

М.Г. ФИРСОВА

**ОНТОЛОГИЧЕСКАЯ МОДЕЛЬ ДЛЯ АНАЛИЗА
ЗАЩИЩЁННОСТИ СИСТЕМ ИНТЕРНЕТА ВЕЩЕЙ***Фирсова М.Г. Онтологическая модель для анализа защищённости систем Интернета вещей.*

Аннотация. В статье рассматривается задача комплексного анализа защищённости систем Интернета вещей. Актуальность работы обусловлена ростом сложности и гетерогенности систем Интернета вещей, а также необходимостью масштабируемой оценки их защищённости с учётом архитектурных и функциональных особенностей. На основе анализа архитектуры систем Интернета вещей разработана многоуровневая онтологическая модель, объединяющая предметную область Интернета вещей и информационной безопасности, дополненная учётом функциональной роли компонентов системы, типов передаваемых сообщений и целей системы, что позволяет отобразить зависимость критичности компонента от выполняемой функции и влияние угрозы на нарушение целевой функции системы. На основе модели предложен механизм вывода угроз, выявляющий каскадный эффект распространения деструктивного воздействия по топологическому вектору и функциональному вектору. В отличие от подходов, рассматривающих угрозу исключительно через факт первичной компрометации элемента, новизна заключается в том, что глубина и последствия угрозы определяются не начальной компрометацией, а структурно-семантическими характеристиками межкомпонентных связей, задающих развитие каскадного эффекта. Онтология интегрирована с внешними репозиториями знаний (CVE, RVD, CAPEC, CWE, БДУ ФСТЭК) посредством ETL-процесса, а вывод скрытых связей автоматизирован с помощью SWRL-правил в среде Protege. Практическая апробация модели выполнена на экспериментальном стенде «Умная лаборатория». Результатом работы является реализованная онтологическая модель, обеспечивающая автоматическое выявление угроз безопасности для систем Интернета вещей. Оценка качества модели по методике OntoMod показала расширение семантического покрытия предметной области в 3 раза и снижение структурной сложности в 1,5-2,4 раза по сравнению с рассмотренными аналогами. Сформулирован вывод о применимости предложенного онтологического подхода для оперативной поддержки принятия решений операторами и проектировщиками распределённых сетей Интернета вещей.

Ключевые слова: онтология, Интернет вещей, угроза, структурно-функциональная характеристика, информационная безопасность.

1. Введение. В настоящее время наблюдается устойчивый рост числа компьютерных атак, нацеленных на системы Интернета вещей. При этом сами устройства Интернета вещей все чаще становятся не только целью, но и инструментом для осуществления этих атак, будучи объединёнными в ботнет-сети. Данная тенденция усугубляется неконтролируемым расширением поверхности атаки, что обусловлено как увеличением количества гетерогенных компонентов, так и множеством технологий, лежащих в основе их реализации [1].

В связи с этим возникает явное противоречие, с которым сталкивается оператор системы Интернета вещей: с одной стороны, наблюдается стремительный рост числа потенциальных угроз,

напрямую связанный с усложнением архитектуры системы [1, 2]. С другой стороны, перед оператором стоит практическая задача: не устранить все угрозы, а идентифицировать и структурировать актуальные угрозы для их последующей минимизации. Ситуация усугубляется спецификой самой парадигмы Интернета вещей. В отличие от традиционных информационных систем, Интернет вещей характеризуется тесной интеграцией вычислительных и физических процессов, использованием ограниченных по ресурсам устройств и многоуровневой архитектурой взаимодействия [2]. Устройства зачастую имеют прямой выход в глобальную сеть, располагаются в физически неконтролируемой зоне и используют беспроводные каналы связи, которые по своей природе являются уязвимыми. Компрометация отдельных компонентов может приводить к каскадным последствиям и оказывать влияние на функционирование физического объекта управления. Более того, попытка устранить угрозы, связанные, например, с особенностями беспроводной передачи данных, может повлечь неоправданно высокие затраты, ставящие под вопрос экономическую целесообразность всей системы.

Существующие подходы к обеспечению информационной безопасности Интернета вещей, как правило, базируются либо на применении обобщённых классификаций уязвимостей и атак, либо на экспертных методах [3 – 5]. При этом в большинстве работ отсутствует формализованное представление взаимосвязей между архитектурными компонентами системы, уязвимостями, атаками и возникающими угрозами, что затрудняет воспроизводимую и масштабируемую идентификацию угроз.

Таким образом, актуальной становится задача получения объективной картины угроз. Ключевая потребность заключается в определении того, на какие именно компоненты системы эти угрозы могут повлиять, какова степень критичности этих компонентов в контексте работы всей системы. При этом определить каким образом эта критичность зависит от конкретной области применения и сферы функционирования Интернета вещей – домена. Решение этой задачи требует формализации не столько самих угроз, сколько связей между архитектурными компонентами системы, их уязвимостями и сценариями атак с учетом контекста функционирования. Для такой формализации предлагается использовать онтологический подход, позволяющий построить модель формализованного представления знаний о предметной области (онтологию), объединяющую предметные области Интернета вещей и информационной безопасности. Именно такой подход способен отразить киберфизическую природу Интернета

вещей, где компрометация отдельного датчика или может привести к каскадным последствиям в физическом мире.

Целью работы является разработка онтологической модели, объединяющей предметную область Интернета вещей и информационной безопасности, обеспечивающей автоматизированную идентификацию и детальный анализ угроз с учётом архитектурных особенностей, межкомпонентного взаимодействия и специфики домена применения систем.

Объектом исследования является система Интернета вещей, предметом исследования – модели, семантические связи и логические правила формализации знаний предметной области, определяющие процессы автоматизированного выявления угроз информационной безопасности систем Интернета вещей на основе онтологического инжиниринга.

Для достижения поставленной цели необходимо решить следующие задачи:

- анализ архитектурных особенностей систем Интернета вещей, определяющих их уязвимость к компьютерным атакам и специфику каскадного распространения угроз;
- систематический обзор и сравнительный анализ существующих онтологических моделей и подходов к моделированию и оценке угроз информационной безопасности в сфере Интернета вещей;
- разработка структуры концептуальной и формальной онтологической модели, объединяющей концепты Интернета вещей и информационной безопасности;
- реализация разработанной онтологической модели в среде Protege и наполнение базы знаний экземплярами классов на основе интеграции гетерогенных внешних источников (CVE, RVD, CAPEC, CWE, БДУ ФСТЭК) с применением методов ETL;
- разработка системы логических правил на языке SWRL для автоматизации вывода скрытых зависимостей и сквозной декомпозиции векторов атак, угроз и рисков безопасности;
- экспериментальное исследование разработанной онтологии на натурном стенде «Умная лаборатория» и оценка её структурно-поведенческого качества с использованием метрик OntoMod в сравнении с существующими решениями.

Научная новизна работы заключается в разработке метода автоматического выявления каскадных угроз в системах Интернета вещей, основанного на логическом выводе с использованием онтологической модели, которая интегрирует предметные области Интернета вещей и информационной безопасности, формализует межкомпонентные связи

через семантически нагруженные объектные свойства и обеспечивает возможность многопараметрической оценки угроз для повышения защищенности систем Интернета вещей. Многопараметрическая оценка угрозы определяется совместным учётом трёх разнородных параметров: самого факта воздействия на компонент, количества связанных компонентов и семантического класса передаваемых сообщений. В отличие от подходов, оценивающих угрозу исключительно по факту первичной компрометации элемента, предложенный метод создает производные угрозы и распространяет их по функциональным связям, что позволяет выявить каскадный эффект, определяемый не простой связностью компонентов, а семантикой их взаимодействия. На основе выведенных производных угроз и связи компонентов с целями системы метод устанавливает факт нарушения целевых функций, делая вывод о компрометации цели не напрямую из факта взлома, а как результат цепочки логически обоснованных шагов, что обеспечивает выявление скрытых путей нарушения целей функций системы.

2. Архитектурные особенности систем Интернета вещей.

Типовая архитектура систем Интернета вещей включает совокупность гетерогенных устройств, сетевых компонентов, программного обеспечения и сервисов обработки данных, которые являются структурно-функциональными характеристиками (СФХ) системы и взаимодействуют в рамках распределённой среды. В общем случае в структуре Интернета вещей выделяют сенсорный уровень (сенсоры и актуаторы), сетевой уровень для передачи данных и сетевого взаимодействия, а также уровень контроллеров, реализуемый с использованием локальных, туманных или облачных вычислений для управления и обработки данных [6] (Рис. 1).

Оператор инициализирует работу системы, передавая управляющие команды («1» стрелка). Далее сигнал через сеть направляется к контроллеру, который непосредственно отвечает за приведение в действие актуатора. В современных реализациях промежуточным звеном может выступать облачная платформа [5], выполняющая роль аналитического центра, или же уровень туманных вычислений, обеспечивающий локальную обработку и снижение сетевых задержек [7]. Результатом прохождения команды является физическое изменение состояния объекта управления.

Потоки данных иницируются либо запросом от оператора, либо автоматически – сенсорами. При запросе данных («5» стрелка) контроллер, получив сигнал через сеть, опрашивает соответствующий сенсор и возвращает информацию оператору («4» стрелка). В фоновом режиме сенсоры непрерывно генерируют измерительную информацию

о параметрах внешней среды («2» стрелка). Эта информация передается на контроллер для анализа. Для повышения эффективности анализа могут привлекаться технологии искусственного интеллекта или машинного обучения, при этом вычислительная нагрузка может распределяться между облаком и туманными узлами. Результаты мониторинга выводятся на интерфейс оператора, а в случае обнаружения отклонений или по заданному алгоритму контроллер инициирует управляющее воздействие, передавая команду актуатору («3» стрелка), как того требует принцип обратной связи [8].



Рис. 1. Схема взаимодействия компонентов системы Интернета вещей

Задача идентификации угроз, как части процесса оценки защищённости, состоит из определения множества актуальных угроз для заданной системы Интернета вещей на основе анализа её архитектуры и известных уязвимостей. Угроза считается актуальной, если она может быть реализована в системе Интернета вещей и впоследствии приводит к негативным последствиям для самой системы, окружающей среды или человека.

Для решения поставленной задачи предлагается использовать онтологический подход, в рамках которого онтология выступает не только как средство формализованного описания предметных областей Интернета вещей и информационной безопасности, но и как инструмент логического вывода. Онтологическая модель позволяет явно задать отношения между компонентами системы, уязвимостями, атаками и угрозами, а система логических правил обеспечивает автоматическое порождение новых знаний.

3. Обзор существующих онтологий и подходов к идентификации и оценке угроз. Исследования в области информационной безопасности систем Интернета вещей охватывают широкий спектр подходов, включая анализ архитектурных уязвимостей, разработку моделей угроз и применение онтологий для представления знаний. Целью данного обзора является анализ существующих исследований, посвященных моделированию и идентификации угроз систем Интернета вещей с фокусом на работы, использующие онтологический подход. Критериями анализа выступают: (1) полнота охвата архитектуры Интернета вещей, (2) концепты онтологии, (3) интеграция с внешними базами знаний, (4) наличие формализованного механизма вывода скрытых логических связей для идентификации угроз, (5) учёт домена функционирования системы, целей системы, (6) наличие взаимосвязей между компонентами системы.

Проведенный анализ позволяет выделить несколько направлений в развитии подходов к анализу защищённости на основе онтологий и графов атак. Основы применения онтологического подхода в области информационной безопасности были заложены в [9]. Была предложена публично доступная онтология, моделирующая активы, угрозы, уязвимости и контрмеры. Ключевой особенностью данной онтологии является введение отношения, связывающего контрмеру с активом, целью безопасности и стратегией защиты. Авторы впервые продемонстрировали возможность использования логического вывода для классификации угроз и контрмер по целям безопасности, что позволяет отвечать на сложные запросы к модели без ручного перечисления, а также служит основой для будущих алгоритмов ранжирования контрмер против конкретных угроз. Развивая концептуальные основы, в [10] была обоснована необходимость модульного построения онтологий информационной безопасности с выделением верхнего, среднего и доменного уровней, а также подчеркнута важность интеграции с существующими стандартами, такими как CVE (Common Vulnerabilities and Exposures), CAPEC (Common Attack Pattern Enumeration and Classification) и MAEC (Malware Attribute Enumeration and Characterization).

Применение семантических технологий для анализа компьютерных атак получило развитие в работе [11], предложившей онтологию графа атак на основе RDF (Resource Description Framework)/OWL (Web Ontology Language) с транзитивным свойством, позволяющим выполнять логический вывод о возможных векторах атак. Развивая этот подход, в работе [12] была реализована онтология

графа атак в среде Protege, показав преимущества SPARQL-запросов для извлечения информации о векторах атак и привилегиях в корпоративных сетях. Однако данные работы не рассматривали подробно СФХ системы и не позволяли выстроить причинно-следственные связи между компонентами системы и их угрозами. Более поздние исследования интегрируют онтологии с графами атак для динамической оценки и реагирования. В [13] было предложено комплексное решение, в котором логический граф атак обогащается данными из онтологии уязвимостей VDO (Vulnerability Description Ontology), а выбор контрмер осуществляется путем сопоставления с базой знаний D3FEND (Detection, Denial, and Disruption Framework Empowering Network Defense) и многокритериальной оптимизации реагирования для киберфизических систем (CPS).

С ростом популярности парадигмы Интернета вещей появились работы, адаптирующие онтологические подходы к специфике этой предметной области. Масштабный систематический обзор [14] показал, что основной целью применения онтологий в Интернете вещей является обеспечение совместимости (31,30% работ), в то время как вопросам безопасности и приватности уделяется существенно меньше внимания (13,04% работ). Отмечается, что лишь 27,83% исследований повторно используют существующие онтологические наработки, что свидетельствует о фрагментарности и отсутствии единой стандартизированной модели. В работе [1] описан подход, который позволяет рассчитывать интегральные метрики безопасности для устройств Интернета вещей на основе анализа уязвимостей, разрешений приложений и онтологического представления политик конфиденциальности. Другой подход предложен в работе [15], где разработана базовая онтология безопасности для промышленного Интернета вещей (IIoT), включающая классы Asset, Attack, Fault, SecurityGoal, Vulnerability, Risk, Capability, Criticality. Модель учитывает асимметричные связи между возможностями атакующего и атаками, а также связь риска с критичностью. Однако она не интегрируется с внешними базами данных уязвимостей и не содержит формализованных правил логического вывода для идентификации угроз (лишь декларируется возможность использования SWRL в будущем). Практическая реализация онтологического подхода для обеспечения безопасности устройств Интернета вещей представлена в [16], где разработана онтология, связывающая уязвимости устройств с лучшими практиками безопасности. На примере промышленного стенда авторы продемонстрировали использование онтологии и данных из базы CVE

для анализа реализованных мер защиты, однако предложенный подход не включает механизмы логического вывода с целью вывода скрытых связей о состоянии защищённости системы. Близкий по духу подход, но с фокусом на SIEM, представлен в [17, 18], где формализованы понятия сообщения и события, введены явные и неявные связи между устройствами, однако отсутствует интеграция с внешними базами знаний об уязвимостях и нет реализации онтологии на OWL.

