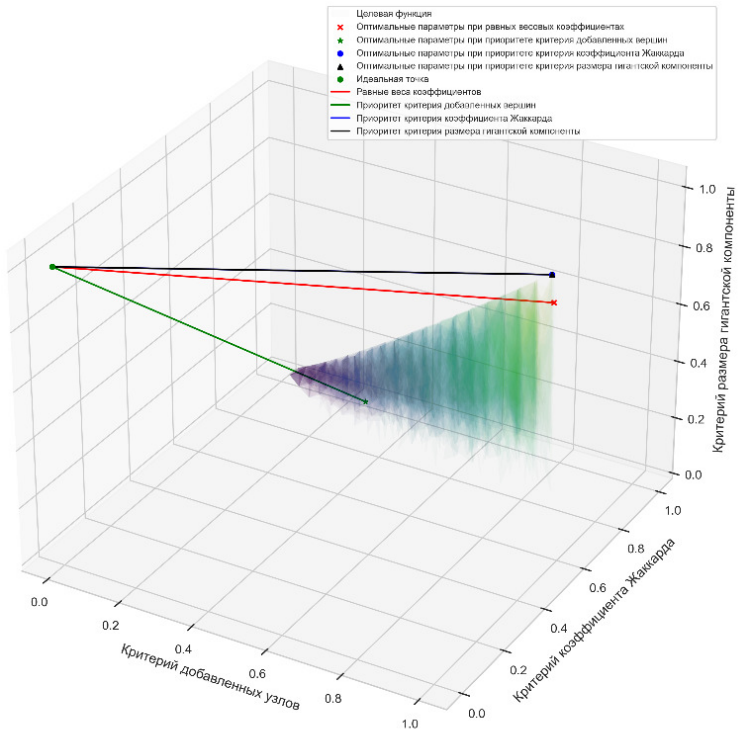

Рис. 24. Достижимое множество пространства частных критериев количества добавленных узлов, коэффициента Жаккарда и размера гигантской компоненты

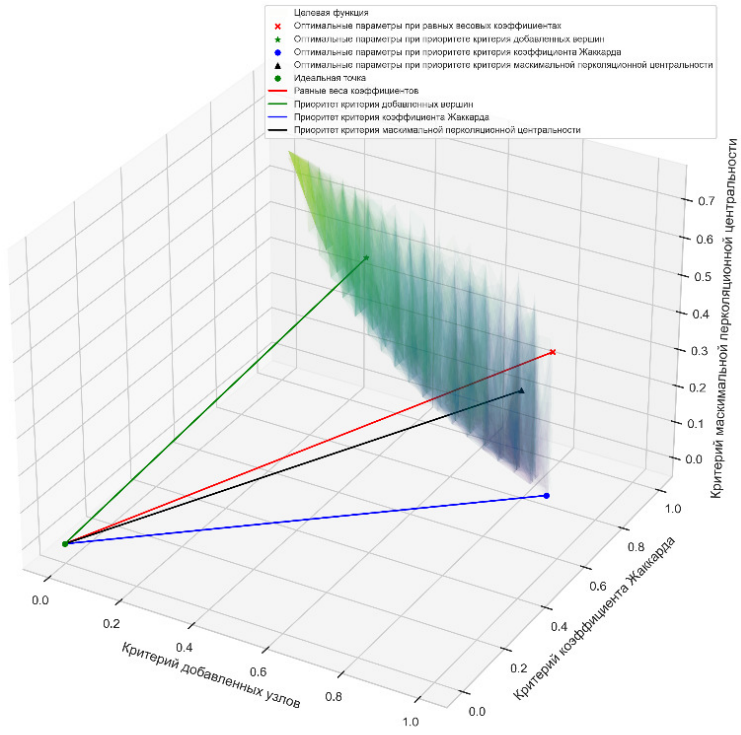

Рис. 25. Достижимое множество пространства частных критериев перколяционной центральности, коэффициента Жаккарда и количества добавленных узлов

№4. 2024
ISSN 2410-9916

Оценка результативности маскирования методом случайных блужданий производилась для набора графов с 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 вершинами. Вероятность обхода вершин оригинального графа в маскированном за 100 попыток для различных ситуаций *S*1-*S*5 коэффициентов важности представлена на рис. 26.

Рис. 26. Вероятность обхода вершин оригинального графа в маскированном в зависимости от конфигурации графа

В итоге, после применения описанной методики, исходный двудольный граф (рис. 4) преобразовался в двудольный граф маскированной КИС (рис. 27).

