ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М.А. Бонч-Бруевича»

На правах рукописи

АЛЬ БАХРИ МАХМУД САИД НАССЕР

РАЗРАБОТКА МОДЕЛЕЙ И МЕТОДОВ ИДЕНТИФИКАЦИИ УСТРОЙСТВ И ПРИЛОЖЕНИЙ ИНТЕРНЕТА ВЕЩЕЙ НА БАЗЕ АРХИТЕКТУРЫ ЦИФРОВЫХ ОБЪЕКТОВ

Специальность: 05.12.13 – Системы, сети и устройства телекоммуникаций

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель
доктор технических наук
Киричек Руслан Валентинович

Санкт-Петербург – 2019

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы исследования

Интернет вещей (IoT – Internet of Things) является современной концепцией, подразумевающей объединение объектов, «вещей», в единую всемирную сеть, которая позволяет вещам быть умными для взаимодействия как с друг с другом, так и с человеком в любое время и в любом месте. На сегодняшний день число устройств, подключенных к сети, превышает число всех жителей планеты и продолжает стремительно увеличиваться, что поднимает вопрос о присвоении каждому объекту уникального адреса, обеспечения конфиденциальности и безопасности при передаче данных. Несмотря на это, до сих пор нет общепринятого метода идентификации вещей, который бы удовлетворял всем требованиям как для существующих устройств и приложений Интернета вещей, так и для вновь создаваемых.

Идентификатор представляет собой выделенный, публично известный атрибут или имя (или набор атрибутов и имен) для отдельного устройства. Как правило, идентификаторы действуют в пределах определенной области или сети, что затрудняет идентификацию вещей в глобальном масштабе. Ввиду сложности и высокой производительности современных устройств Интернета вещей они могут иметь более одного идентификатора. В тоже время, существуют различные методы идентификации, которые не могут использоваться многими устройствами Интернета вещей по различным причинам. Современные методы анонимизации и огромное число устройств Интернета вещей, подключенных к сетям связи общего пользования (ССОП), делают современные сети и системы связи уязвимыми перед злоумышленниками. Уязвимость сетевой безопасности, заключающейся в невозможности аутентификации устройств Интернета вещей, открывает для злоумышленников возможность для производства контрафактной физических и виртуальных вещей.

Одним из направлений обеспечения гарантированной и однозначной идентификации устройств Интернета вещей (ИВ) является использование уникального идентификатора устройства ИВ в ССОП в совокупности с

параметрами самого устройства. При этом надо учитывать, что так называемый универсальный идентификатор должен поддерживать (быть совместим) с уже существующими методами идентификации, такими как IMEI, MAC и другие. Необходимо также отметить, что от уровня к уровню идентификация устройств может подменяться, т.е. оконечному устройству интернета вещей с определенным физическим адресом на канальном уровне сначала назначается соответствующий логический адрес на сетевом уровне, который в последствии может быть заменен на идентификатор на уровне платформы. При этом очень важным свойством является фиксированность соотношения идентификатора с фактическим устройством Интернета вещей (физическим адресом), а также универсальность в применении идентификатора в различных отраслях.

С учетом того, что, по последним сведениям, количество уже подключенных устройств на планете достигает 9 миллиардов, которые расположены по всему миру необходимо также учитывать поддержку всех типов языков и децентрализацию систем регистрации цифровых объектов в интернете, чтобы обеспечить децентрализованную систему управления цифровыми объектами.

В связи с этим одной из самых важных проблем является выбор системы идентификации для всех устройств ИВ, подключенных к ССОП. В качестве уникального глобального идентификатора предлагается множество различных программных и аппаратных решений. Одним из решений, которое удовлетворяет предъявляемым требованиям по идентификации устройств и приложений интернета вещей является архитектура цифровых объектов DOA (Digital Object Architecture).