Важным направлением исследований является разработка контекстно-зависимых онтологий безопасности, учитывающих специфику окружения и роли участников в системе Интернета вещей. В работе [19] предложена онтология контекстно-зависимой безопасности для Интернета вещей, построенная на базе STAC (Security Toolbox: Attacks & Countermeasures) и реализованная с использованием SWRL-правил в Protege. Авторы демонстрируют, как на основе контекста (тип пользователя, архитектура, протокол) можно выполнять логический вывод для рекомендации соответствующих контрмер. Данный подход учитывает доменную специфику на примере здравоохранения, однако не интегрируется с открытыми базами данных.

Наряду с академическими исследованиями, важную роль в стандартизации играют официальные спецификации, разработанные международными организациями. Эталонная онтология SAREF (Smart Applications REference) [20], разработанная Европейским институтом телекоммуникационных стандартов (ETSI), представляет собой пример успешной стандартизации в области Интернета вещей, обеспечивая семантическую совместимость за счёт модульной структуры и согласования с существующими стандартами, такими как W3C® Semantic Sensor Network (SSN) и oneM2M. Архитектурный принцип расширяемости SAREF позволяет создавать доменно-специфичные модули, что концептуально близко к вводимому в настоящей работе понятию «домена функционирования». Однако SAREF не охватывает аспекты информационной безопасности. Другой важной инициативой является онтология безопасности Web of Things (WoT) Security Ontology [21], представляющая собой стандартизированный словарь для описания механизмов аутентификации и авторизации, включающий классы для различных схем безопасности (OAuth2, Basic, Bearer, PSK и др.). Вместе с тем она ограничена описанием требований к безопасности и не охватывает моделирование угроз, уязвимостей или рисков.

Параллельно с развитием онтологических моделей формировалось направление исследований, ориентированное на анализ и оценку защищённости с использованием открытых баз знаний. В работе [22] предложена иерархия статических и динамических показателей, объединяющая оценки рисков нарушения CIA, рисков на основе анализа политик конфиденциальности и рисков на основе анализа APK-файлов. Ключевой особенностью является разработка оригинальных алгоритмов вычисления интегральной оценки с нелинейным логарифмическим масштабированием, что позволяет не «усреднять» показатели, а сохранять вклад наиболее критичного из них. При этом авторы для анализа защищённости используют данные только об APK-файлах, не учитывая полную архитектуру системы. В [23] предлагается онтологическая модель, автоматизирующая расчет интегральных метрик защищённости в реальном времени и динамически адаптирующаяся к домену через корреляцию событий и подключение баз знаний, включая матрицы MITRE ATT&CK и ATT&CK for ICS. В работе [24] предложен подход к оцениванию защищённости гетерогенных инфраструктур на основе графов атак, формируемых с использованием баз NVD и MITRE ATT&CK, и реализован расчёт вероятностей реализации атак с применением байесовского вывода. В [25] был представлен масштабный обзор угроз безопасности Интернета вещей, систематизировав атаки по уровням архитектуры и детально проанализировав анатомию вредоносного программного обеспечения, что сформировало требования к полноте разрабатываемых моделей для систем Интернета вещей. Важность учёта контекста применения системы при анализе защищённости подчеркивается в работе [26], в которой при обзоре проблем безопасности робототехники явно выделены различные домены применения (медицина, промышленность, энергетика и т.д.) и показано, что последствия реализации атак существенно различаются в зависимости от домена. Однако предложенный авторами анализ носит описательный характер и не предлагает формального способа учёта домена.

Анализ существующих онтологических решений в области Интернета вещей и информационной безопасности показал, что большинство рассмотренных онтологий ориентированы на решение частных задач и не обеспечивают комплексную поддержку анализа защищённости систем Интернета вещей. Существенная часть онтологий предназначена для описания устройств, протоколов взаимодействия и сервисов Интернета вещей и используется преимущественно для обеспечения совместимости и интеграции компонентов. В таких моделях аспекты информационной безопасности либо отсутствуют, либо представлены без формализованных связей

с архитектурой системы. Отдельную группу составляют онтологии, направленные на представление знаний об уязвимостях и атаках. Эти решения позволяют структурировать сведения о типах уязвимостей и сценариях атак, однако, как правило, не учитывают архитектурные особенности Интернета вещей и не связывают угрозы с компонентами объекта. В результате идентификация угроз осуществляется вне контекста реальной конфигурации системы, что ограничивает применимость таких онтологий для анализа защищённости.

В таблице 1 представлено сравнение работ, в которых онтологическая модель объединяет понятия Интернета вещей и информационной безопасности.

Таблица 1. Сравнительный анализ работ

Работа	Охват архитектуры	Концепты модели	Интеграция с базами знаний	Механизм вывода	Учёт домена и целей системы	Наличие межкомпонентного взаимодействия
[1]	Нет	Static CIA score, APK based score, Privacy policy score, Readability score, Dynamic CIA/privacy score	NVD, CPE, CVE	Нет	Косвенно	Нет
[15]	Да	Asset, Attack, Fault, SecurityGoal, SecurityMechanism, Vulnerability, Capability, Criticality, Risk	IICSF, INSSA	Потенциально	Нет	Да
[16]	Да	Vulnerability, Best Security Practice, System Component, Stakeholder	CVE, IEEE Sensors Registry	SPARQL-запросы	Косвенно	Да
[17]	Да	Message, Event, Security Threat, Device, Network Parameter	Нет	Да	Нет	Нет
[19]	Да	TypeOfContext, Threat, SecurityAsset, SecurityProperty, SecurityMechanism, ArchitectureType, Protocol	STAC, LOV4IoT	SWRL	Да	Да
[20]	Да	Device, Function, Command, State, Property, Service, Measurement, Task, Commodity, Profile, FeatureOfInterest	SSN, oneM2M, Time Ontology, OM	OWL (подразумевается)	Да	Да
[21]	Да	No, Auto, Basic, Digest, APIKey, Bearer, PSK, OAuth2, Combo	OpenAPI, IETF RFC	RDF/SPARQL	Косвенно	Да
[22]	Да	Confidentiality, Integrity, Availability	CVE, CPE, NVD	Нет	Косвенно	Да

Продолжение таблицы 1. Сравнительный анализ работ

[25]	Да	Уровни Интернета вещей, Угрозы, Уязвимости, Атаки, Malware, Ботнеты, Контрмеры	NVD (CVE), CERT	Не рассматривается	Упоминается, но не используется	Да
[26]	Да	Домены, Уязвимости, Атаки, Риски, Контрмеры	NVD (CVE)	Не рассматривается	Да	Да

В существующих исследованиях отсутствует комплексная семантическая модель анализа защищенности систем Интернета вещей, которая одновременно учитывала бы (1) топологическую и функциональную связность компонентов, (2) распространение угроз по этим связям, включая явные и неявные логические зависимости, (3) функциональные цели системы и то, как их нарушение связано с компрометацией компонентов. Кроме того, отсутствует целостная онтологическая цепочка «компонент – его уязвимость – атака на компонент – распространение угрозы на связанные компоненты – риск». Без такой цепочки невозможно формализовать каскадные эффекты компрометации и, как следствие, оценить, как реализация атаки на отдельном компоненте приводит к нарушению работы системы как целого. Необходимость построения новой онтологической модели обусловлена потребностью преодоления этого разрыва.

4. Разработка онтологической модели информационной безопасности систем Интернета вещей. Предлагаемая онтологическая модель формализует знания в области безопасности Интернета вещей, объединяя архитектурные характеристики систем, сведения об уязвимостях и атаках из общедоступных баз данных, а также контекстную информацию о домене функционирования объекта, целях системы, зависимостях между компонентами и потоках сообщений. Ключевое отличие модели – ориентация на киберфизическую природу Интернета вещей с учётом динамики информационного обмена и каскадных эффектов.

На основе анализа архитектуры систем Интернета вещей [2, 6] выделены ключевые понятия предметной области. Рисунок 2 представляет концептуальную модель, которая задает каркас для построения онтологической модели, детально описанной в разделах 4.1-4.5.

Данная концептуальная модель отражает предметную область Интернета вещей и информационной безопасности, взаимосвязи между ними, а также сохраняет преемственность по отношению к базовой архитектуре системы (Рис. 1). Модель наглядно показывает, как деструктивные воздействия влияют на каждый уровень системы, а также описывает цепочку «уязвимость – атака – угроза – риск – ущерб», интегрируя технические аспекты безопасности с их конечными

последствиями. Основными концептами модели являются «Объект» (совокупность уровней системы), «СФХ» (компоненты каждого уровня), «Уязвимость», «Атака», «Риск», «Угроза». В рамках данной работы под объектом понимается система Интернета вещей.

В рамках предлагаемой концептуальной модели именно угроза является центральным, системообразующим понятием, которое придает смысл и контекст всем остальным элементам. Угрозо-ориентированный подход не исключает анализ уязвимостей и атак, а интегрирует их, обеспечивая причинно-следственную связь. Его ключевое преимущество заключается в том, что угроза представляет собой комплексное понятие, включающее не только технические аспекты (атаки и уязвимости), но и контекст реализации, потенциальные последствия и особенности защищаемого объекта. Именно эта способность к интеграции делает угрозу центральным концептом модели, преобразующим набор разрозненных данных (перечень уязвимостей, классификация атак) в целостную картину безопасности системы Интернета вещей.

Определение основ и структуры концептуальной модели Интернета вещей с точки зрения безопасности позволяет перейти к построению онтологической модели, объединяющей предметные области Интернета вещей и информационной безопасности (Рис. 3).

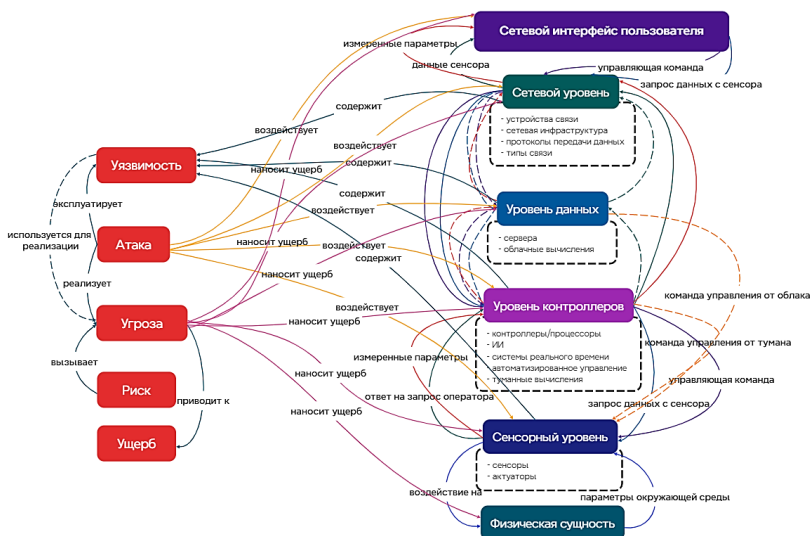


Рис. 2. Концептуальная модель системы Интернета вещей с точки зрения безопасности

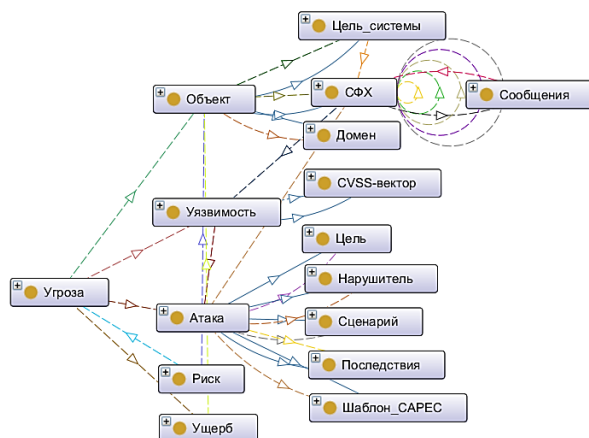


Рис. 3. Ключевые концепты онтологической модели, объединяющей предметную область Интернета вещей и информационной безопасности

Для формального представления структуры разрабатываемой онтологической модели определим её как кортеж, задающий сигнатуру знаний:

$$\Sigma = \langle C, R, D, A \rangle, \quad (1)$$

где C – конечное множество классов (концептов) предметной области, R – множество объектных свойств (бинарных отношений), связывающих экземпляры, D – множество свойств-данных, связывающих экземпляры с литеральными значениями, A – множество аксиом, определяющих семантику классов и свойств.

В рамках определённого таким образом базиса Σ содержательное наполнение онтологии задаётся путём спецификации подмножеств классов, свойств и аксиом, описывающих ключевые концепты предметной области. В частности, вводятся следующие онтологические соглашения, связывающие понятия Интернета вещей и информационной безопасности.

Система Интернета вещей представлена классом объект (Object). Каждый экземпляр этого класса характеризуется областью функционирования, – с этой целью в онтологию введён класс «домен» (Domain), связанный с объектом отношением hasDomain. Любой объект состоит из множества компонентов – экземпляров класса СФХ (SFC), – соединённых с объектом отношением hasSFC.

Система Интернета вещей всегда предназначена для достижения некоторого конечного результата, что формализовано введением класса «цель системы» (SystemGoal). Объект связан с одной или несколькими целями через отношение hasGoal, а цель, в свою очередь, реализуется конкретными СФХ (отношение realizedBySFC).

Для моделирования информационного обмена между компонентами введён класс «сообщение» (Message). В отличие от прямых объектных связей между структурно-функциональными характеристиками, которые фиксируют исключительно факт и направление взаимодействия, сообщения как самостоятельные сущности позволяют учесть семантику передаваемой информации, а именно – тип данных и целевое назначение передаваемого потока. Каждое сообщение характеризуется отправителем и получателем (объектные свойства hasSrc и hasDst соответственно), а также атрибутами, определяющими тип и роль сообщения в функционировании системы. Выделение сообщений в качестве отдельных элементов онтологической модели имеет принципиальное значение для анализа распространения угроз. Компрометация компонента-отправителя не означает автоматической компрометации всех связанных с ним компонентов; вместо этого характер деструктивного воздействия, передаваемого по цепочке взаимодействия, определяется типом сообщения. Так, подмена данных мониторинга и искажение управляющей команды представляют собой качественно различные виды воздействия, приводящие к различным последствиям для целевых функций системы. Таким образом, сообщения выступают связующим звеном, через которое первичная угроза преобразуется в производные угрозы, классифицированные в соответствии с семантикой взаимодействия.