DOI: 10.24411/2410-9916-2024-4-179-222
URL: https://sccs.intelgr.com/archive/2024-04/07-Maximov.pdf

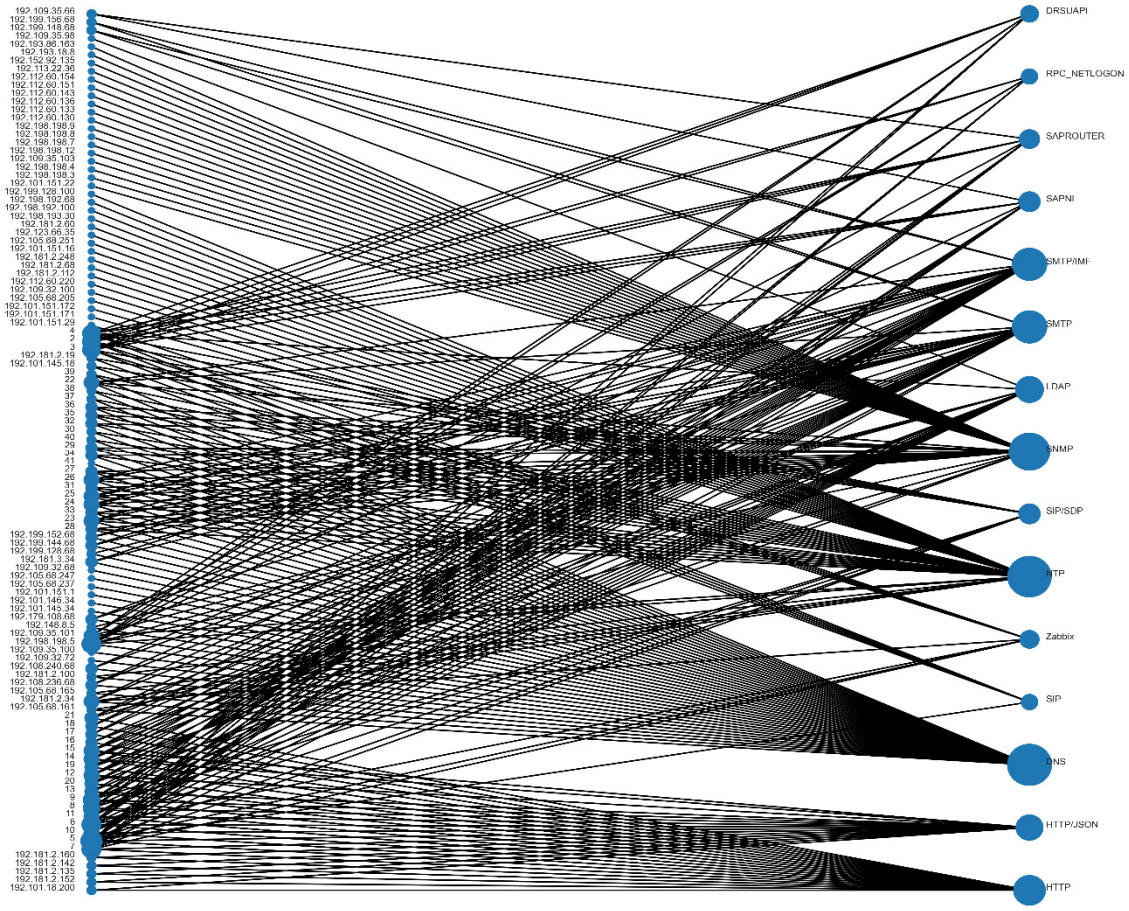

Рис. 27. Двудольный граф маскированной КИС

Выводы

Предложенная методика противодействия идентификации метаструктур корпоративных информационных систем на уровне перколяционных кластеров киберпространства, отличающаяся от известных ([48], в которой используется модификация алгоритмов минимальных остовных деревьев, их применение для решения задачи синтеза структуры сети маскирующего обмена и реализация маскирующего обмена при дифференциации маршрутов между узлами связи по критерию безопасности транзитных узлов связи, [49], в которой обеспечивают повышение скрытности связи и затруднение идентификации абонентов сети несанкционированными абонентами за счет непрерывного изменения идентификаторов абонентов сети в передаваемых пакетах сообщений, передачу пакетов сообщений по всем допустимым маршрутам связи и передачу маскирующих сообщений по маскирующим маршрутам связи) использованием темпоральной графовой структуры для представления сетевой топологии и применением маскирования на уровне физической сети, оценкой перколяционных центральностей для каждого узла в графе, что позволяет учитывать влияние добавленных узлов на структуру графа, целевой функцией маскирования как суммы метрик перколяционной адаптивности графа, позволяет учитывать несколько критериев оптимальности и создавать более сложные и эффективные механизмы про-

тиводействия идентификации метаструктур КИС в киберпространстве, увеличивая сложность анализа и обнаружения потенциальных угроз противником.

Литература

1. Blowers M. Evolution of Cyber Technologies and Operations to 2035 – Springer International Publishing, 2015. – 194 p. doi: 10.1007/978-3-319-23585-1.

2. Стародубцев Ю. И., Закалкин П. В, Иванов С. А. Техносферная война как основной способ разрешения конфликтов в условиях глобализации // Военная Мысль. 2020. № 10. С. 16-21.