Архитектура цифровых объектов и ее базовая система резолюции "Handle system" была изначально создана как система резолюции идентификаторов, обладающая достаточной гибкостью использования. Идентификаторы содержат актуальную информацию об объекте – размещение, условия использования, ключи шифрования и т.д. Двухуровневая система резолюции и распределенная архитектура технологии позволяет быстро отображать изменения свойств объектов

и использовать собственную бизнес-модель для каждого администратора и сервера.

В связи с тем, что архитектура цифровых объектов наиболее полно удовлетворяет перечисленным выше требованиям разработка моделей и методов для идентификации устройств и приложений ИВ представляется весьма актуальным.

Степень разработанности темы. На сегодняшний день в научных школах, возглавляемых российскими и зарубежными учеными А. Е. Кучерявым, В. М. Вишневским, К. Е. Самуйловым, С. Н. Степановым, А. П. Пшеничниковым, А. В. Росляковым, В. Г. Карташевским, В. К. Сарьяном, Е. А. Кучерявым, М. А. Медришом, Р. В. Киричком и зарубежными Р. Э. Каном, К. Бланки, Л. Ланном, П. А. Лайонсом, Г. Манепалли, С. Саном и др. ведутся работы по исследованию Интернета вещей, а также сетей связи пятого поколения. Идентификация устройств и приложений Интернета Вещей является одной из задача рассматриваемых в перечисленных научных школах. Дальнейшее увеличение количества устройств Интернета вещей диктует необходимость уже сегодня предпринимать решительные действия по исследованию и внедрению новых методов идентификации. В связи с этим тема диссертационной работы «Разработка моделей и методов идентификации устройств и приложений интернета вещей на базе архитектуры цифровых объектов» является актуальной и направлена на совершенствование научно-методического аппарата исследования функционирования устройств Интернета вещей в гетерогенных сетях связи.

Цель работы и задачи исследования. Целью диссертационной работы является разработка моделей и методов идентификации устройств и приложений Интернета вещей на базе архитектуры цифровых объектов.

Для достижения поставленной цели решаются следующие задачи:

– проанализировать существующие методы идентификации для проводных и беспроводных технологий Интернета вещей;

– проанализировать архитектуру цифровых объектов и возможности ее использования в качестве платформы идентификации в современных телекоммуникационных сетях связи;

– разработать методы модернизации сетевой архитектуры цифровых объектов для улучшения параметров качества обслуживания при идентификации устройств и приложений Интернета вещей;

– проанализировать эффективность использования протоколов сигнализации архитектуры цифровых объектов для обеспечения сетевой безопасности устройств Интернета вещей в гетерогенных сетях связи на базе архитектуре цифровых объектов;

– разработать модель системы массового обслуживания характеризующую время обслуживания заявок и распределение времени между поступления заявок на сервер глобального регистра;

– разработать методы интеграции идентификаторов DOA для идентификации устройств и приложений Интернета вещей в гетерогенных сетях связи для проводных и беспроводных технологий передачи данных;

– провести серию натурных экспериментов для исследования параметров качества обслуживания при использовании архитектуры цифровых объектов для идентификации устройств и приложений Интернета вещей.

Объект исследования. Объектом исследования являются физические и виртуальные Интернет вещи.

Предмет исследования. Предметом исследования является идентификация устройств и приложений Интернета вещей в гетерогенных сетях связи.

Методологические и теоретические основы исследования. Проводимые исследования базируются на теории массового обслуживания, математической статистике, методах моделирования и натурных экспериментах. Моделирование фрагмента сети ИВ проведено на основе пакета имитационного моделирования Anylogic.

Научная новизна исследования.

1. Разработанный метод построения сетевой архитектуры цифровых объектов с промежуточным уровнем взаимодействия отличается от известных тем,

что позволяет снизить сетевую задержку при обмене служебными сообщениями между локальными и глобальными регистрами DOA.