Кроме того, между СФХ определены прямые объектные отношения, отражающие физические и логические связи в системе:

- connectsTo – обобщённая связь (родительское свойство);
- sendsDataTo – передача данных;
- controls – управляющее воздействие;
- receivesFrom – обратное свойство к sendsDataTo.

Данные отношения задают базовую архитектуру системы, а их совместное использование с сообщениями обеспечивает возможность моделирования каскадного распространения угроз не на основе простого отношения связности, а с учётом типа взаимодействия и содержания передаваемой информации. Компрометация одного компонента порождает производные угрозы для связанных с ним элементов, при этом характер порождаемой угрозы

определяется семантикой связи и типом сообщения, что позволяет дифференцировать последствия каскадного эффекта в зависимости от функциональной роли затронутых компонентов.

Каждая СФХ может содержать слабые места, для их представления служит класс «уязвимость» (Vulnerability), связанный с классом SFC отношением hasVulnerability. Каждая уязвимость может быть проэксплуатирована атакой – этот факт фиксируется при помощи класса «атака» (Attack) и отношения exploitedBy между атакой и уязвимостью. Успешная реализация атаки приводит к возникновению угрозы (Threat), которая связывается с атакой отношением realizedBy. Наконец, риск (Risk) является вероятностью реализации угрозы (свойство causedByThreat) с последующим ущербом (свойство leadsToDamage).

Область исследования ограничена угрозами от целенаправленных атак нарушителя, что обосновано спецификой Интернета вещей, где такие атаки представляют наихудший сценарий с потенциальным физическим ущербом. В контексте разрабатываемой онтологической модели нарушитель рассматривается как неотъемлемый атрибут атаки, что позволяет сохранить иерархическую стройность модели без потери семантической полноты. Исключение случайных событий (ошибок, отказов) необходимо для избежания громоздкости моделей и обеспечения последовательности угроз-ориентированного подхода.

Далее для компактной формальной записи структуры классов и связей в онтологии используется нотация, приближенная к синтаксису языка описания онтологий *Ontology Web Language* (OWL). В данной нотации символ $\subseteq$ означает «подкласс» (subClassOf), $\exists$ – экзистенциальное ограничение «существует хотя бы один», $\cap$ – пересечение классов (логическое «И»), а запись вида $\exists R.C$ соответствует наличию отношения R хотя бы с одним экземпляром класса C . Объектные свойства обозначаются именем свойства (например, $\exists \text{hasSFC.SFC}$). Данные обозначения используются исключительно для формального описания онтологии.

Онтологическая модель расположена на веб-сервисе GitHub [27].

4.1. Объект. Угроза направлена на объект, который характеризуется областью функционирования, определяющей предметную область его применения, целью, к достижению которой стремится система, а также совокупностью СФХ, необходимых для достижения целей системы и выступающих первичным фактором уязвимости, архитектуры и функциональных возможностей, и информационными потоками (сообщениями) между СФХ, включая связи управления и передачи данных, что позволяет моделировать

каскадное распространение угроз. Формально «Объект» можно представить следующим образом:

$$Object \subseteq (= 1hasDomain.Domain) \cap (\exists hasSFC.SFC) \cap (\exists hasGoal.SystemGoal), \quad (2)$$

где *Object* – класс системы Интернета вещей, *Domain* – класс домена (области функционирования), *SFC* – класс структурно-функциональных характеристик, *SystemGoal* – класс цели системы; *hasDomain*, *hasSFC*, *hasGoal* – объектные свойства, связывающие объект с соответствующими сущностями. Данное ограничение утверждает, что каждый экземпляр объекта обязан иметь один домен, хотя бы одну СФХ и хотя бы одну цель.

Объекту сопоставляется свойство «уровень критичности», которое отображает процент нарушения работоспособности объекта, функционирующего в определенной области, в случае отказа СФХ, реализующих определенную цель объекта:

$$S_{Object} = [S_D, S_G, S_{SFC}], \quad (3)$$

где S_{Object} – уровень критичности объекта, S_D – уровень критичности домена, S_G – уровень критичности цели объекта, S_{SFC} – максимальная опасность от каждой узвимой СФХ.

На рисунке 4 представлена модель «Объекта».

Цели системы определены на основе анализа стандартов в области Интернета вещей [2, 6, 28 – 31]. Домены выделены в соответствии со сферами функционирования субъектов критической информационной инфраструктуры Российской Федерации согласно [32], что придает модели практическую значимость. Для определения СФХ задействованы параметры из стандарта ETSI TS 103 264 [33] и онтологии SAREF [20]. Использование стандарта ETSI TS 103 264 позволяет привести модель в соответствие с глобальными архитектурными подходами к Интернету вещей, а применение онтологии SAREF обеспечивает совместимость с существующими решениями в этой области.

Как было сказано ранее, между СФХ были определены отношения, отражающие физические и логические связи в системе (Рис. 5).

Представленная схема отражает транзитивный характер связей между компонентами, что позволит в последующем выявлять каскадные угрозы. Например, компрометация контроллера влечет автоматическое распространение угрозы на все связанные актуаторы.

Анализ задействованности компонентов позволит объективно оценить их критичность по топологической значимости. Это смещает фокус внимания на «дешевые», но высокосвязанные элементы, которые могут служить векторами проникновения.

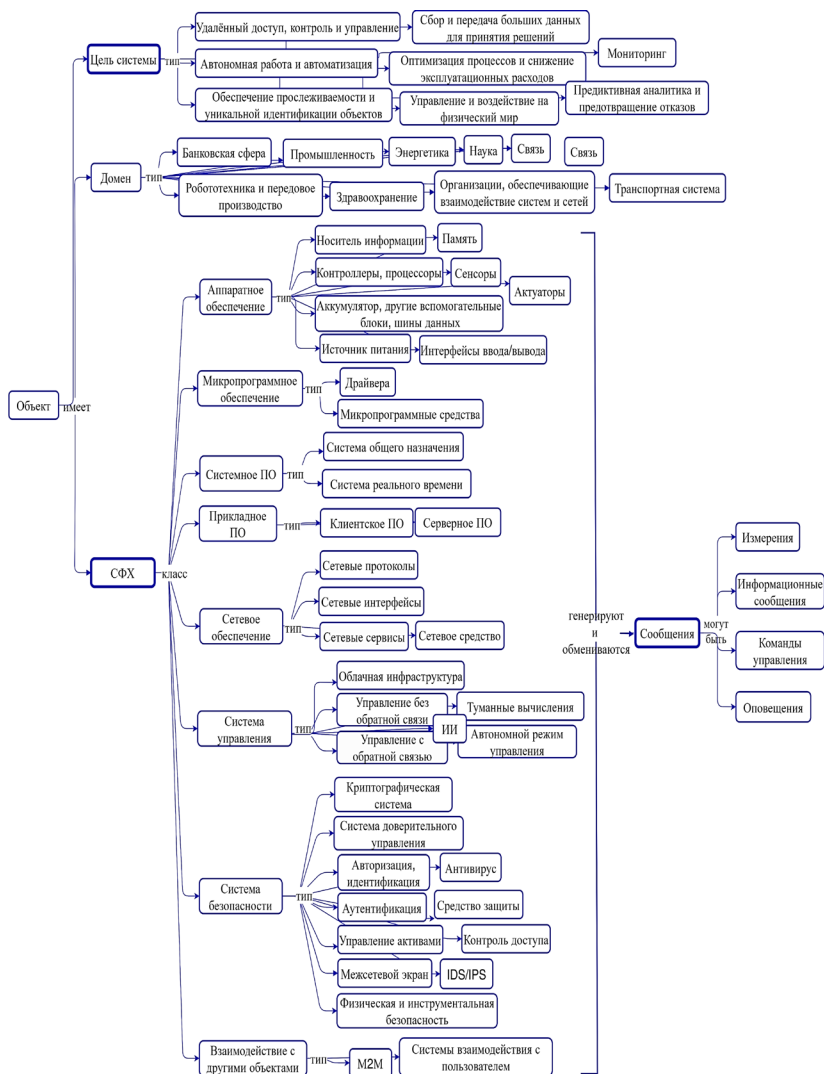


Рис. 4. Модель «Объекта»

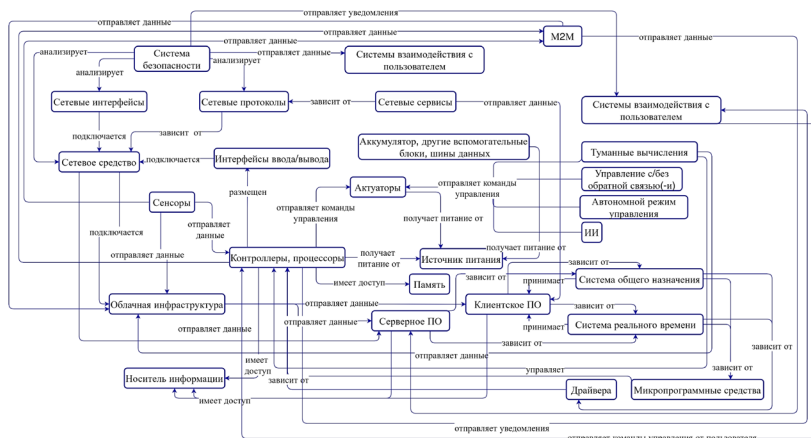


Рис. 5. Схема связей между СФХ

4.2. Уязвимость. Уязвимость рассматривается как недостаток или слабость реализации СФХ объекта, эксплуатация которой возможна в процессе атаки и приводит к формированию угрозы. Формально «Уязвимость» определяется как:

$$Vu l n e r a b i l i t y \subseteq (\exists h a s V u l n e r a b i l i t y ^{-}. S F C) \cap (\exists e x p l o i t e d B y . A t t a c k), \quad (4)$$

где $h a s V u l n e r a b i l i t y ^{-}$ – инверсное свойство к $h a s V u l n e r a b i l i t y$, связывающее уязвимость с компонентом, $e x p l o i t e d B y$ – «эксплуатируется атакой», $S F C$ – класс СФХ; $A t t a c k$ – класс атак.

Источниками данных об уязвимостях выбраны базы CVE [34] и RVD [35]. Обращение к репозиторию CVE обеспечивает методологическую совместимость результатов с устоявшимися практиками в области информационной безопасности и создает предпосылки для корректной интеграции разработанной модели с иными отраслевыми стандартами. Включение базы RVD позволяет охватить спектр специфичных для систем Интернета вещей уязвимостей, которые могут не отражаться в общих реестрах. Существенным преимуществом выступает наличие механизма соотнесения записей RVD с идентификаторами CVE, что гарантирует непротиворечивость и сравнительную сопоставимость данных в ходе аналитической обработки.

Описание уязвимостей осуществляется с использованием векторов оценки CVSS [36] или RVSS, позволяющих формализовать их характеристики. Данный выбор обусловлен тремя ключевыми

факторами: (1) наличием формализованных, машиночитаемых форматов и API, что критически важно для автоматизации последующих процессов ETL (Extract, Transform, Load); (2) широкой распространённостью и признанием в академическом и индустриальном сообществе, что обеспечивает воспроизводимость и сравнимость результатов с мировыми аналогами; (3) прямыми семантическими связями между базами (CVE – CWE – CAPEC), что позволяет строить формальные цепочки логического вывода. В соответствии с данными базы CVE уязвимости сопоставляется свойство уровень опасности, отражающий степень потенциального воздействия на объект:

$$S_{Vulnerability} = \{m_E, m_I, m_S\}, \quad (5)$$

где $S_{Vulnerability}$ – уровень опасности уязвимости, m_E, m_I, m_S – базовые метрики уязвимости в соответствии с CVSS.

Наряду с международными источниками, разработанная онтологическая модель ориентирована на совместимость с нормативной базой Российской Федерации. В частности, для оценки уровня критичности уязвимостей может быть использована «Методика оценки уровня критичности уязвимостей программных, программно-аппаратных средств» [37]. Данная методика устанавливает порядок оценки и обязательна к применению операторами информационных систем, в том числе объектов критической информационной инфраструктуры. Она определяет уровень критичности уязвимости на основе базовой оценки по шкале CVSS, дополненной контекстными метриками. Таким образом, предлагаемая модель не противопоставляет международный и отечественный подходы, а обеспечивает возможность работы с обоими.

4.3. Атака. Реализация угрозы осуществляется посредством атаки, эксплуатирующей уязвимость. Атака описывается через набор ключевых атрибутов, включающих субъект реализации (нарушитель), целевой компонент (цель), последовательность действий (сценарий) и потенциальные результаты (последствия). Формально «Атака» определяется в следующем виде:

$$Attack \subseteq (\exists exploitedBy^-. Vulnerability) \cap (\exists hasTarget.SFC) \cap (\exists realizedBy^-. Threat), \quad (6)$$

где $exploitedBy^-$ – «эксплуатирует уязвимость», $hasTarget$ – «нацелена на СФХ», $realizedBy^-$ – обратное к $realizedBy$, означающее, что атака реализует угрозу.

В качестве источника данных об атаках выбран каталог шаблонов атак CAPEC [38], который имеет семантические связи с базой уязвимостей CVE, что позволяет проследивать корреляцию между конкретными уязвимостями и потенциальными векторами их эксплуатации. Кроме того, CAPEC совместим с каталогом CWE, что повышает точность кросс-анализа данных.

В соответствии с каталогом CAPEC атака характеризуется вероятностью реализации, определяющей успешность её выполнения нарушителем, а также уровнем влияния, отражающим масштаб последствий при успешной реализации, которые в совокупности формируют свойство «уровень критичность атаки»:

$$S_{Attack} = [L_{Attack}, I_{Attack}], \quad (7)$$

где S_{Attack} – уровень критичности атаки, L_{Attack} – вероятность реализации атаки (заимствованный параметр «Likelihood of Attack» из CAPEC), I_{Attack} – уровень влияния атаки (заимствованный параметр «Typical Severity» из CAPEC).

На рисунке 6 представлена модель «Атаки».

В качестве целей атак выбраны ключевые параметры безопасности Интернета вещей, выделенные в эталонной архитектуре [2]. Типология нарушителей в разработанной модели соответствует подходам, зафиксированным в Методике оценки угроз безопасности информации ФСТЭК России от 5 февраля 2021 г. [39]. За основу последствий реализации атак взята классификация «ConsequencesType» из каталога CAPEC. Однако предложенная модель не ограничивается исключительно стандартизированными категориями: она дополнена перечнем потенциальных последствий, выявленных в ходе анализа исследований [31, 40 – 43]. Такое расширение таксономии позволяет оценивать воздействие атак не только на технологическую составляющую системы, но и на конечных пользователей.

Хотя разработанная онтологическая модель фокусируется на технических аспектах атак, для обеспечения совместимости с существующими методологиями в нее заложена базовая типизация нарушителя. В текущей реализации атрибуты нарушителя не участвуют в правилах логического вывода для идентификации угроз, однако класс «Нарушитель» введен в модель. Типология нарушителей соответствует подходам, зафиксированным в Методике оценки угроз безопасности информации ФСТЭК России. На текущем этапе исследования, посвященном идентификации наличия угроз, учет нарушителя не является критическим, так как наличие задокументированного

шаблона атаки (CAPEC) уже свидетельствует о принципиальной возможности ее реализации.