3. ГОСТ IEC 60050-732-2017. Международный электротехнический словарь. Часть 732. Технологии компьютерных сетей. – М.: Стандартинформ. – 2020. – 41 с.

4. Joint Chiefs of Staff. Cyberspace operations. Joint Chiefs of Staff (US); 19 2022 Dec 19. Joint Publication No.: JP 3-12. // Official Website of the Joint Chiefs of Staff [Электронный ресурс]. 26.08.2023. – URL: https://www.jcs.mil/Doctrine/Joint-Doctrine-Pubs/3-0-Operations-Series/ (дата обращения: 26.08.2023).

5. Величко В. М., Сидоренко Е. Н., Суюндукова А. А. Техническая компьютерная разведка – основная угроза сети современности // Наука и образование: проблемы и стратегия развития. 2017. Т. 2. № 1(3). С. 18-20.

6. Макаренко С. И. Аудит безопасности критической инфраструктуры специальными информационными воздействиями. Монография. – СПб.: Наукоемкие технологии, 2018. – 122 с. – URL: http://sccs.intelgr.com/editors/Makarenko/makarenko-audit_ib_2018.pdf (дата обращения: 11.11.2024)

7. Кибербезопасность цифровой индустрии. Теория и практика функциональной устойчивости к кибератакам / Под редакцией Д. П. Зегжды – М.: Горячая линия–Телеком, 2022. – 560 с.

8. Теленьга А. П. Маскирование метаструктур информационных систем в киберпространстве // Вопросы кибербезопасности. 2023. № 5(57). С. 50-59. doi: 10.21681/4311-3456-2023-5-50-59.

9. Макаренко С. И. Информационный конфликт системы связи с системой дестабилизирующих воздействий. Часть III: Управление системой связи в условиях конфликта // Техника радиосвязи. 2021. № 1(48). С. 103-116. doi: 10.33286/2075-8693-2021-48-103-116.

10. Rawski M., Jalowski Ł., Zmuda M. A survey on moving target defense for networks: A practical view // Electronics. 2022. Vol. 11. № 18. P. 2886 doi: 10.3390/electronics11182886.

11. Cai G., Wang B., Hu W., Wang T. Moving target defense: state of the art and characteristics // Frontiers of Information Technology & Electronic Engineering. 2016. Vol. 17. № 5. P. 1122-1153. doi: 10.1631/FITEE.1601321.

12. Hong J. B., Kim D. S. Assessing the effectiveness of moving target defenses using security models // IEEE Transactions on Dependable and Secure Computing. 2015. Vol. 12. № 1. P. 10–22. doi: 10.1109/TDSC.2014.2316819.

13. Cho J. H., Sharma D. P., Alavizadeh H. Toward proactive, adaptive defense: A survey on moving target defense // IEEE Communications Surveys & Tutorials. 2020. Vol. 22. № 1. P. 709–745. doi: 10.1109/COMST.2019.2945097.

14. Кравцов К. Н. Передача данных в сетях с динамической рандомизацией адресного пространства // Труды XVII Международной конференции DAMDro/RCDL. – 2015. – С. 273-277.

15. Андриенко А. А., Кожевников Д. А., Колбасова Г. С. и др. Способ (варианты) и устройство (варианты) защиты канала связи вычислительной сети // Патент на изобретение RU 2306599 C1, опубл. 20.09.2007, бюл. № 11.

16. Кожевников Д. А., Максимов Р. В., Павловский А. В. Способ защиты вычислительной сети (варианты) // Патент на изобретение RU 2325694 C1, опубл. 27.05.2008, бюл. № 15.

17. Krylov V., Kravtsov K., Sokolova E. Fast IP hopping protocol SDI implementation // Indian Journal of Science and Technology. 2015. Vol. 8. № 36. P. 90557. doi: 10.17485/ijst/2015/v8i36/90557.

18. Привалов А. А., Скуднева Е. В. Подход к оценке маскирования информационного обмена в сетях передачи данных оперативно технологического назначения при целевых атаках // Известия Петербургского университета путей сообщения. 2017. № 3. С. 452-460. – URL: https://cyberleninka.ru/article/n/podhod-k-otsenke-maskirovaniya-informatsionnogo-obmena-v-setyah-peredachi-dannyh-operativno-tehnologicheskogo-naznacheniya-pri (дата обращения: 11.11.2024).

19. Казарин О. В. Методы и средства проактивной защиты программного обеспечения информационных систем специального назначения: Автореф. дисс. … док. техн. наук – М., 2012. – 39 с.

20. Лыков Н. Ю. Методика управления ресурсами маскираторов информационных направлений распределенных интегрированных инфокоммуникационных систем ведомственного назначения // Инженерный вестник Дона. 2018. № 4 (51). С. 134.

21. Шерстобитов, Р. С. Модель маскирования информационного обмена в сети передачи данных ведомственного назначения // Системы управления, связи и безопасности. 2024. № 1. С. 1-25. doi: 10.24412/2410-9916-2024-1-001-025.