2. Разработана модель обслуживания заявок на сервере GHR, отличающаяся от известных тем, что для представления этого процесса используется система массового обслуживания типа M/M/n/m. Это позволяет оценить производительность системы с произвольным распределением входных и выходных потоков и временем обслуживания на GHR и тем самым увеличить производительность (обслуживание заявок) в 15 раз при максимальной интенсивности нагрузки

3. Предложенный метод идентификации устройств и приложений Интернета вещей на базе архитектуре цифровых объектов, отличается от известных тем, что для идентификации устройств и приложений Интернета вещей предложено использовать архитектуру цифровых объектов, что позволяет интегрировать все уникальные параметры устройств, существующие идентификаторы и другие метаданные как для проводных, так и беспроводных технологий передачи данных.

4. Разработана модельная сеть, позволяющая проводить тестирование идентификации устройств и приложений Интернета вещей на базе архитектуры цифровых объектов, отличающаяся тем, что позволяет провести тестирования идентификаторов Интернета вещей для наиболее распространенных беспроводных технологий передачи данных.

Теоретическая и практическая значимость исследования: Теоретическая значимость диссертационной работы состоит в том, что на основе разработки моделей и методов идентификации устройств и приложений Интернета вещей были получены новые результаты, позволившие рассмотреть возможные сценарии внедрения идентификации на базе архитектуры цифровых объектов и предложить технические решения по обеспечению совместимости с существующими методами идентификации и функционирование в гетерогенных сетях связи.

Практическая значимость диссертационной работы подтверждается актом внедрения и состоит в разработке модельной сети для тестирования методов

идентификации устройств и приложений Интернета вещей на базе архитектуры цифровых объектов.

Основные положения и результаты, выносимые на защиту

1. Метод построения сетевой архитектуры цифровых объектов за счет введения промежуточного уровня взаимодействия.
2. Модель повышения производительности архитектуры цифровых объектов.
3. Метод идентификации устройств и приложений Интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов.

Достоверность полученных автором научных и практических результатов определяется обоснованным выбором исходных данных при постановке частных задач исследования, основных допущений и ограничений, принятых в процессе математического моделирования, соответствием расчетов с результатами экспериментальных исследований, проведенных лично автором, согласованностью с данными, полученными другими авторами и апробацией результатов исследований на международных, всероссийских и ведомственных научно-технических конференциях и конгрессах.

Степень достоверности и апробация результатов. Основные теоретические и практические результаты работы реализованы в учебном процессе кафедры Сетей связи и передачи данных Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича при чтении лекций, проведении практических занятий и лабораторных работ. Кроме того, научные результаты, полученные Аль Бахри М.С.Н., были использованы при подготовке вкладов СПбГУТ в Сектор Стандартизации Телекоммуникаций Международного Союза Электросвязи.

Апробация результатов исследования

Основные результаты диссертационной работы докладывались и обсуждались на 4-й Международной научно-технической и научно-методической конференции «Актуальные проблемы инфотелекоммуникаций в науке и образовании» 2015, Международной научной конференции «Молодежная научная школа по прикладной теории вероятностей и телекоммуникационным технологиям» (АРСТ) 2017, 71–73 Всероссийской научно-технической конференции, посвященной Дню радио – 2016–2018 гг., 3ей Международной

конференции молодых ученых «Интернете вещей и его приложения» INTHITEN 2017, 10-м Международном конгрессе по ультрасовременным системам телекоммуникаций и управления (ICUMT 2018), 18-й Международной конференции «Интернет вещей, умные пространства, сети и системы следующего поколения» (NEW2AN 2018), 4-й Международной конференции по мобильным пограничным вычислениям «Большие данные и умный город» (ICBDSC 2019).

Лабораторные стенды демонстрировались на Региональном форуме МСЭ «Интернет вещей, телекоммуникационные сети и большие данные как базовая инфраструктура для цифровой экономики» 4–6 июня 2018 года в Санкт-Петербурге и семинаре-практикуме МСЭ «Глобальные подходы к борьбе с контрафактом и похищенными устройствами ИКТ» 23 июня 2018 года в Женеве.