Рис. 6. Модель «Атаки»

4.4. Угроза. Угроза определяется как потенциальное событие, при котором нарушитель эксплуатирует уязвимости СФХ объекта, функционирующего в определенной области, путём реализации атаки, что приводит к негативным последствиям как для объекта, так и для окружающей среды и человека. Формально «Угроза» определяется в следующем виде:

$$Threat \subseteq (\exists affectObject. Object) \cap (\exists realizedBy. Attack) \cap (\exists causedByVuIne rability. VuIne rability), \quad (8)$$

где *affectObject* – «воздействует на объект», *realizedBy* – «реализуется атакой», *causedByVulnerability* – «обусловлена уязвимостью».

В данной работе СФХ являются параметрами безопасности. Угроза существует для объекта, если СФХ имеет уязвимости и возможность проэксплуатировать эти уязвимости путём реализации атаки [44]. При этом следует учитывать, что отсутствие известного вектора атаки в текущий момент не гарантирует защищённости в долгосрочной перспективе: уязвимость может быть потенциально опасной, если для нее теоретически возможна эксплуатация, даже если соответствующий шаблон атаки еще не зафиксирован в общедоступных базах знаний. В разработанной онтологической модели данный аспект учитывается путем классификации угроз по статусу актуальности. Угроза считается актуальной, если для неё существует известная атака, и потенциальной – при наличии уязвимости в отсутствие известного эксплойта.

Формализация процедуры идентификации угроз базируется на анализе онтологических взаимосвязей, установленных между СФХ объекта, уязвимостями и атаками. В качестве основы для идентификации угроз используется банк данных угроз безопасности информации ФСТЭК России (БДУ ФСТЭК) [45], где для каждой угрозы определены объекты воздействия. В контексте настоящего исследования такими объектами воздействия выступают СФХ.

Угроза имеет свойство «уровень опасности», которое указывает на потенциальную возможность нарушения безопасности и нанесения ущерба объекту, что определяется вероятностью эксплуатации слабых мест СФХ системы нарушителем путём реализации атак [46]:

$$S_{Threat} = [P, SFC], \quad (9)$$

где S_{Threat} – уровень опасности угрозы, $P = P(Vulnerability, Attack)$ – вероятность эксплуатации слабых мест атак.

4.5. Риск. Риск интерпретируется как сочетание вероятности реализации угрозы и ожидаемого ущерба от её реализации [46]. Формально «Риск» определяется в следующем виде:

$$Risk \subseteq (\exists causedByThreat.Threat) \cap (\exists leadsToDamage.Damage), \quad (10)$$

где *causedByThreat* – «вызван угрозой», *leadsToDamage* – «приводит к ущербу».

Риск имеет свойство «уровень критичности», отражающее степень негативного воздействия угрозы и масштаба потенциального ущерба для объекта:

$$S_{Risk} = [S_{Threat}, Damage], \quad (11)$$

где S_{Risk} – уровень критичности риска.

Ущерб (*Damage*) введён в онтологическую модель для обеспечения целостного и завершённого представления предметной области информационной безопасности систем Интернета вещей. Ущерб определяется как мера потенциального негативного воздействия от реализации угрозы. Для формализации ущерба присваивается свойство «уровень критичности», характеризующее процент тяжести негативных последствий для объекта в контексте его области функционирования. Контекстно-зависимый характер риска является критически важным требованием для гетерогенных сред Интернета вещей, где семантика одного и того же события варьируется в зависимости от сценария использования.

Ввиду отсутствия в открытых источниках готовых классификаций рисков для систем Интернета вещей, в работе проведен анализ авторитетных источников: отчеты ENISA [47], публикации IoT Security Foundation [48], база угроз Лаборатории Касперского [49], а также исследования [31, 40 – 44]. На основе этого анализа выделены классы рисков безопасности, релевантные для систем Интернета вещей (Рис. 7).

Модель представляет собой конечную точку пути от конкретной СФХ до итогового риска. В результате формируется не абстрактный перечень угроз, а множество последовательностей деструктивных воздействий, каждая из которых завершается негативным событием. Перечисленные риски характерны именно для систем Интернета вещей и отличаются от рисков, присущих чисто программным информационным системам, что критически важно при анализе защищённости гетерогенных систем Интернета вещей.

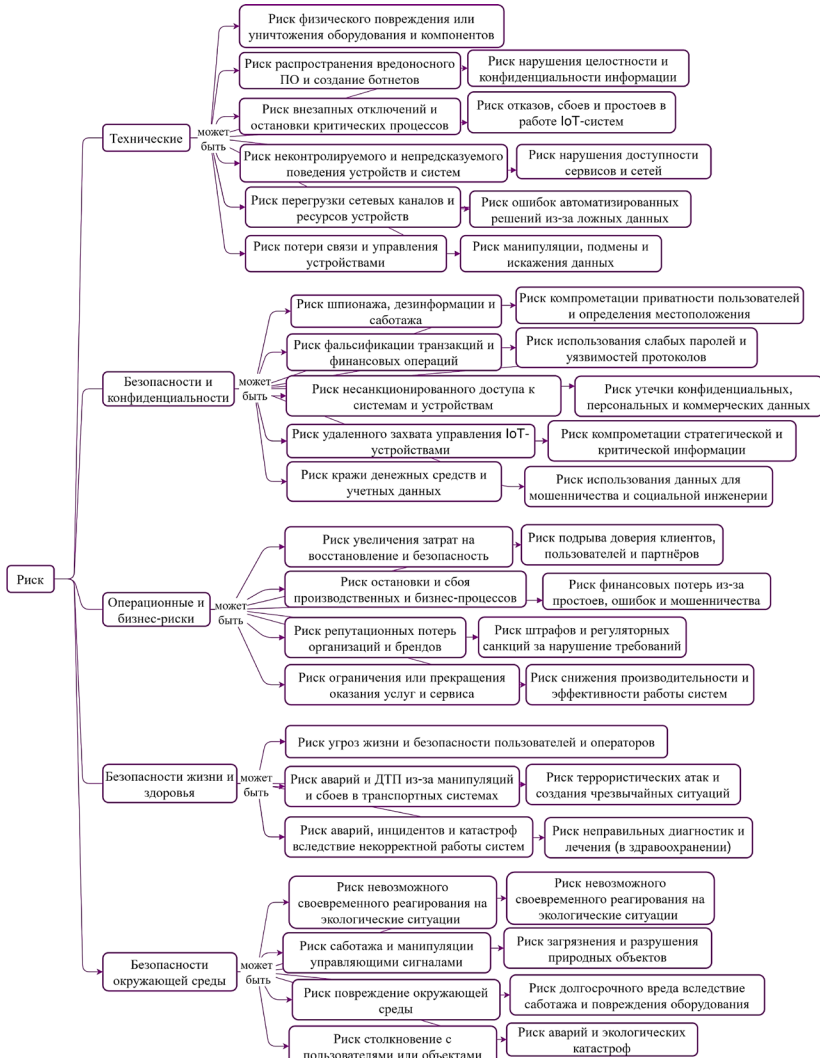


Рис. 7. Модель «Риска»

4.6. Реализация онтологической модели. Использование онтологического подхода обеспечивает не только структурирование информации, но и формализацию семантических связей между элементами предметной области. Это создаёт предпосылки для применения логического вывода и автоматизированных

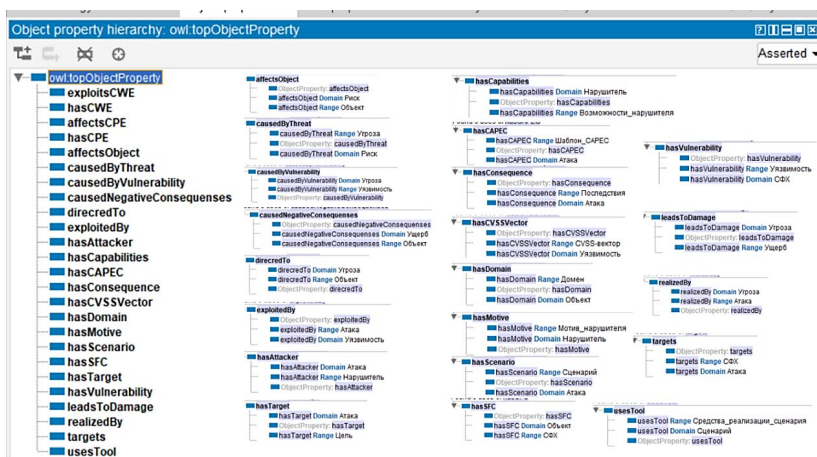


Рис. 9. Назначенные Object Properties

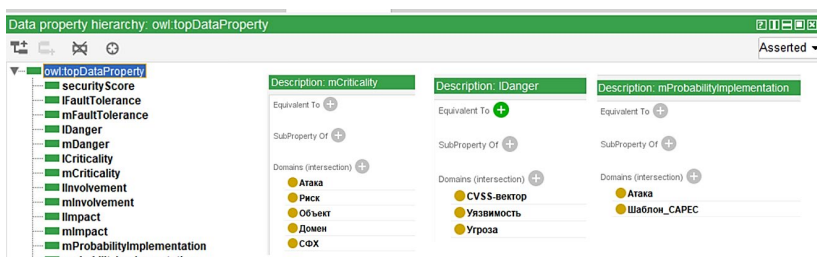


Рис. 10. Назначенные Data Properties

Для отражения ресурсных ограничений компонентов системы Интернета вещей введены дата свойства, которые также характеризуют и объект исследования. Например, для класса СФХ «Актuatorы» введены дата свойства `hasCurrentConsumptionMax`, отражающий напряжение питания, `hasSupplyVoltageMax`, отражающий потребляемый ток, и `hasControlType`, отражающий тип управления (Рис. 11). Данные свойства показывают, что могут использоваться только простые электрические приводы и реле.

Наполнение онтологии экземплярами классов производилось как вручную – с помощью вкладки «Individuals» редактора Protege, так и полуавтоматически – с использованием модуля «Cellfie» для импорта предварительно преобразованных данных. Для наполнения использовались внешние базы знаний: БДУ ФСТЭК, CVE, RVD, CWE, CPE (Common Platform Enumeration) [51], CAPEC, NVD [52]. Поскольку перечисленные источники используют различные

форматы данных, для их интеграции в единое непротиворечивое хранилище применяется метод ETL (Extract, Transform, Load). Ключевым этапом, обеспечивающим совместимость и целостность знаний, является Transform, в рамках которого выполняется установление соответствия (маппинг) исходных данных на единый формат, заданный онтологией. Единый формат описания классов (Рис. 12), выработанный на основе анализа всех используемых источников, гарантирует, что каждый элемент будет представлен в базе знаний с унифицированным набором атрибутов и связей. Это создаёт основу для последующего логического вывода.

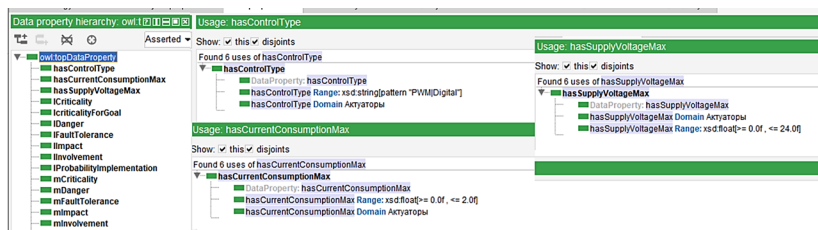


Рис. 11. Назначенные Data Properties для актуаторов

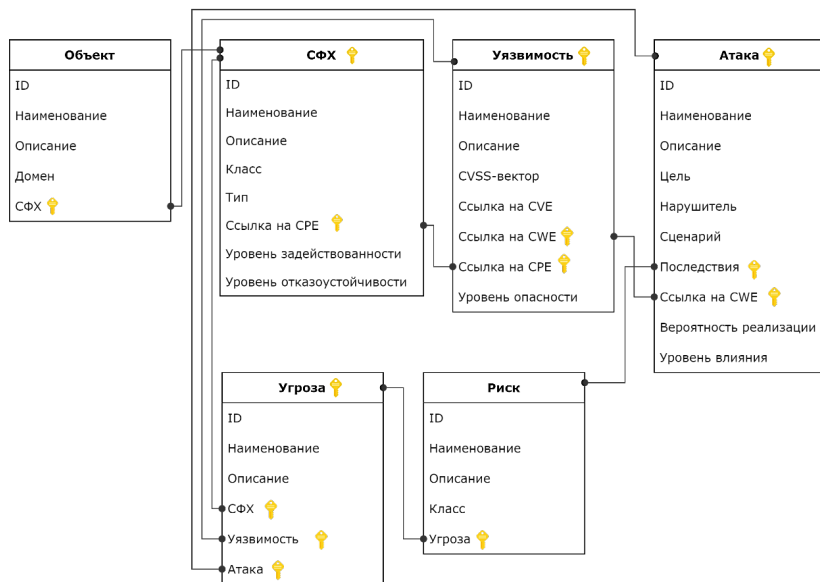


Рис. 12. Единый формат описания классов в базе знаний

Для отражения взаимодействия между СФХ и последующего вывода каскадных угроз созданы объектные отношения, описанные в разделе 4. Пример таких отношений для класса «Контроллеры, процессоры» показан на рисунке 13.

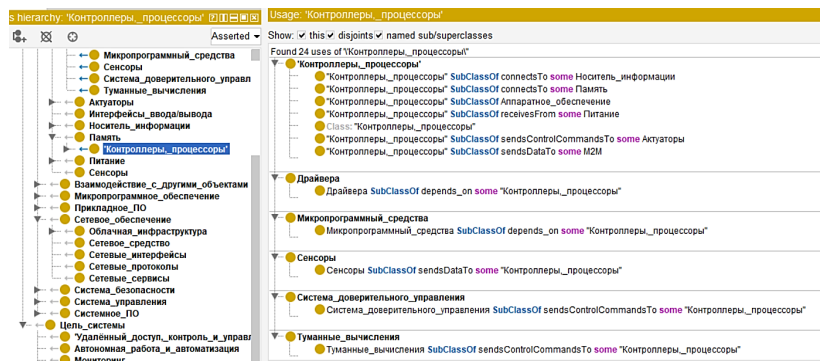


Рис. 13. Назначенные Object Properties для отражения взаимодействия между СФХ на примере класса «Контроллеры, процессоры»

Для демонстрации ETL-процесса ниже приведен пример интеграции записи о программном обеспечении датчика для мониторинга показателей электроэнергии Panasonic KW Watcher, который может быть частью системы Интернета вещей (таблица 2).