22. Лазарев А. А., Калач А. В., Пысин С. А. Методика оптимизации функционально-эквивалентной структуры вычислительных сетей специального назначения // Вестник Воронежского института ФСИН России. 2009. № 1. С. 89-94.

23. 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.

24. Maximov R. V., Ivanov I I., Sharifullin S. P. Network Topology Masking in Distributed Information Systems // CEUR Workshop Proceedings. – 2017. – Т. 2081. – С. 83-87.

25. 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.

26. Лебедкина Т. В., Соколовский С П. Модель функционирования защищенной технологии файлового обмена // Вопросы кибербезопасности. 2021. № 5 (45). С. 52-62. doi: 10.21681/2311-3456-2021-5-52-62.

27. Соколовский С. П. Параметрическая оптимизация информационных систем при решении задачи проактивной защиты сервиса передачи данных от сетевой разведки // Вестник компьютерных и информационных технологий. 2022. Т. 19. № 5 (215). С. 49-57. doi: 10.14489/vkit.2022.05.pp.049-057.

28. Кучуров В. В., Максимов Р. В., Шерстобитов Р. С. Модель и методика маскирования адресации корреспондентов в киберпространстве // Вопросы кибербезопасности. 2020. № 6 (40). С. 2-13. doi: 10.21681/2311-3456-2020-06-2-13.

29. Соколовский С. П. Комплекс проактивной защиты информационных систем от сетевой разведки // Вестник компьютерных и информационных технологий. 2021. Т. 18. № 11 (209). С. 53-62. doi: 10.14489/vkit.2021.11.pp.053-062.

30. Горбачев А. А. Модель и параметрическая оптимизация проактивной защиты сервиса электронной почты от сетевой разведки // Вопросы кибербезопасности. 2022. № 3 (49). С. 69-81. doi: 10.21681/4311-3456-2022-3-69-81.

31. Москвин А. А., Максимов Р. В., Горбачев А. А. Модель, оптимизация и оценка эффективности применения многоадресных сетевых соединений в условиях сетевой разведки // Вопросы кибербезопасности. 2023. № 3(55). С. 13-22. doi: 10.21681/2311-3456-2023-3-13-22.

32. Горбачев А. А., Соколовский С. П., Каплин М. А. Определение оптимальных параметров конфигурирования информационных систем в условиях сетевой разведки // Вопросы кибербезопасности. 2022. № 4 (50). С. 80-90. doi: 10.21681/2311-3456-2022-4-80-90.

33. Соколовский С. П., Теленьга А. П. Методика формирования ложного сетевого трафика информационных систем для защиты от сетевой разведки // Вестник компьютерных и информационных технологий. 2022. Т. 19. № 2 (212). С. 40-47. doi: 10.1489/vkit.2022.02.pp.040-047.

34. Федер Е. Фракталы: Пер. с англ. 2-е изд. – М.: УРСС, 2014. – 264 с.

35. Grimmett G. Percolation. – Cambridge: Springer, 1999. – 444 p.

36. Шуваев Ф. Л., Татарка М. В. Анализ динамики мер центральности математических моделей случайных графов // Научно-технический вестник информационных технологий, механики и оптики. 2020. Т. 20. № 2. С. 249-256. doi: 10.17586/2226-1494-2020-20-2-249-256.

37. Piraveenan M., Prokopenko M., Hossain L. Percolation Centrality: Quantifying Graph-Theoretic Impact of Nodes during Percolation in Networks // PLoS ONE. 2013. Vol. 8. № 1. P. 53095. doi: 10.1371/journal.pone.0053095.

38. Берест П. А., Богачев К. Г., Выговский Л. С. и др. Способ сравнительной оценки структур информационно-вычислительной сети // Патент на изобретение RU 2408928 C1, опубл. 10.01.2011, бюл. № 1.

39. Игнатенко А. В., Ковалевский С. Г., Максимов Р. В. и др. Способ сравнительной оценки структур сетей связи // Патент на изобретение RU 2450338 C1, опубл. 10.05.2012, бюл. № 13.

40. Апарин Н. Н., Астахов А. И., Жираковский А. А. и др. Способ сравнительной оценки структур сетей связи // Патент на изобретение RU 2460123 C1, опубл. 27.08.2012, бюл. № 24.

41. Искольный Б. Б., Лазарев А. А., Лыков Н. Ю. и др. Способ сравнительной оценки структур сети связи // Патент на изобретение RU 2626099 C1, опубл. 21.07.2017, бюл. № 21.

42. Tsallis C., Stariolo D. A. Generalized Simulated Annealing // Physica A. 1996. Vol. 233. P. 395–406.

43. Xiang Y., Sun D. Y., Fan W., Gong X. G. Generalized Simulated Annealing Algorithm and Its Application to the Thomson Model // Physics Letters A. 1997. Vol. 233. P. 216–220.

44. Storn R., Price K. Differential Evolution - a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces // Journal of Global Optimization. 1997. Vol. 11. P. 341–359.