Публикации по теме диссертации. По теме диссертации опубликовано 16 научных работ, из них: 4 в рецензируемых научных изданиях; 6 в изданиях, индексируемых в международных базах данных; 6 в других изданиях и материалах конференций.

Личный вклад автора. Основные результаты теоретических и экспериментальных исследований получены автором самостоятельно. В работах, опубликованных в соавторстве, соискателю принадлежит основная роль при постановке и решении задач, а также обобщении полученных результатов.

Соответствие специальности. Диссертационная работа соответствует пунктам 3, 10, 14 паспорта специальности 05.12.13 – «Системы, сети и устройства телекоммуникаций».

Структура и объем диссертации. Диссертация состоит из введения, четырех глав с выводами по каждой из них, заключения, списка литературы и 2 приложений. Общий объём работы –153 страниц, из них основного текста 142 страниц. Работа содержит 13 рисунков и 18 таблицы. Список литературы включает 142 источника.

Глава 1. АНАЛИЗ МЕТОДОВ ИДЕНТИФИКАЦИИ УСТРОЙСТВ И ПРИЛОЖЕНИЙ ИНТЕРНЕТА ВЕЩЕЙ

1.1. Идентификация в эпоху Интернете вещей: новые вызовы и возможности

В настоящее время Интернет вещей является общепризнанной концепцией развития сетей связи в краткосрочной и долгосрочной перспективах, а также передовой платформой в рамках развития цифрового интеллекта в концепции «Умная страна» [49]. По мнению большинства консалтинговых аналитических компаний, в течение следующих пяти лет в каждой из сфер жизнедеятельности человека будет присутствовать более 25 миллиардов устройств. Таким образом, можно говорить о всепроникающем характере проникновения Интернета вещей в нашу повседневную жизнь.

Как известно, фраза «Интернет вещей» впервые прозвучала от Кевина Эштона в 1999 году на презентации инновационных решений компании «Проктер и Гэмбл». Эштону предложил нанести RFID метки на продукцию, выпускаемую компанией, и таким образом обеспечить её взаимодействие с радиоприемником. Кевин Этон предположил, что такой сбор данных может быть использован для решения многих проблем в реальном мире. В результате в настоящее время многие устройства могут обмениваться данными через Интернет, взаимодействуя со смартфонами, друг с другом и с аналогичными похожими устройствами [69]. В 2001 году исследовательский центр Auto-ID Массачусетского технологического института, в котором работал Кевин Эштон, адаптировал использование RFID меток для разнообразной продукции, местонахождение которой стало возможно отслеживать через Интернет. В 2005 году термин Интернет вещей был официально использован Международным союзом электросвязи (МСЭ) в техническом отчете, посвящённом перспективным концепциям развития сетей связи [3].

В последнее десятилетие Интернет вещей стал одной из прорывных технологий, общепризнанных всеми странами Мира. ИВ позволяет людям и вещам взаимодействовать где угодно, когда угодно, и в любых сочетаниях при использовании инфраструктуры Интернета вещей. Экосистема ИВ предполагает

сбор данных с датчиков (либо отправку команд на исполнительные устройства), их передачу через сеть связи на облачные платформы для последующего анализа с целью предоставления интеллектуальных услуг для людей. На Рисунке 1.1 представлены ключевые компоненты, необходимые для построения систем ИВ. Согласно рисунку, датчики и устройства съема информации собирают различные виды данных о том или ином объекте, затем эти данные могут быть дополнительно обработаны и проанализированы для извлечения полезной информации с целью предоставления интеллектуальных услуг [88]. Интернет вещей можно рассматривать как совокупность четырех основных элементов:

1) интернет: для обеспечения связи в любое время и в любом месте между любыми участниками межсетевого обмена. Также предполагаются облачные вычисления, интеллектуальные веб-сервисы и др.;

2) аппаратное обеспечение: предполагает коммуникационное оборудование, а также оконечные устройства съема, такое как датчики, метки, исполнительные механизмы и приема-передатчики;

3) промежуточное программное обеспечение: используется для хранения данных, вычислений и анализу передаваемых данных;

4) интерфейс: используется с целью визуализации и интерпретации собранных результатов для различных платформ и приложений.