Для автоматического вывода связей между сущностями используется язык логических правил Semantic Web Rule Language (SWRL). Правила SWRL позволяют выразить логические зависимости между свойствами и классами онтологии. Правило состоит из двух частей: антецедента (условие) – конъюнкция атомарных утверждений (атомов), и консеквента (следствие) – также конъюнкция атомарных утверждений. Если все атомы в антецеденте истинны (выполняются), то все атомы в консеквенте также должны быть истинны. Логические правила для вывода информации о связанных с СФХ уязвимостях, атаках, угрозах и рисках для объекта приведены ниже.

Правило 1. Вывод связанных с СФХ уязвимостей: $IoT:hasCPE(?sfc, ?cpe) \wedge IoT:hasCPE(?vuln, ?cpe) \rightarrow IoT:hasVulnerability(?sfc, ?vuln)$ если СФХ имеет идентификатор уязвимого продукта (CPE), и уязвимость также влияет на этот же самый идентификатор продукта (CPE), то можно сделать вывод, что данная СФХ обладает этой уязвимостью.

Таблица 2. Процессы ETL для наполнения базы знаний

Этап	Действие	Пример для на примере СФХ Panasonic KW Watcher
Extract (Извлечение)	Идентификация и сбор сырых данных из приведённых источников	Извлечение записи для <code>сре:/a:panasonic:control_fwwin_pro:6.414</code> из базы данных CPE
Transform (Преобразование)	Нормализация и установление соответствий с базой знаний. Данные из каждого источника преобразуются к единому формату, определяемому классами и свойствами онтологии	Вход: Запись <code>сре:/a:panasonic:control_fwwin_pro:6.414</code> из CPE Процесс: 1. Создается новый экземпляр класса «СФХ» 2. Заполнение свойств: - Идентификатор = «CPE-ID» - Наименование = «Panasonic KW Watcher» - Описание = «Description» (текст из CPE) - Класс = «Прикладное ПО» - Тип = «Клиентское ПО» - Ссылка на CPE = <code>«сре:/a:panasonic:kw_watcher:-»</code> - Уровень задействованности = «0,2» - Уровень отказоустойчивости = «0,5» 3. Устанавливается связь <code>hasVulnerability</code> с соответствующими экземплярами класса
Load (Загрузка)	Сохранение в базу знаний. Преобразованные и связанные данные загружаются в базу знаний	Структурированный экземпляр уязвимости CVE-2023-3472 со всеми свойствами и связями сохраняется в базе знаний

Правило 2. Вывод связанных с уязвимостью атак: $IoT:hasCWE(?vuln, ?cwe) \wedge IoT:exploitsCWE(?attack, ?cwe) \rightarrow IoT:exploits(?attack, ?vuln)$ если уязвимость имеет идентификатор слабого места (CWE), и атака также имеет этот же самый идентификатор слабого места (CWE), то можно сделать вывод, что данная атака эксплуатирует данную уязвимость.

Правило 3. Вывод угроз: $IoT:hasSFC(?obj, ?sfc) \wedge IoT:hasVulnerability(?sfc, ?vuln) \wedge IoT:exploits(?attack, ?vuln) \rightarrow IoT:hasThreat(?obj, ?threat) \wedge IoT:comprises(?threat, ?sfc, ?vuln, ?attack)$ если объект содержит СФХ, И эта СФХ имеет уязвимость, и существует атака, которая эксплуатирует данную уязвимость, то можно сделать вывод, что для объекта возникает угроза, которая представляет собой

комбинацию уязвимого компонента, конкретной уязвимости и реализующей её атаки.

Правило 4. Вывод каскадных угроз через связи СФХ: $IoT:comprises(?threat, ?sfc1, ?vuln, ?attack) \wedge IoT:connectsTo(?sfc1, ?sfc2) \rightarrow IoT:comprises(?threat, ?sfc2, ?vuln, ?attack)$ если угроза уже связана с компонентом СФХ1, и этот компонент имеет связь connectsTo с компонентом СФХ2, то угроза распространяется и на СФХ2.

Правило 5. Вывод каскадного распространения угроз через потоки сообщений: $IoT:comprises(?threat, ?sfc1, ?vuln, ?attack) \wedge IoT:hasSrc(msg, ?sfc1) \wedge IoT:hasDst(msg, ?sfc2) \rightarrow IoT:threatensMessage(?threat, msg) \wedge IoT:comprises(?threat, ?sfc2, ?vuln, ?attack)$ если скомпрометирован СФХ, который является отправителем сообщения, то под угрозой оказывается само сообщение, и эта же угроза каскадно проецируется на СФХ-получатель.

Правило 6. Связь угрозы компонента с целью системы: $IoT:comprises(?threat, ?sfc, ?vuln, ?attack) \wedge IoT:realizedBySFC(?goal, ?sfc) \rightarrow IoT:threatensGoal(?threat, ?goal)$ если угроза ассоциирована с конкретной СФХ, и данная СФХ задействована в реализации целевой функции системы, то данная угроза качественно классифицируется как компрометирующая цель системы.

Правило 7. Вывод рисков: $IoT:hasSFC(?obj, ?sfc) \wedge IoT:hasThreatForComponent(?sfc, ?threat) \wedge IoT:hasProbabilistic(?threat, ?risk) \rightarrow IoT:hasRisk(?obj, ?risk)$ если объект содержит СФХ, и для СФХ существует угроза (связка из правила 3), и для этой угрозы есть связанный с ней риск, то можно сделать вывод, что для объекта существует данный риск.

Рассмотрим синтаксис на примере Правила 1: $IoT:hasCPE(?sfc, ?cpe) \wedge IoT:hasCPE(?vuln, ?cpe) \rightarrow IoT:hasVulnerability(?sfc, ?vuln)$. Здесь ?sfc, ?cpe, ?vuln – переменные, связывающие экземпляры. Часть до стрелки ($\rightarrow$) – антецедент (условие «если»), а после – консеквент (вывод «то»). Правило срабатывает для всех экземпляров онтологии, для которых истинен антецедент.

Выполнение правил инициируется запуском механизма логического вывода резонера, такого как Pellet, непосредственно в среде Protege. Резонер в автоматическом режиме проверяет аксиомы онтологии и правила, добавляя выведенные отношения между экземплярами, например, утверждение о наличии уязвимости у конкретной СФХ. Последовательность применения правил определяется самим резонером на основе имеющихся фактов, что и обеспечивает автоматизированный вывод.

Представленные правила формализуют цепочки логических выводов, позволяя на основе базовых фактов о конфигурации системы и общеизвестных баз знаний автоматизированно выводиться скрытые логические связи о потенциальных угрозах и рисках.

Разработанная в настоящей работе онтологическая модель отличается от существующих решений тем, что ориентирована не только на структурирование знаний, но и на их использование в процессе анализа защищённости систем Интернета вещей. Ключевой особенностью модели является явное представление архитектуры системы Интернета вещей через СФХ и выделение домена для определения контекста функционирования объекта, а также выделение рисков, специфичных для Интернета вещей, и установление формализованных связей между компонентами, уязвимостями, атаками, угрозами и рисками. Такой подход позволяет учитывать архитектурно-зависимый характер угроз и обеспечивает идентификацию только тех угроз, которые актуальны для конкретной конфигурации системы.

Важным свойством разработанной модели является возможность её расширения за счёт добавления новых классов, атрибутов и отношений без нарушения целостности уже сформированного представления знаний. Это позволяет адаптировать онтологию к различным предметным областям применения Интернета вещей, а также учитывать появление новых типов устройств, протоколов взаимодействия и угроз информационной безопасности. Кроме того, модульная структура онтологии упрощает её интеграцию с внешними источниками знаний и средствами анализа безопасности.

4.7. Место онтологической модели в процессе анализа защищённости. Разработанная онтологическая модель не является самостоятельным средством количественной оценки защищённости. Она выполняет роль семантического каркаса, который формализует и связывает в единую структуру следующие элементы:

- архитектуру системы Интернета вещей (через классы Object, SFC, Domain и отношения hasSFC, hasDomain);
- известные уязвимости (класс Vulnerability с данными из CVE, RVD);
- шаблоны компьютерных атак (класс Attack с данными из CAPEC);
- возникающие угрозы безопасности информации (класс Threat);
- потенциальный ущерб и риски (классы Damage, Risk).

Модель выводит связи между этими элементами с помощью SWRL-правил. Полученный в результате граф знаний, в котором каждый компонент системы (SFC) явно связан с актуальными для него

уязвимостями, а те, в свою очередь, – с конкретными атаками и итоговыми угрозами.

Этот граф знаний служит формализованным входом для последующих этапов анализа защищённости, в частности:

- количественной оценки защищённости: на основе выведенных связей могут быть рассчитаны вероятности реализации угроз (например, с использованием байесовских сетей или марковских процессов), а также интегральные показатели защищённости с учётом критичности компонентов;

- расчёта рисков: классы Risk и Damage в модели предоставляют необходимую структуру для применения стандартных методик оценки рисков (в том числе совместимых с подходами ФСТЭК России);

- обоснованного выбора контрмер: зная, какие угрозы связаны с какими компонентами, оператор может целенаправленно выбирать средства защиты, минимизируя затраты.

Важно подчеркнуть, что сами количественные расчёты (вычисление численных значений вероятностей, ущерба, интегральных показателей) выходят за рамки данной работы. Их реализация требует привлечения дополнительных математических методов и является отдельной задачей. Однако без предложенной онтологии любые такие расчёты будут опираться на неформализованные или неполные данные о связях между компонентами, уязвимостями и атаками.

Следует подчеркнуть, что представленные в разделах 4.1-4.5 математические выражения формируют теоретический и алгоритмический базис для аналитического программного модуля. В рамках настоящего этапа работы разработанная онтологическая модель решает критически важную задачу верхнего уровня – агрегацию гетерогенных данных и построение базы знаний, которая выступает в качестве структурированного источника входных данных для последующих автоматизированных количественных расчётов

Таким образом, модель не подменяет полный цикл анализа защищённости, а создаёт его фундамент – формализованное, воспроизводимое и расширяемое представление знаний об угрозах. Это позволяет отделить логическую идентификацию угроз (что делает онтология) от количественной оценки и принятия решений (что остаётся за рамками, но может быть надстроено на основе результатов работы модели).

5. Оценка качества разработанной онтологии. Для оценки качества разработанной онтологической модели применены метрики методологии OntoMod [53], включающие показатели комплексной структурной сложности (обобщенный показатель, охватывающий

связанность и сложность онтологии), инкапсуляции (степень скрытия внутренней структуры модулей онтологии при сохранении необходимых связей между ними), покрытия (полнота представления ключевых понятий предметной области) и глубины иерархии (уровень детализации классов). Онтология разделена на ключевые тематические модули: «Угроза», «Объект», «Уязвимость», «Атака» и «Риск». Исходные параметры для оценки (таблица 3) определялись на основе онтологических метрик из Protege и результатов SPARQL-запросов, которые включали сведения о классах и их модулях, объектных свойствах, отношениях подклассов и модульной структуре онтологии.

Для расчета комплексной структурной сложности применяется формула:

$$ComplexOnto\ score = 0,25 * (LD + LC + LR + CC_{norm}), \quad (12)$$

где *ComplexOnto score* – комплексная структурная сложность, $LD = \frac{2 \cdot (op+dp)}{n(n-1)}$ – плотность связей, где *op* – количество объектных свойств, *dp* – количество дата-свойств, *n* – общее число концептов в онтологии, $LC = \frac{op+dp}{n}$ – среднее количество связей на концепт, $LR = \frac{op+dp}{op+dp+sc}$ – насыщенность связей, где *sc* – количество подкласс-аксиом, $CC_{norm} = \frac{op+dp+sc-n+2}{n}$ – нормализованная цикломатическая сложность.

На основе данных из таблицы 3 рассчитаем комплексную структурную сложность по формуле (12). Результаты представлены в таблице 4.

Чтобы определить инкапсуляцию для всей онтологии необходимо определить вес (вклад) каждого модуля:

$$\omega_i = \frac{NCMO(m_i) + NRolesMO(m_i) + N_iRolesMO(m_i)}{\sum_{j=1}^k NCMO(m_j) + NRolesMO(m_j) + N_jRolesMO(m_i)}, \quad (13)$$

где ω_i – инкапсуляция каждого модуля, m_i – выделенный модуль онтологии, $NCMO(m_i)$ – количество концептов (классов) в модуле m_i , $NRolesMO(m_i)$ – количество связей (объектных свойств) в модуле m_i , $N_iRolesMO(m_i)$ – количество связей (объектных свойств) с другими модулями, $\sum_{j=1}^k NCMO(m_j) + NRolesMO(m_j) + N_jRolesMO(m_i)$ – общий размер онтологии с учётом всех её модулей.

Таблица 3. Количественные данные об онтологии

Метрика	Угроза	Объект	Уязвимость	Атака	Риск	Общая онтология
Количество концептов (n)	1	49	5	20	7	88
Связи внутри модуля	1	7	2	6	1	
Связи между модулями	10	7	3	3	2	
Объектные свойства (op)						35
Дата-свойства (dp)						27
Подкласс-аксиомы (sc)						111
Экземпляры						209

Таблица 4. Расчёт ComplexOnto score

Показатель	Расчёт
Плотность связей	$LD = (35 + 27)/(88 * (88 - 1)/2) = 0,02$
Среднее количество связей на концепт	$LC = (35 + 27)/88 = 0,7$
Насыщенность связей	$LR = (35 + 27)/(35 + 27 + 111) = 0,4$
Цикломатическая сложность	$CC_{norm} = (35 + 27 + 111 - 88 + 2)/88 = 0,9$
Комплексная структурная сложность	$ComplexOnto = 0,25 * (0,02 + 0,7 + 0,4 + 0,9) = 0,5$

Для расчёта общей инкапсуляции используем формулу:

$$knowEncap(O) = \sum_i \omega_i * knowEncap(m_i), \quad (14)$$

где $knowEncap(O)$ – общая инкапсуляция онтологии.

Для расчётов используем данные из таблицы 3. Результаты расчётов представлены в таблице 5.

Для расчета покрытия применяется формула:

$$Cov(O) = \alpha * ConCov * RelCov, \quad (15)$$

где $Cov(O)$ – покрытие онтологии, α – эмпирический коэффициент для масштабирования или коррекции оценки покрытия онтологии, $ConCov = \frac{1}{k} \sum_{j=1}^k \sum_{i=1}^n f(c_i, t_j)$ – покрытие концепций онтологии, где k – эмпирический коэффициент корректировки, который служит для учета различных факторов, $f(c_i, t_j)$ – функция соответствия

метки концепта c_i термину поиска t_j , $RelCov = \frac{2}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n g(c_i, c_j)$ – покрытие отношений онтологии, где n – число всех релевантных отношений (свойств, связей), которые характеризуют предметную область; $g(c_i, c_j)$ – функция наличия отношений между парами концепций c_i, c_j .