45. Endres S., Sandrock C., Focke W. A simplicial homology algorithm for Lipschitz optimisation // Journal of Global Optimization. 2018. Vol. 72. P. 181-217. doi: 10.1007/s10898-018-0645-y.

46. Gablonsky J., Kelley C. A Locally-Biased form of the DIRECT Algorithm // Journal of Global Optimization. 2001. Vol. 21. P. 27–37.

47. Jones D. R., Perttunen C. D., Stuckman B. E. Lipschitzian optimization without the Lipschitz constant // J. Optim. Theory Appl. 1993. Vol. 79. P. 157–181.

48. 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.

49. Голуб Б. В., Краснов В. А., Лыков Н. Ю., Максимов Р. В. Способ маскирования структуры сети связи // Патент на изобретение RU 2645292, опубл. 19.02.2018, бюл. № 5.

50. Styugin M., Patokin N. Multilevel Decentralized Protection Scheme Based on Moving Targets // International Journal of Security and Its Applications. 2016. Vol. 10. № 1. P. 45-54. doi: 10.14257/ijsia.2016.10.1.05.

51. Стюгин М. Защита интернет-ресурсов по технологии движущейся цели // Доклады ТУСУР. 2015. № 2 (36). С. 80-85 – URL: https://cyberleninka.ru/article/n/zaschita-internet-resursov-po-tehnologii-dvizhuscheysya-tseli (дата обращения: 02.03.2024).

References

1. Blowers M. *Evolution of Cyber Technologies and Operations to 2035*, Springer International Publishing, 2015. 194 p. doi: 10.1007/978-3-319-23585-1.

2. Starodubtsev Yu. I., Zakalkin P. V., Ivanov S.A. Technosphere Warfare as the Chief Method of Conflict Settlement in Conditions of Globalization. *Military Thought*, 2020, no. 10, pp. 16-21 (in Russian).

3. State Standard IEC 60050-732-2017. International Electrotechnical Vocabulary. Part 732. Computer Network Technology. Moscow, Standartov Publ., 2020. 41 p. (in Russian).

4. Joint Chiefs of Staff. Cyberspace Operations. Joint Chiefs of Staff (US). Joint Publication No. JP 3-12. *Official Website of the Joint Chiefs of Staff*. Available at: https://www.jcs.mil/Doctrine/Joint-Doctrine-Pubs/3-0-Operations-Series (accessed: 26.08.2023).

5. Velichko V. M. Tekhnicheskaya Kompyuternaya Razvedka – Osnovnaya Ugroza Seti Sovremennosti [Technical Computer Rreconnaissance is the major network threat of our time]. *Nauka i obrazovanie: problemi i strategiya razvitiya* [Science and education: problems and development strategy], 2017, vol. 2, no. 1(3), pp. 18-20 (in Russian).

6. Makarenko S. I. *Audit Bezopasnosti Kriticheskoi Infrastruktury Spetsialnymi Informatsionnymi Vozdeistviiami* [Security Audit of Critical Infrastructure with Special Information Impacts]. St. Petersburg, Naukoemkie Tekhnologii Publ., 2018. 122 p. Available at: http://sccs.intelgr.com/editors/Makarenko/makarenko-audit_ib_2018.pdf (accessed 11 November 2024) (in Russian).

7. *Kiberbezopasnost Tsifrovoi Industrii. Teoriya i Praktika Funktsionalnoi Ustoichivosti k Kiberatakam* [Cybersecurity of the Digital Industry. Theory and Practice of Functional Resilience to Cyberattacks]. Moscow, Goryachaya liniya. Telekom Publ., 2022. 560 p. (in Russian).

8. Telenga A. P. Masking Metastructures of Information Systems in Cyberspace. *Voprosi Kiberbezopasnosti*, 2023, no. 5(57), pp. 50-59 (in Russian).

9. Makarenko S. I. Informatsionnii Konflikt Sistemi Svyazi s Sistemoi Destabiliziruyushchikh Vozdeistvii. Chast III: Upravlenie Sistemoi Svyazi v Usloviyakh Konflikta [Information Conflict of a Communication System with a System of Destabilizing Influences. Part III: Managing the Communication System in Conflict]. *Tekhnika Radiosvyazi*, 2021, no. 1(48), pp. 103-116. doi: 10.33286/2075-8693-2021-48-103-116 (in Russian).

10. Rawski M., Jalowski Ł., Zmuda M. A Survey on Moving Target Defense for Networks: A Practical View. *Electronics*, 2022, vol. 11, no. 1, pp. 2886. doi: 10.3390/electronics11182886.

11. Cai G., Wang B., Hu W., Wang T. Moving Target Defense: State of the Art and Characteristics. *Frontiers of Information Technology & Electronic Engineering*, 2016, vol. 17, no. 5, pp. 1122-1153. doi: 10.1631/FITEE.1601321.