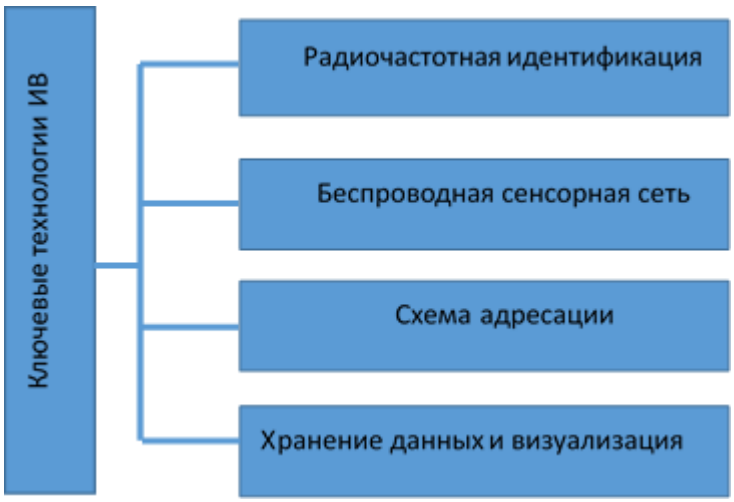


Рисунок 1.1 – Основные компоненты Интернета вещей

Существуют различные приложения ИВ, которые направлены на решения конкретных задач. Среди типовых приложений можно выделить: управление данными, аналитику, визуализацию, управление гетерогенными сетями, исследовательские цели и др. [70]. Тем не менее, исследования ИВ все еще продолжают находиться в зачаточном состоянии, ввиду существования многих нерешенных проблем, например, проблем, связанных с временем автономной работы, простотой «легковесности» технологий передачи данных, выполнением действий в зависимости от контекста происходящего, вопросами идентификации и безопасности, стоимости оконечных устройств, масштабируемости и гетерогенности [115].

Несмотря на все преимущества Интернета вещей в последнее время появились случаи раскрытия данных, собираемых устройствами ИВ, что заставляет беспокоиться о идентичности устройств и приложений в рамках концепции Интернета вещей. Действительно, идентификация играет важную роль в Интернете вещей. К примеру, злоумышленники могут использовать портативные RFID/NFC-считыватели для кражи персональных данных с банковских карт в общественном транспорте, используя уязвимости технологии типа PayPass. Это возможно благодаря отсутствию подтверждения личности владельца RFID-считывателя. Другим примером является возможность перехвата злоумышленником данных сетей устройств ИВ в целях получения IMEI-идентификаторов различных оконечных устройств, оснащенных модемами, с целью последующей широковещательной рассылки преднамеренно искаженных сообщений.

Текущие решения, известные во всем мире, направлены, в основном, на привязку устройства или приложения ИВ с идентификатором, подобным IP-адресу или номеру мобильного телефона, по которому можно понять: кто пользуется тем или иным устройством. Исследования в этой области были начаты в результате обсуждения этих проблем в регулирующем органе BEREC (Body of European Regulators for Electronic Communications) [86]. В то же время, идентификация имеет гораздо более широкие масштабы и является более уместной для множества приложений и сущностей (субъектов) в ИВ. Помимо целей идентификации в сфере

коммуникаций, проводимые исследования включают вопросы идентификации физических и виртуальных вещей, таких как услуги для пользователей с использованием ИВ, собираемых данных, местоположения. Различные схемы идентификации, существующие на сегодняшний день, уже стандартизованы, а также внедрены на в множестве устройств, доступных в открытой продаже [68; 87].