Таблица 5. Расчёт knowEncap

Показатель	Расчёт
ω_i для угрозы	$\omega_i = (1 + 1 + 10)/150 = 0,08$
ω_i для объекта	$\omega_i = (49 + 7 + 7)/150 = 0,42$
ω_i для уязвимости	$\omega_i = (5 + 2 + 3)/150 = 0,06$
ω_i для атаки	$\omega_i = (20 + 6 + 3)/150 = 0,2$
ω_i для риска	$\omega_i = (7 + 1 + 2)/150 = 0,06$
Общая инкапсуляция онтологии	$knowEncap(O) = 0,56$

Для оценки покрытия онтологии необходимо установить соответствие между метками концептов и поисковыми запросами. Под меткой каждого концепта понимается его собственное наименование. В качестве поисковых терминов задействован следующий перечень: «IoT», «Интернет вещей», «Объект», «Уязвимость», «Угроза», «Атака», «Риск», «Ущерб», «Сенсор», «Актuator», «Контроллер». Функция $f(c_i, t_j)$ для $ConCov$ представляется в виде матрицы для определения соответствия метки концепта термину поиска. Матрица заполняется по правилу $f(c_i, t_j) = \begin{cases} 1, & \text{если метка концепта } c_i \text{ совпадает с термином } t_i \\ 0, & \text{иначе} \end{cases}$. Учтем,

что термин «IoT» совпадает с меткой «Объект». В качестве коэффициента k примем количество выделенных терминов. Ввиду значительного объема полученных данных матрица не приводится в тексте статьи. Функцией $g(c_i, c_j)$ для $RelCov$ является количество объектных свойств в онтологии в соответствии с таблицей 3. Для расчёта $Cov(O)$ коэффициент α вычисляется как соотношение числа экземпляров и концептов, что отражает как структурную полноту, так и качество наполнения онтологии экземплярами.

Результаты расчета покрытия по формуле (15) приведены в таблице 6.

Глубина иерархии определяется по формуле:

$$DIT = \max(\sum D|c_i|), \quad (16)$$

где DIT – глубина иерархии онтологии, $D|c_i|$ – длина пути от корневого класса Thing до листового концепта c_i .

Таблица 6. Расчёт $Cov(O)$

Показатель	Расчёт
Покрытие концепций	$k = 11, f(c_i, t_j) = 6$ $ConCov = 6/11 = 0,5$
Покрытие связей	$n = 10, g(c_i, c_j) = 35$ $RelCov = (2 * 35)/(10 * 9) = 0,7$
Покрытие онтологии	$\alpha = 209/88 = 2,4$ $Cov(O) = 2,4 * 0,5 * 0,7 = 0,84$

В результате получим $DIT = 3$, так как максимальное количество ребер по иерархии классов во всех случаях равно 3.

Итоговым показателем качества онтологии служит *Behavioral Quality*. Это оценка онтологии с точки зрения её богатства и иерархической структуры, которая определяется по формуле:

$$Behavioral\ Quality = \omega_1 * knowEncap + \omega_2 * Cov + \omega_3 * DIT, \quad (17)$$

где *Behavioral Quality* – поведенческое качество онтологии, ω_i – веса, определяющие важность каждого аспекта (по умолчанию могут быть равными).

Так как ранее все параметры были вычислены, то получим $Behavioral\ Quality = 0,3 * 0,56 + 0,5 * 0,84 + 0,2 * 3 = 1,2$.

Для сравнения взяты онтологии из работ [15, 19, 21]. Для обеспечения объективности сравнительного анализа все метрики были рассчитаны по единой методологии OntoMod на основе анализа данных из [19] и OWL-файлов онтологий, полученных из публичных репозиториях, указанных в [21, 15]. Для каждой из сравниваемых и разработанной онтологий был произведен подсчет количества классов, объектных свойств, свойств-данных и аксиом подклассов, после чего по формулам (12-17) были вычислены итоговые показатели. Таким образом, все онтологии были оценены в одинаковых условиях на основе их формальных описаний безотносительно к конкретным наборам экземпляров. Результаты оценки представлены в таблице 7.

Проведённый сравнительный анализ позволяет сделать следующие выводы. Разработанная онтология обладает наилучшей структурной организацией среди всех аналогов, что выражается в снижении комплексной структурной сложности в 1,5-2,4 раза. По полноте покрытия предметной области разработанная онтология

превосходит рассмотренные решения в 3,36 раза, что является наиболее существенным преимуществом. Глубина иерархии находится на максимальном уровне среди сравнимых онтологий. Несколько более низкое значение инкапсуляции (снижение в 1,8 раза) является следствием модульной архитектуры и рассматривается не как недостаток, а как архитектурный компромисс, обеспечивающий гибкость.

Таблица 7. Расчёт метрик онтологических моделей
для сравнительного анализа

Показатель	SOtoTS [15]	Security of heathcare [19]	WoT [21]	Разработанная онтология
Плотность связей (<i>LD</i>)	0,5	0,07	0,33	0,02
Среднее количество связей на концепт (<i>LC</i>)	2	1,07	1,5	0,7
Насыщенность связей (<i>LR</i>)	1	0,44	0,63	0,4
Цикломатическая сложность (<i>CC_{norm}</i>)	1,2	1,47	1,6	0,9
Комплексная структурная сложность (<i>ComplexOnto</i>)	1,2	0,76	1,015	0,5
Инкапсуляция знаний (<i>knowEncap</i>)	1	1	1	0,56
Покрывтие онтологии (<i>Cov</i>)	0,25	0,25	0,25	0,84
Глубина онтологии (<i>DIT</i>)	1	3	2	3
Поведенческое качество (<i>Behavioral Quality</i>)	0,74	1,4	1,07	1,2

Таким образом, с точки зрения практического применения разработанная онтология обеспечивает: высокую сопровождаемость и лёгкость расширения (благодаря минимальной структурной сложности); максимальную полноту покрытия предметной области (благодаря высокому значению покрытия); хорошую выразительную силу (благодаря максимальной глубине иерархии); готовность к использованию в модульных приложениях (благодаря модульной архитектуре).

Рекомендуемым направлением дальнейшего улучшения является повышение инкапсуляции знаний, в частности, оптимизация интерфейсных связей модуля «Угроза», что позволит дополнительно повысить интегральное поведенческое качество.

6. Экспериментальный анализ результатов. Ключевым преимуществом разработанной онтологической модели является возможность автоматизированного вывода скрытых логических связей, отражающих знания об угрозах для конкретной системы Интернета вещей. В отличие от существующих подходов, где идентификация угроз выполняется вручную экспертом или основывается на статических списках,

предложенная модель позволяет формализовать причинно-следственные связи между архитектурой системы и потенциальными угрозами.

Для проведения эксперимента использована натурная модель «Умная лаборатория» – программно-аппаратный комплекс на базе концепции Интернета вещей, разработанный в Южном федеральном университете (Рис. 14). В состав натурной модели, функционирующей в домене «Наука», входят следующие компоненты, реализующие типовые СФХ системы Интернета вещей, которые представлены в таблице 8.

Репрезентативность стенда обусловлена тем, что он включает три ключевых уровня архитектуры Интернета вещей (аппаратный, сетевой, программный), использует протоколы передачи данных и физические устройства. Это позволяет проследить полную цепочку реализации угрозы: от атаки до физического воздействия (например, изменений рабочих функций одноплатного компьютера), что полностью соответствует киберфизической природе систем, для анализа которых предназначена модель.

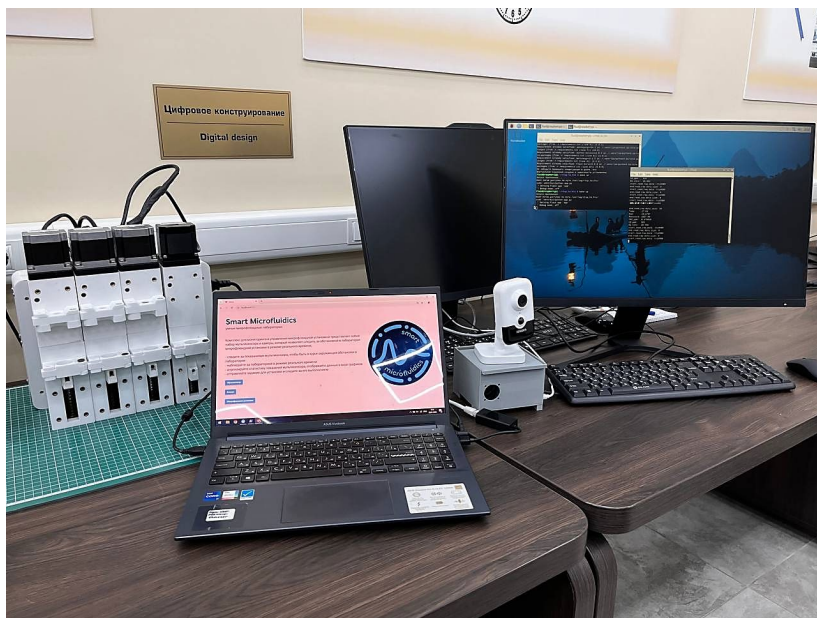


Рис. 14. Натурная модель «Умной лаборатории»

Таблица 8. Компоненты натурной модели «Умная лаборатория»

Класс СФХ	Тип СФХ	Реализация в стенде
Аппаратное обеспечение	Сенсор	Сенсор углекислого газа MH-Z19
		Сенсор угарного газа MQ-7
		Сенсор влажности, состава атмосферы, температуры BME-280
		Сенсор концентрация кислорода AO-02
	Контроллеры, процессоры	Микроконтроллер ESP32
	Интерфейсы ввода/вывода	Одноплатный компьютер Raspberry Pi
Сетевое обеспечение	Сетевые протоколы	IP-камера
		Протокол MQTT
		Протокол Modbus
		Протокол RTSP
Прикладное ПО	Серверное ПО	Rabbit MQ
Взаимодействие с другими объектами	Системы взаимодействия с пользователем	PostgreSQL
		Веб-приложение

В онтологию были загружены экземпляры классов, соответствующие СФХ лаборатории. Процесс идентификации угроз был автоматизирован с помощью набора SWRL-правил, описанных в разделе 4.6. Правила последовательно выводят связи между сущностями, наполняя базу знаний новыми фактам. На рисунке 15 представлен фрагмент графа онтологии, визуализирующий выведенные связи в Protege. Он наглядно демонстрирует, как онтология связывает конкретный компонент системы (микроконтроллер ESP32) с уязвимостями, векторами атак, итоговыми угрозами и рисками. Также на схеме показаны каскадные эффекты от угроз, переходящие на компоненты, связанные с микроконтроллером.

На представленной схеме стрелкой 1 обозначена взаимосвязь между микроконтроллером и уязвимостями; стрелкой 2 – взаимосвязь между уязвимостями и атаками; стрелкой 3 – взаимосвязь между атаками и угрозами. Группа стрелок 4 иллюстрирует каскадное распространение угроз посредством связей СФХ, а стрелки 5 – каскадное распространение угроз через потоки сообщений. Связи между угрозами и целевой системой (целью системы) на данной схеме не показаны. Стрелкой 7 отражена связь между рисками и угрозами.

На рисунке 16 представлены количественные результаты автоматического логического вывода, выполненного резонером Pellet на полном наборе SWRL-правил (Правила 1-3, 7) и экземпляров, описывающих стенд «Умная лаборатория». Правила были запущены последовательно в одном сеансе вывода. По оси ординат указано

абсолютное количество выведенных экземпляров соответствующих классов и свойств.

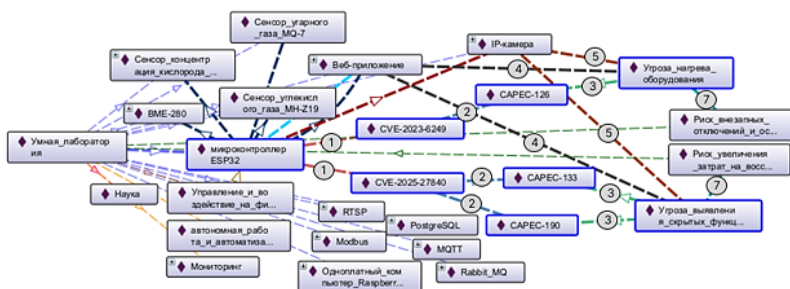


Рис. 15. Фрагмент графа онтологии, демонстрирующий выведенные связи «СФХ-Уязвимость-Атака-Угроза»

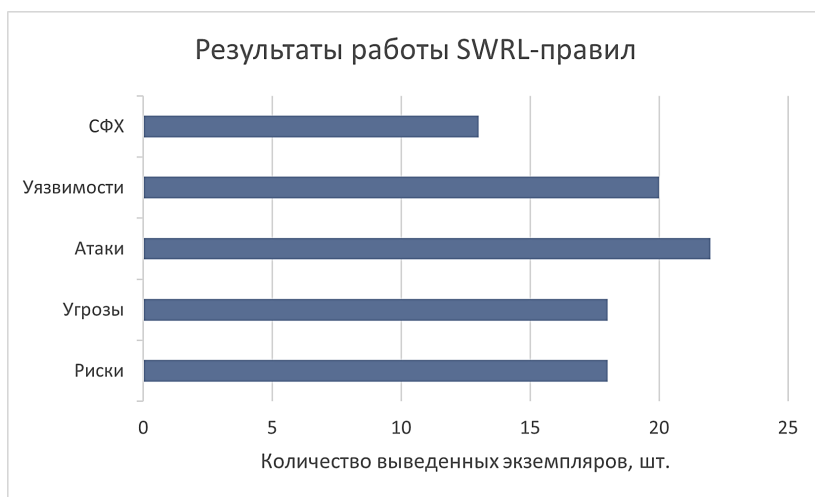


Рис. 16. Результаты работы SWRL-правил

Для иллюстрации процесса вывода (трассировки) рассмотрим получение одной из угроз. Исходный факт: экземпляр микроконтроллер ESP32 (СФХ) связан свойством hasCPE с cpe:/h:espressif:esp32:-. Резонер, проверив Правило 1, находит экземпляр CVE-2025-27840 (Уязвимость) с тем же CPE и генерирует новый факт: микроконтроллер ESP32 hasVulnerability CVE-2025-27840. Далее, согласно Правилу 2, у уязвимости CVE-2025-27840 есть CWE-идентификатор CWE-912,

и у атаки CAPEC-190 есть такой же CWE-идентификатор. Соответственно создается связь CAPEC-190 exploits CVE-2025-27840. Наконец, Правило 3, имея все три факта, создает экземпляр класса Threat, связывая его с объектом натурной модели hasThreat Угроза выявления скрытых функций, а также связывая вместе уязвимый компонент, атаку и угрозу.