12. Hong J. B., Kim D. S. Assessing the Effectiveness of Moving Target Defenses Using Security Models. *IEEE Transactions on Dependable and Secure Computing*, 2015, vol. 12, no. 1, pp. 10–22. doi: 10.1109/TDSC.2014.2316819.

13. Cho J. H., Sharma D. P., Alavizadeh H. Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense. *IEEE Communications Surveys & Tutorials*, 2020, vol. 22, no. 1, pp. 709–745. doi: 10.1109/COMST.2019.2945097.

14. Kravtsov K. N. Peredacha Dannikh v Setyakh s Dinamicheskoi Randomizatsiei Adresnogo Prostranstva [Data Transmission in Networks with Dynamic Address Space Randomization]. *Selected Papers of the XVII International Conference on Data Analytics and Management in Data Intensive Domains (DAMID/RCDL 2015).* Obninsk, Russia, 2016, pp. 273-277 (in Russian).

15. Andrienko A. A., Kozhevnikov D. A., Kolbasova G. S. *Sposob (Varianti) i Ustroistvo (Varianti) Zashchiti Kanala Svyazi Vichislitelnoi Seti* [Method (Variants)

and Device (Variants) for Protecting the Communication Channel of a Computer Network] Patent Russia, no. 2306599 C1, 2006.

16. Kozhevnikov D. A, Maximov R. V., Pavlovskii A. V. *Sposob Zashchiti Vichislitelnoi Seti (Varianti)* [Method of Protection of a Computer Network (Variants)]. Patent Russia, no. 2325694 C1, 2006.

17. Krylov V. Fast IP Hopping Protocol SDI Implementation. *Indian Journal of Science and Technology*, 2015, vol. 8, no. 36, p. 90557. doi: 10.17485/ijst/2015/v8i36/90557.

18. Privalov A. A., Skudneva Ye. V. Information Traffic Masking Approach in Data Communication Networks of Operationally Technological Purpose During Targeted Attacks. *Izvestiya Peterburgskogo universiteta putei soobshcheniya*, 2017, no. 3, pp. 452-460. Available at: https://cyberleninka.ru/article/n/podhod-k-otsenke-maskirovaniya-informatsionnogo-obmena-v-setyah-peredachi-dannyh-operativno-tehnologicheskogo-naznacheniya-pri (accessed 11 November 2024) (in Russian).

19. Kazarin O. V. *Metodi i Sredstva Proaktivnoi Zashchiti Programmnogo Obespecheniya Informatsionnikh Sistem Spetsialnogo Naznacheniya*. Dis. dokt. tekh. nauk [Methods and Means of Proactive Software Protection of Special-Purpose Information Systems. Extended Abstract of Dr. habil. Thesis]. Moscow. 2012. 39 p. (in Russian).

20. Likov N. Yu. Metodika Upravleniya Resursami Maskiratorov Informatsionnikh Napravlenii Raspredelennikh Integrirovannikh Infokommunikatsionnikh Sistem Vedomstvennogo Naznacheniya [Methodology of Management of Resources of Maskers of Information Directions of Distributed Integrated Info-Communication Systems of Departmental Purpose]. *Inzhenernii vestnik Dona*, 2018, no. 4(51), pp. 134 (in Russian).

21. Sherstobitov R. S. The Model of Information Exchange Masking in the Departmental Communication Network. *Systems of Control, Communication and Security*, 2024, no. 1, pp. 1-25. doi: 10.24412/2410-9916-2024-1-001-025 (in Russian).

22. Lazarev A. A. Metodika Optimizatsii Funktsionalno-Ekvivalentnoi Strukturi Vichislitelnikh Setei Spetsialnogo Naznacheniya [Methodology of Optimization of Functional-equivalent Structure of Special-purpose Computer Networks]. *Vestnik of Voronezh Institute of the Russian Federal Penitentiary Service*, 2009, no. 1, pp. 89-94 (in Russian).

23. Voronchikhin I. S. Masking of Distributed Information Systems Structure in Cyber Space. *Voprosi Kiberbezopasnosti*, 2019, no. 6(34), pp. 92-101. doi: 10.21681/2311-3456-2019-6-92-101 (in Russian).

24. Maximov R. V., Ivanov I. I., Sharifullin S. R. Network Topology Masking in Distributed Information Systems. *CEUR Workshop Proceedings*. Moscow, Russia, 2017, pp. 83-87 (in Russian).

25. 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).

26. Lebedkina T. V. Model of Secure File Exchange Information Technology Operation. *Voprosi Kiberbezopasnosti*, 2021, no. 5(45), pp. 52-62. doi: 10.21681/2311-3456-2021-5-52-62 (in Russian).

27. Sokolovsky S. P. Parametric Optimization of Information Systems for Proactive Protection of Data Transmission Service from Network Reconnaissance. *Herald of Computer and Information Technologies*, 2022, vol. 19, no. 5 (215), pp. 49-57. doi: 10.14489/vkit.2022.05.pp.049-057 (in Russian).

28. Kuchurov V. V. Model and Technique for Abonent Address Masking in Cyberspace. *Voprosi Kiberbezopasnosti*, 2020, no. 6 (40). pp. 2-13. doi: 10.21681/2311-3456-2020-06-2-13 (in Russian).

29. Sokolovsky S. P. Information Systems Proactive Protection Suite Against Network Reconnaissance. *Herald of Computer and Information Technologies*, 2021, vol. 18, no. 11(209), pp. 53-62 doi: 10.14489/vkit.2021.11.pp.053-062 (in Russian).

30. Gorbachev A. A. Model and Parametric Optimization of Proactive Protection of the Email Service from Network Intelligence. *Voprosi Kiberbezopasnosti*, 2022, no. 3 (49), pp. 69-81. doi: 10.21681/4311-3456-2022-3-69-81 (in Russian).

31. Moskvin A. A. Model, Optimization and Efficiency Evaluation of Application Multicast Network Connections in Conditions of Network Intelligence. *Voprosi Kiberbezopasnosti*, 2023, no. 3 (55), pp. 13-22. doi: 10.21681/2311-3456-2023-3-13-22 (in Russian).

32. Kaplin M. A. Determination of Optimal Parameters for Configuring Information Systems in the Conditions of Network Intelligence. *Voprosi Kiberbezopasnosti*, 2022, no. 4 (50), pp. 80-90. doi: 10.21681/2311-3456-2022-4-80-90 (in Russian).

33. Sokolovsky S. P. Methodology for the Formation of Information Systems False Network Traffic for Protection Against Network Reconnaissance. *Herald of Computer and Information Technologies*, 2022, vol. 19, no. 2 (212), pp. 40-47. doi: 10.1489/vkit.2022.02.PP.040-047 (in Russian).

34. Feder J. *Fractals*. Moscow: URSS-LELAND Publ., 2014. 264 p (in Russian).

35. Grimmett G. *Percolation*. Cambridge: Springer, 1999. 444 p.

36. Shuvaev F. L., Tatarka M. V. Dynamics of Centrality Measures of Random Graph Mathematical Models. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2020, vol. 20, no. 2, pp. 249–256. doi: 10.17586/2226-1494-2020-20-2-249-256 (in Russian).

37. Piraveenan M., Prokopenko, M., Hossain L. Percolation Centrality: Quantifying Graph-Theoretic Impact of Nodes during Percolation in Networks. *PLoS ONE,* 2013, vol. 8, no. 1, pp. 53095. doi:10.1371/journal.pone.0053095.

38. Berest P. A., Bogachev K. G., Vigovskii L. S. *Sposob Sravnitelnoi Otsenki Struktur Informatsionno-Vichislitelnoi Seti* [Method of Comparative Evaluation of Information-Computer Network Structures]. Patent Russia, no. 2408928 C1, 2009 (in Russian).

39. Ignatenko A. V., Kovalevskii S. G., Maksimov R. V. *Sposob Sravnitelnoi Otsenki Struktur Informatsionno-Vichislitelnoi Seti* [Method of Comparative

Evaluation of Information-Computer Network Structures]. Patent Russia, no. 2450338 C1, 2011.

40. Aparin N. N., Astakhov A. I., Zhirakovskii A. A. *Sposob Sravnitelnoi Otsenki Struktur Informatsionno-Vichislitelnoi Seti* [Method of Comparative Evaluation of Information-Computer Network Structures]. Patent Russia, no. 2460123 C1, 2011.

41. Iskolnii B. B., Lazarev A. A., Likov N. Yu. *Sposob Sravnitelnoi Otsenki Struktur Informatsionno-Vichislitelnoi Seti* [Method of Comparative Evaluation of Information-Computer Network Structures]. Patent Russia, no. 2626099 C1, 2016.

42. Tsallis C., Stariolo D. A. Generalized Simulated Annealing. *Physica A*, 1996, vol. 233, pp. 395-406.

43. Xiang Y., Sun D. Y., Fan W., Gong X. G. Generalized Simulated Annealing Algorithm and Its Application to the Thomson Model. *Physics Letters A*, 1997, vol. 233, pp. 216-220.

44. Storn R., Price K. Differential Evolution – a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, *Journal of Global Optimization*, 1997, vol. 11, pp. 341–359.

45. Endres S., Sandrock C., Focke W. (2018). A Simplicial Homology Algorithm for Lipschitz Optimisation. *Journal of Global Optimization*, 2018, vol. 72, pp. 181-217. doi: 10.1007/s10898-018-0645-y.

46. Gablonsky J., Kelley C. A Locally-Biased form of the DIRECT Algorithm. *Journal of Global Optimization,* 2001, vol. 21, pp. 27-37.

47. Jones D. R., Perttunen C. D, Stuckman B. E. Lipschitzian Optimization without the Lipschitz Constant. *J Optim. Theory Appl.,* 1993, vol. 79, pp. 157-181.