Для оценки полноты идентификации угроз был проведён сравнительный эксперимент. В качестве инструмента «ручного» (регламентированного) построения исходного перечня угроз использовался автоматизированный конструктор угроз на сайте Банка данных угроз ФСТЭК России [54], который формирует агрегированный перечень угроз для заданной архитектуры системы.

Для натурной модели «Умная лаборатория» конструктор выделил 2 системных сценария: «Фальсификация данных эксперимента» и «Компрометация ядра системы». Разработанная онтология с использованием SWRL-правил автоматически выявила 18 конкретных угроз, каждая из которых привязана к конкретным компонентам (СФХ), уязвимостям (CVE/RVD) и шаблонам атак (CAPEC). Все 2 сценария, сформированные конструктором ФСТЭК, полностью покрываются набором выведенных онтологий угроз.

Таким образом, онтология обеспечивает как минимум ту же полноту на агрегированном уровне, но дополнительно предоставляет детализированное техническое описание угроз, что невозможно получить напрямую из конструктора ФСТЭК. Это позволяет автоматизировать последующий этап анализа – выбор контрмер и оценку рисков.

7. Заключение. Проведенное исследование направлено на решение ключевого противоречия в области безопасности Интернета вещей: между стремительным ростом числа угроз, обусловленным гетерогенностью и усложнением архитектур, и потребностью операторов в объективном анализе защищённости таких систем. Основной научный результат работы заключается в разработке онтологической модели, которая не просто объединяет предметные области Интернета вещей и информационной безопасности, но и впервые предлагает формальный механизм выявления каскадного распространения угроз, учитывающий не только топологическую связность компонентов, но и семантику передаваемых сообщений, а также функциональную роль каждого элемента в достижении целевых функций системы. Ключевое отличие предложенного подхода от аналогов состоит в том, что глубина и последствия угрозы определяются не фактом первичной компрометации, а структурно-семантическими характеристиками

межкомпонентных связей, что позволяет выявлять скрытые пути нарушения работы системы. Практическая значимость работы раскрывается через возможность автоматизации ключевого этапа анализа защищённости – идентификации угроз. Разработанная онтология совместно с SWRL-правилами позволяет получить исчерпывающий и воспроизводимый перечень конкретных угроз, привязанных к реальным компонентам системы, уязвимостям и шаблонам атак. Апробация на экспериментальном стенде «Умная лаборатория» продемонстрировала, что модель выявляет детализированные технические угрозы, привязанные к конкретным компонентам, уязвимостям и шаблонам атак, обеспечивая полноту не ниже агрегированных сценариев из БДУ ФСТЭК. Практическая применимость модели не ограничивается лабораторным стендом. Её модульная архитектура обеспечивает масштабирование на реальные системы за счёт интеграции с авторитетными базами знаний (CVE, CAPEC), которые непрерывно пополняются. Оценка качества по методологии OntoMod показала расширение семантического покрытия предметной области в 3 раза и снижение структурной сложности в 1,5-2,4 раза по сравнению с рассмотренными аналогами, что свидетельствует о высокой сопровождаемости и пригодности модели для практического использования операторами и проектировщиками распределённых сетей Интернета вещей.

Вместе с тем исследование имеет ряд ограничений. Во-первых, ограничения модели связаны с границами её онтологического базиса: идентифицируются только те угрозы, которые реализуются через известные и задокументированные уязвимости (CVE/RVD) и шаблоны атак (CAPEC). Угрозы класса «нулевого дня» и атаки, специфичные для уникальных аппаратных решений, не имеющих CPE-идентификаторов, остаются за рамками автоматического анализа. Важно отметить, что ограничения на типы охватываемых систем и угроз задаются не самой моделью, а составом эталонного классификатора СФХ. Разработанный классификатор покрывает широкий спектр типовых компонентов (сенсоры, контроллеры, ПО, протоколы), что позволяет описывать большинство распространённых архитектур. Для интеграции узкоспециализированной системы достаточно расширить таксономию СФХ, не изменяя логические правила, заложенные в онтологию. Во-вторых, область исследования ограничена целенаправленными атаками нарушителя, исключая случайные события, ошибки и отказы, что, хотя и обосновано спецификой Интернета вещей, может не охватывать все сценарии нарушения функционирования. В-третьих, актуальность выявленных угроз напрямую зависит

от полноты и своевременности обновления внешних баз знаний, что накладывает требования к эксплуатационному сопровождению модели. Несмотря на указанные ограничения, разработанная онтологическая модель представляет собой законченное и практически применимое решение, создающее формализованный фундамент для последующего этапа количественного анализа защищённости и обоснованного выбора контрмер в системах Интернета вещей.

Литература

1. Doynikova E., Novikova E., Murenin I., et. al. Security Measuring System for IoT Devices // Lecture Notes in Computer Science. Springer. 2022. vol. 13106. pp. 256–275.
2. ISO/IEC 30141:2024. Internet of Things (IoT) – Reference architecture: international standard // Geneva: ISO. 2024. 61 p.
3. Parsons E., Panaousis E., Loukas G., Sakellari G. Survey on Cyber Risk Management for the Internet of Things // Applied Sciences. 2023. vol. 13. no. 15. pp. 1–44.
4. Salayma M. Risk and threat mitigation techniques in internet of things (IoT) environments: a survey // Frontiers in the Internet of Things. 2023. vol. 2. pp. 1–22.
5. Dione D., Seck B., Diop I., et. al. Hardware Security for IoT in the Quantum Era: Survey and Challenges // Journal of Information Security. 2023. vol. 14. pp. 227–249.
6. ГОСТ Р 70924-2023. Информационные технологии. Интернет вещей. Типовая архитектура. Издание официальное // М.: Российский институт стандартизации. 2024.
7. Gupta H., Bharti A. Fog Computing & IoT: Overview, Architecture and Applications // International Journal of Advanced Research in Computer and Communication Engineering. 2018. vol. 7. pp. 30–34.
8. Zahoor S., Sofi S. Internet of Things: A Review of Functionality, Applications, Architectures and Challenges // Journal of the University of Shanghai for Science and Technology. 2021. vol. 23. no. 9. pp. 1216–1224.
9. Herzog A., Shahmehri N., Duma C. An Ontology of Information Security // International Journal of Information Security and Privacy. 2007. vol. 1. no. 4. pp. 1–23.
10. Moller D.P.F. Cybersecurity Ontology // Cybersecurity in Digital Transformation. Springer. 2020. pp. 79–96.
11. Lee J., Moon D., Kim I., Lee Y. A semantic approach to improving machine readability of a large-scale attack graph // The Journal of Supercomputing. 2019. vol. 75. pp. 3028–3045.
12. Falodiya K., Das M.L. Security Vulnerability Analysis using Ontology-based Attack Graphs // Proceedings of the IEEE India Council International Conference. 2017. pp. 1–5.
13. Saint-Hilaire K., Neal C., Cuppens F., et. al. A real-time automated attack-defense graph generation approach // Journal of Information Security and Applications. 2025. vol. 94. pp. 1–22.
14. Qaswar F., Rahmah M., Raza M., et. al. Applications of Ontology in the Internet of Things: A Systematic Analysis // Electronics. 2023. vol. 12. no. 1. pp. 1–30.
15. Jarwar M., Watson J., Ani U., Chalmers S. Industrial Internet of Things Security Modelling using Ontological Methods // Proceedings of the ACM Workshop on Secure and Trustworthy Cyber-Physical Systems. 2022. pp. 163–170.
16. Bhole M., Kastner W., Sauter T. Towards Unveiling Vulnerabilities and Securing IoT Devices: An Ontology-Based Approach // Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation. 2024. pp. 1–8.
17. Lavrova D.S., Vasilev Y.S. An ontological model of the domain of applications for the Internet of Things in analyzing information security // Automatic Control and Computer Sciences. 2017. vol. 51. no. 8. pp. 817–823.

18. Zegzhda D.P., Lavrova D.S. Approach to Internet of Things detection of security incidents using SIEM technology // Интеллектуальные технологии на транспорте. 2017. №1. pp. 35–41.
19. Nazir A., Sholla S., Bashir A. An Ontology based Approach for Context-Aware Security in the Internet of Things (IoT) // International Journal of Wireless and Microwave Technologies (IJWMT). 2021. vol. 11. no. 1. pp. 28–46.
20. SAREF: Smart Applications REference ontology: v. 3.1.1 // European Telecommunications Standards Institute. URL: <https://saref.etsi.org/core/v3.1.1/> (дата обращения: 05.02.2026).
21. Web of Things (WoT) Security Ontology // World Wide Web Consortium (W3C). URL: <https://www.w3.org/2019/wot/security> (дата обращения: 05.02.2026).
22. Федорченко Е.В., Новикова Е.С., Котенко И.В., Гайфулина Д.А., Тушканова О.Н. Левшун Д.С., Мелешко А.В., Муренин И.Н., Коломеец М.В. Система измерения защищенности информации и персональных данных для устройств интернета вещей // Вопросы кибербезопасности. 2022. №5. С. 28–46.
23. Kotenko I., Fedorchenko A., Doynikova E., Chechulin A. An Ontology-based Storage of Security Information // Information Technology and Control. 2018. vol. 47. no. 4. pp. 1–13.
24. Fedorchenko E.V., Kryukov R.O., Kotenko I.V., et. al. Security assessment based on attack graphs using NVD and MITRE ATT & CK database for heterogeneous infrastructures // Information and Control Systems. 2024. no. 2(129). pp. 39–50.
25. Makhdoom I., Abolhasan M., et. al. Anatomy of Threats to the Internet of Things // IEEE Communications Surveys & Tutorials. 2019. vol. 21. no. 2. pp. 1636–1675.
26. Yaacoub J.P.A., Noura H.N., Salman O., Chehab A. Robotics cyber security: vulnerabilities, attacks, countermeasures, and recommendations // International Journal of Information Security. 2022. vol. 21. pp. 115–158.
27. Mishura M. CyberSecOnto-IoT: онтология кибербезопасности Интернета вещей // GitHub. URL: <https://github.com/MashaMishura/CyberSecOnto-IoT.git> (дата обращения 05.02.2026).
28. Приказ Министерства цифрового развития, связи и массовых коммуникаций РФ от 29 марта 2019 г. №113 «Об утверждении Концепции построения и развития узкополосных беспроводных сетей связи «Интернета вещей» на территории Российской Федерации». URL: <https://docs.cntd.ru/document/554066760> (дата обращения 05.02.2026).
29. ISO/IEC 27400:2022. Cybersecurity – IoT security and privacy – Guidelines: international standard // Geneva: ISO. 2022.
30. NISTIR 8228. Considerations for Managing Internet of Things (IoT) Cybersecurity and Privacy Risks // NIST. 2019.
31. NISTIR 8259A. IoT Device Cybersecurity Capability Core Baseline / National Institute of Standards and Technology // NIST. 2020.
32. Федеральный закон Российской Федерации от 26 июля 2017 г. №187 «О безопасности критической информационной инфраструктуры Российской Федерации». <https://docs.cntd.ru/document/436752114> (дата обращения 05.02.2026).
33. ETSI TS 103 264 V3.1.1 (2020-02). SmartM2M; Smart Applications; Reference Ontology and oneM2M Mapping: Technical Specification // European Telecommunications Standards Institute. Sophia Antipolis: ETSI. 2020. 78 p.
34. CVE: Common Vulnerabilities and Exposures // The MITRE Corporation. URL: <https://cve.mitre.org/> (дата обращения: 05.02.2026).
35. RVD: Robotics Vulnerability Database // Alias Robotics. URL: <https://github.com/aliasrobotics/RVD> (дата обращения: 05.02.2026).
36. CVSS v3.0 specification document // FIRST. URL: <https://www.first.org/cvss/specification-document> (дата обращения: 05.02.2026).

37. Методика оценки уровня критичности уязвимостей программных, программно-аппаратных средств: методический документ: утвержден ФСТЭК России 30 июня 2025 г. URL: <https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-30-iyunya-2025-g> (дата обращения: 05.02.2026).
38. CAPEC – Common Attack Pattern Enumeration and Classification // The MITRE Corporation. URL: <https://capec.mitre.org/> (дата обращения: 05.02.2026).
39. Методика оценки угроз безопасности информации: методический документ: утвержден ФСТЭК России 5 февраля 2021 г. URL: <https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-5-fevralya-2021-g> (дата обращения: 05.02.2026).
40. Wu C.K. Architectures and security measures // Internet of Things security. Springer. 2021. pp. 13–26.
41. Dubey A., Yadav S. Basics of Internet of Things // International Journal of Scientific Research in Engineering and Management (IJSREM). 2024. vol. 8. no. 10. pp. 1–6.
42. Blinowski G., Piotrowski P. CVE based classification of vulnerable IoT systems // arXiv preprint arXiv:2006.16640. 2020.
43. Afroze M., Reena J. A Survey Study on IoT Application and its Attacks // International Research Journal on Advanced Engineering Hub (IRJAEH). vol. 2. no. 02. pp. 80–86.
44. Alam M., Khan I. Cyber Physical Attacks and IoT // Intelligent Cyber-Physical Systems Security for Industry 4.0. 2022. pp. 79–104.
45. Банк данных угроз безопасности информации. URL: <https://bdu.fstec.ru/threat> (дата обращения: 07.02.2026).
46. Дойникова Е.В., Котенко И.В. Оценивание защищённости и выбор контрмер для управления кибербезопасностью // М.: Российская академия наук, 2021. 184 с.
47. ENISA Threat Landscape 2024 // European Union Agency for Cybersecurity. Athens: ENISA. 2024. 112 p.
48. Best Practice Guidelines // IoT Security Foundation. URL: <https://iotsecurityfoundation.org/best-practice-guidelines/> (дата обращения: 07.02.2026).
49. Kaspersky IoT Threat Data Feed. URL: <https://www.kaspersky.ru/iot-threat-data-feed> (дата обращения: 07.02.2026).
50. Protege. URL: <http://protege.stanford.edu/> (дата обращения: 07.02.2026).
51. Common Platform Enumeration (CPE) // National Institute of Standards and Technology. URL: <https://nvd.nist.gov/products/cpe> (дата обращения: 07.02.2026).
52. National Vulnerability Database (NVD) // National Institute of Standards and Technology. URL: <https://nvd.nist.gov> (дата обращения: 07.02.2026).
53. Kumar S., Baliyan N. Semantic Web-Based Systems // SpringerBriefs in Computer Science. Springer. 2018. pp. 19–50.
54. Банк данных угроз ФСТЭК России. Формирование перечня угроз. URL: <https://bdu.fstec.ru/threat-section/shaper-threats> (дата обращения: 26.05.2026).