48. Sherstobitov R. S., Maksimov R. V., Sharifullin S. R. Masking of Departmental-purpose Integrated Communication Networks. System of Control, Communication and Security, 2018, vol. 4, pp. 136-175 (in Russian).

49. Golub B. V., Krasnov V. A., Likov N. Yu., Maksimov R. V. *Sposob Maskirovaniya Strukturi Seti Svyazi* [Method of Masking the Structure of the Communication Network]. Patent Russia, no. 2645292, 2018.

50. Styugin M. Multilevel Decentralized Protection Scheme Based on Moving Targets. *International Journal of Security and Its Applications*, 2016, vol. 10, no. 1, pp. 45-54. doi: 10.14257/ijsia.2016.10.1.05 (in Russian).

51. Styugin M. Zashchita Internet-resursov po Tekhnologii Dvizhushcheisya Tseli [Protecting Online Resources on Moving Target Technology]. *Dokladi TUSUR*, 2015, no. 2 (36), pp. 80-85. Available at: https://cyberleninka.ru/article/n/zaschita-internet-resursov-po-tehnologii-dvizhuscheysya-tseli (accessed 11 November 2024) (in Russian).

Статья поступила 18 ноября 2024 г.

Информация об авторах

Максимов Роман Викторович – доктор технических наук, профессор. Заслуженный изобретатель Российской Федерации. Профессор 33 кафедры. Крас-

нодарское высшее военное училище. Область научных интересов: маскирование и проактивная защита информационных систем. E–mail: rvmaxim@ya.ru

Теленьга Александр Павлович – кандидат педагогических наук. Докторант. Краснодарское высшее военное училище. Область научных интересов: маскирование и проактивная защита информационных систем. E–mail: alexander.telenga@yandex.ru

Адрес: 350063, Россия, г. Краснодар, ул. Красина, д. 4.

The Concept of Countering the Identification of Corporate Information Systems Metastructures at the Level of Percolation Clusters in Cyberspace

R. V. Maximov, A. P. Telenga

Problem statement: *corporate information systems (CIS) metastructures are secondary structures that are formed under the influence of various processes and can be used to analyze and uncover the processes of CIS functioning in cyberspace using cyber intelligence (CI). The problem of countering the identification of information systems metastructures at the level of cyberspace percolation clusters can be formulated as a multi-criteria optimization problem using the metric of percolation adaptability as a masking optimality criterion.* ***Purpose.*** *The goal of this work is to modify a graph representing an information system by adding new vertices and edges to its single-mode projections in such a way as to increase its percolation adaptability, i.e., reduce the percolation centrality of vertices for each projection, reduce the similarity between the original and modified graph, and at the same time maximize the size of the connected component by adding a minimum number of vertices.* ***Methods used:*** *the solution of the problem of counteracting the CIS metastructures identification at the level of the cyberspace percolation clusters is based on the developed methodology of counteracting identification, including algorithms of masking and performance evaluation by random walk for the proposed model of the CIS at the level of the cyberspace percolation clusters, allowing the method of optimization DIRECT to determine the optimal parameters of masking.* ***Novelty.*** *The novelty element of the presented solution is the use of the percolation centrality index and the introduction of a new set of actions and links between them to determine the importance of graph vertices and the subsequent formation of a modified graph taking into account these criteria, which allows us to take into account not only the topological information of the graph, but also its percolation structure for the generation of new vertex and edge information.* ***Results:*** *the use of the presented solution to counteract the identification of CIS metastructures at the level of the cyberspace percolation clusters allows to increase its percolation adaptability, i.e. to reduce the percolation centrality of vertices for each projection, to reduce the similarity between the original and modified graph, and at the same time to maximize the size of the connected component by adding a minimum number of vertices. The simulation conducted to evaluate the performance of masking by random walks for a set of graphs with 5, 10, 15, 20, 25, 30, 35, 40, 45, 50 vertices showed a 40% reduction in the probability of traversing the vertices of the original graph in the masked graph (i.e., uncovering the original CIS structure) in 100 attempts for different criterion importance ratios.* ***Practical relevance.*** *The presented solution could be realized in the form of mathematical support of deception network information objects. This will allow to create more adaptive and effective mechanisms to counteract the identification of CIS metastructures in cyberspace, increasing the complexity of analyzing and detecting potential threats by an attacker.*

Key words: *metastructure, corporate information systems, masking, percolation, percolation adaptability, cyberspace, perplexity, t-SNE.*

Information about Authors

Roman Victorovich Maximov – Dr. habil. of Engineering Sciences, Full Professor. Professor of the Department number 33. Krasnodar higher military school. Field of research: masking and proactive defense of information systems. E-mail: rvmaxim@ya.ru

Alexander Pavlovich Telenga – Ph.D. of Pedagogical Sciences. Doctoral Student. Krasnodar higher military school. Field of research: masking and proactive defense of information systems. E-mail: alexander.telenga@yandex.ru

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