Фирсова Мария Геннадьевна — старший преподаватель, кафедры безопасности информационных технологий им. О.Б. Макаревича института компьютерных технологий и информационной безопасности, Федеральное государственное автономное образовательное учреждение высшего образования «Южный федеральный университет» (ФГАОУ ВО «ЮФУ»). Область научных интересов: анализ информационной безопасности киберфизических систем и систем Интернета вещей, онтологический инжиниринг. Число научных публикаций — 25. mshulika@sfedu.ru; ул. Чехова, 2, 347922, Таганрог, Россия; р.т.: +7(951)502-1484.

Поддержка исследований. Работа выполнена при поддержке Фонда поддержки молодых ученых имени Геннадия Комиссарова в рамках конкурса на получение стипендии имени Евгения Касперского.

M. FIRSOVA

ONTOLOGY MODEL FOR SECURITY ANALYSIS OF INTERNET OF THINGS SYSTEMS

Firsova M. Ontology model for security analysis of Internet of Things systems.

Abstract. The article addresses the problem of comprehensive security analysis of Internet of Things systems. The relevance of the work is driven by the growing complexity and heterogeneity of Internet of Things systems, as well as the need for scalable assessment of their security while taking into account architectural and functional characteristics. Based on an analysis of the architecture of Internet of Things systems, a multi-level ontological model has been developed that integrates the domains of the Internet of Things and information security, supplemented by consideration of the functional role of system components, the types of transmitted messages, and the goals of the system, which makes it possible to represent the dependence of a component's criticality on the function it performs and the influence of a threat on disruption of the system's target function. Based on the model, a threat inference mechanism is proposed that identifies the cascading effect of the propagation of destructive impact along the topological vector and the functional vector. In contrast to approaches that consider a threat exclusively through the fact of primary compromise of an element, the novelty lies in the fact that the depth and consequences of a threat are determined not by the initial compromise, but by the structural-semantic characteristics of inter-component connections that determine the development of the cascading effect. The ontology is integrated with external knowledge repositories (CVE, RVD, CAPEC, CWE, FSTEC BDU) through an ETL process, and the inference of hidden connections is automated using SWRL rules in the Protege environment. Practical validation of the model was carried out on the experimental testbed "Smart Laboratory." The result of the work is an implemented ontological model that provides automatic identification of security threats for Internet of Things systems. Evaluation of the model's quality using the OntoMod methodology showed a threefold expansion of the semantic coverage of the domain and a reduction in structural complexity by 1.5-2.4 times compared with the considered analogues. A conclusion is formulated about the applicability of the proposed ontological approach for prompt decision support for operators and designers of distributed Internet of Things networks.

Keywords: ontology, Internet of Things, threat, structural and functional characteristics, information security.

References

1. Doynikova E., Novikova E., Murenin I., et al. Security Measuring System for IoT Devices. Lecture Notes in Computer Science. 2022. vol. 13106. pp. 256–275.
2. ISO/IEC 30141:2024. Internet of Things (IoT) – Reference architecture: international standard. Geneva: ISO. 2024. 61 p.
3. Parsons E., Panaousis E., Loukas G., Sakellari G. Survey on Cyber Risk Management for the Internet of Things. Applied Sciences. 2023. vol. 13. no. 15. pp. 1–44.
4. Salayma M. Risk and threat mitigation techniques in internet of things (IoT) environments: a survey. Frontiers in the Internet of Things. 2023. vol. 2. pp. 1–22.
5. Dione D., Seck B., Diop I., et al. Hardware Security for IoT in the Quantum Era: Survey and Challenges. Journal of Information Security. 2023. vol. 14. pp. 227–249.
6. GOST R 70924-2023. Informatsionnye tekhnologii. Internet veshchei. Tipovaya arkhitektura. Izdanie ofitsialnoe [Information Technology. Internet of Things. Reference Architecture. Official Edition]. M.: Rossiiskii institut standartizatsii. 2024. (In Russ.).

7. Gupta H., Bharti A. Fog Computing & IoT: Overview, Architecture and Applications. International Journal of Advanced Research in Computer and Communication Engineering. 2018. vol. 7. pp. 30–34.
8. Zahoor S., Sofi S. Internet of Things: A Review of Functionality, Applications, Architectures and Challenges. Journal of the University of Shanghai for Science and Technology. 2021. vol. 23. no. 9. pp. 1216–1224.
9. Herzog A., Shahmehri N., Duma C. An Ontology of Information Security. International Journal of Information Security and Privacy. 2007. vol. 1. no. 4. pp. 1–23.
10. Moller D.P.F. Cybersecurity Ontology. Cybersecurity in Digital Transformation. Springer. 2020. pp. 79–96.
11. Lee J., Moon D., Kim I., Lee Y. A semantic approach to improving machine readability of a large-scale attack graph. The Journal of Supercomputing. 2019. vol. 75. pp. 3028–3045.
12. Falodiya K., Das M.L. Security Vulnerability Analysis using Ontology-based Attack Graphs. Proceedings of the IEEE India Council International Conference. 2017. pp. 1–5.
13. Saint-Hilaire K., Neal C., Cuppens F., et al. A real-time automated attack-defense graph generation approach. Journal of Information Security and Applications. 2025. vol. 94. pp. 1–22.
14. Qaswar F., Rahmah M., Raza M., et al. Applications of Ontology in the Internet of Things: A Systematic Analysis. Electronics. 2023. vol. 12. no. 1. pp. 1–30.
15. Jarwar M., Watson J., Ani U., Chalmers S. Industrial Internet of Things Security Modelling using Ontological Methods. Proceedings of the ACM Workshop on Secure and Trustworthy Cyber-Physical Systems. 2022. pp. 163–170.
16. Bhole M., Kastner W., Sauter T. Towards Unveiling Vulnerabilities and Securing IoT Devices: An Ontology-Based Approach. Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation. 2024. pp. 1–8.
17. Lavrova D.S., Vasilev Y.S. An ontological model of the domain of applications for the Internet of Things in analyzing information security. Automatic Control and Computer Sciences. 2017. vol. 51. no. 8. pp. 817–823.
18. Zegzhda D.P., Lavrova D.S. Approach to Internet of Things detection of security incidents using SIEM technology. Intellectulnye tekhnologii na transporte – Intellectual Technologies on Transport. 2017. no. 1. pp. 35–41. (In Russ.).
19. Nazir A., Sholla S., Bashir A. An Ontology based Approach for Context-Aware Security in the Internet of Things (IoT). International Journal of Wireless and Microwave Technologies (IJWMT). 2021. vol. 11. no. 1. pp. 28–46.
20. SAREF: Smart Applications REference ontology: v. 3.1.1. European Telecommunications Standards Institute. Available at: <https://saref.etsi.org/core/v3.1.1/> (accessed 05.02.2026). (In Russ.).
21. Web of Things (WoT) Security Ontology. World Wide Web Consortium (W3C). Available at: <https://www.w3.org/2019/wot/security> (accessed 05.02.2026). (In Russ.).
22. Fedorchenko E.V., Novikova E.S., Kotenko I.V., Gajfulina D.A., Tushkanova O.N., Levshun D.S., Meleshko A.V., Murenin I.N., Kolomeecz M.V. [Security Measurement System for Information and Personal Data of Internet of Things Devices]. Voprosy kiberbezopasnosti – Cybersecurity Issues. 2022. no. 5. pp. 28–46. (In Russ.).
23. Kotenko I., Fedorchenko A., Doynikova E., Chechulin A. An Ontology-based Storage of Security Information. Information Technology and Control. 2018. vol. 47. no. 4. pp. 1–13.
24. Fedorchenko E.V., Kryukov R.O., Kotenko I.V., et al. Security assessment based on attack graphs using NVD and MITRE ATT&CK database for heterogeneous infrastructures. Information and Control Systems. 2024. no. 2(129). pp. 39–50.
25. Makhdoom I., Abolhasan M., et al. Anatomy of Threats to the Internet of Things. IEEE Communications Surveys & Tutorials. 2019. vol. 21. no. 2. pp. 1636–1675.

26. Yaacoub J.P.A., Noura H.N., Salman O., Chehab A. Robotics cyber security: vulnerabilities, attacks, countermeasures, and recommendations. *International Journal of Information Security*. 2022. vol. 21. pp. 115–158.
27. Mishura M. CyberSecOnto-IoT: ontologiya kiberbezopasnosti Interneta veshhej [CyberSecOnto-IoT: Cybersecurity Ontology of the Internet of Things]. GitHub. Available at: <https://github.com/MashaMishura/CyberSecOnto-IoT.git> (accessed 05.02.2026). (In Russ.).
28. Prikaz Ministerstva cifrovogo razvitiya, svyazi i massovy'x kommunikacij RF ot 29 marta 2019 g. №113 «Ob utverzhdenii Konceptii postroeniya i razvitiya uzkopolosny'x besprovodny'x setej svyazi «Interneta veshhej» na territorii Rossijskoj Federacii» [Order of the Ministry of Digital Development, Communications and Mass Media of the Russian Federation No. 113 dated March 29, 2019 «On Approval of the Concept for the Construction and Development of Narrowband Wireless Communication Networks Internet of Things in the Russian Federation»]. Available at: <https://docs.cntd.ru/document/554066760> (accessed 05.02.2026). (In Russ.).
29. ISO/IEC 27400:2022. Cybersecurity – IoT security and privacy – Guidelines: international standard. Geneva: ISO. 2022.
30. NISTIR 8228. Considerations for Managing Internet of Things (IoT) Cybersecurity and Privacy Risks. NIST. 2019.
31. NISTIR 8259A. IoT Device Cybersecurity Capability Core Baseline. National Institute of Standards and Technology. NIST. 2020.
32. Federal'ny'j zakon Rossijskoj Federacii ot 26 iyulya 2017 g. №187 «O bezopasnosti kriticheskoy informacionnoj infrastruktury» Rossijskoj Federacii». [Federal Law of the Russian Federation No. 187 dated July 26, 2017 «On the Security of the Critical Information Infrastructure of the Russian Federation»]. Available at: <https://docs.cntd.ru/document/436752114> (accessed 05.02.2026). (In Russ.).
33. ETSI TS 103 264 V3.1.1 (2020-02). SmartM2M; Smart Applications; Reference Ontology and oneM2M Mapping: Technical Specification. European Telecommunications Standards Institute. Sophia Antipolis: ETSI. 2020. 78 p.
34. CVE: Common Vulnerabilities and Exposures. The MITRE Corporation. Available at: <https://cve.mitre.org/> (accessed 05.02.2026). (In Russ.).
35. RVD: Robotics Vulnerability Database. Alias Robotics. Available at: <https://github.com/aliasrobotics/RVD> (accessed 05.02.2026). (In Russ.).
36. CVSS v3.0 specification document. FIRST. Available at: <https://www.first.org/cvss/specification-document> (accessed 05.02.2026). (In Russ.).
37. Metodika ocenki urovnya kritichnosti uyazvimostej programmny'x, programmno-apparatny'x sredstv: metodicheskij dokument: utverzhden FSTE'K Rossii 30 iyunya 2025 g. [Methodology for Assessing the Criticality Level of Vulnerabilities in Software and Hardware: Methodological Document: Approved by FSTEC of Russia on June 30, 2025]. Available at: <https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-30-iyunya-2025-g> (accessed 05.02.2026). (In Russ.).
38. CAPEC – Common Attack Pattern Enumeration and Classification. The MITRE Corporation. Available at: <https://capec.mitre.org/> (accessed 05.02.2026). (In Russ.).
39. Metodika ocenki ugroz bezopasnosti informacii: metodicheskij dokument: utverzhden FSTE'K Rossii 5 fevralya 2021 g. [Methodology for Assessing Information Security Threats: Methodological Document: Approved by FSTEC of Russia on February 5, 2021]. Available at: <https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-5-fevralya-2021-g> (accessed 05.02.2026). (In Russ.).
40. Wu C.K. Architectures and security measures. *Internet of Things security*. Springer. 2021. pp. 13–26.

41. Dubey A., Yadav S. Basics of Internet of Things. International Journal of Scientific Research in Engineering and Management (IJSREM). 2024. vol. 8. no. 10. pp. 1–6.
42. Blinowski G., Piotrowski P. CVE based classification of vulnerable IoT systems. arXiv preprint arXiv:2006.16640. 2020.
43. Afroze M., Reena J. A Survey Study on IoT Application and its Attacks. International Research Journal on Advanced Engineering Hub (IRJAEH). vol. 2. no. 02. pp. 80–86.
44. Alam M., Khan I. Cyber Physical Attacks and IoT. Intelligent Cyber-Physical Systems Security for Industry 4.0. 2022. pp. 79–104.
45. Bank dannykh ugroz bezopasnosti informacii [Data Bank of Information Security Threats]. Available at: <https://bdu.fstec.ru/threat> (accessed 07.02.2026). (In Russ.).
46. Dojnikova E.V., Kotenko I.V. Ocenivanie zashchishhyonnosti i vy`bor kontrmer dlya upravleniya kiberbezopasnost`yu [Security Assessment and Countermeasure Selection for Cybersecurity Management]. Moscow: Rossijskaya akademiya nauk. 2021. 184 p. (In Russ.).
47. ENISA Threat Landscape 2024. European Union Agency for Cybersecurity. Athens: ENISA. 2024. 112 p.
48. Best Practice Guidelines. IoT Security Foundation. Available at: <https://iotsecurityfoundation.org/best-practice-guidelines/> (accessed 07.02.2026). (In Russ.).
49. Kaspersky IoT Threat Data Feed. Available at: <https://www.kaspersky.ru/iot-threat-data-feed> (accessed 07.02.2026). (In Russ.).
50. Protege. Available at: <http://protege.stanford.edu/> (accessed 07.02.2026). (In Russ.).
51. Common Platform Enumeration (CPE). National Institute of Standards and Technology. Available at: <https://nvd.nist.gov/products/cpe> (accessed 07.02.2026). (In Russ.).
52. National Vulnerability Database (NVD). National Institute of Standards and Technology. Available at: <https://nvd.nist.gov> (accessed 07.02.2026). (In Russ.).
53. Kumar S., Baliyan N. Semantic Web-Based Systems. SpringerBriefs in Computer Science. Springer. 2018. pp. 19–50.
54. Bank dannykh ugroz FSTEK Rossii. Formirovanie perechnya ugroz [FSTEC of Russia Threat Data Bank. Formation of the Threat List]. Available at: <https://bdu.fstec.ru/threat-section/shaper-threats> (accessed 26.05.2026). (In Russ.).

Firsova Maria — Senior Lecture, Department of Information Technology Security Named After O.B. Makarevich, Institute of Computer Technologies and Information Security, Federal State Autonomous Educational Institution of Higher Education “Southern Federal University” (FGAOU VO “SFU”). Research interests: analysis of information security of cyber-physical systems and Internet of Things systems, ontological engineering. The number of publications — 25. mshulika@sfsedu.ru; 2, Chekhov St., 347922, Taganrog, Russia; office phone: +7(951)502-1484.

Acknowledgements. This work is supported by the Gennady Komissarov Foundation for Young Scientists as part of the Eugene Kaspersky Scholarship Competition.