В зависимости от сферы применения и требований пользователей применяются различные типы идентификаторов. В самой основе Интернета вещей лежит взаимодействие между вещами и пользователями вещей с помощью вспомогательных элементов экосистемы: датчики, исполнительные механизмы и беспроводная связь, облачные платформы и др [66]. Вещи и пользователи должны быть однозначно идентифицированы с целью понимания уникальности того или иного объекта взаимодействия. Множество других сущностей также вовлечены во взаимодействие, одновременно являясь частью экосистемы ИВ, для них идентификация также является важным аспектом. Взаимодействие различных сущностей с привязанными идентификаторами в рамках концепции ИВ показаны на примере AIOTI WG03 High Level Architecture [85] (Рисунок 1.2).

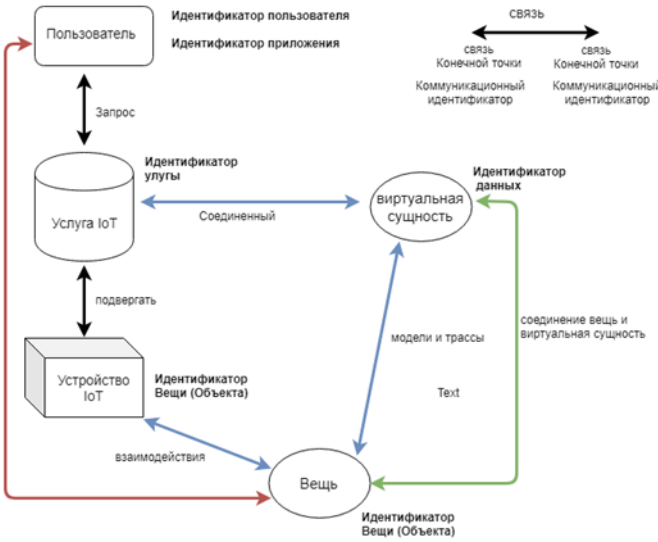


Рисунок 1.2 – Взаимодействие различных сущностей с привязанными идентификаторами в рамках концепции ИВ

1.2. Классификация идентификаторов для Интернета вещей

На сегодняшний день идентификаторы используются для различных целей в приложениях Интернета вещей. Основной задачей идентификатора, присваиваемого той или иной вещи, является идентификация, позволяющая однозначно определять вещи и являться целевыми сущностями приложений Интернета вещей. Помимо идентификации вещей, идентификации также подлежат приложения и услуги, пользователи, данные, оконечное оборудование, протоколы и места нахождения вещей. Ниже будет представлена более точная классификация идентификаторов ИВ.

1.2.1 Идентификатор объекта

Идентификатор объекта определяет целевую сущность приложения Интернета вещей. Это может быть, к примеру, любой физический объект (оборудование, помещения, люди, животные, растения) или цифровые данные (файл, набор данных, метаданные), т. е. что угодно, с чем можно взаимодействовать в реальном и виртуальном мире.

Примеры использования идентификаторов вещей

1. ***Предиктивное обслуживание.*** Компании могут предоставлять услуги по предиктивному обслуживанию их продуктов (например, электроприводы, производственное оборудование). Продукты, при этом, должны иметь встроенные сенсоры и интерфейсы для коммуникации. Сервис по предиктивному обслуживанию располагается в облачном сервисе. Соединение с оборудованием на территории клиента осуществляется через защищенное соединение (например, VPN) при помощи сетевого соединения клиента или посредством мобильного Интернет соединения. Продукт имеет встроенный в энергонезависимую память идентификатор, посредством которого оборудование и определяется на облачной платформе.

2. ***Отслеживание имущества.*** Компании могут следить за собственным имуществом (большого и малого размеров, движимое и недвижимое) путем регулярной проверки его местоположения. В данном случае, любое имущество

имеет собственный идентификатор объекта, выполненный в виде штрих-кода, QR-кода или RFID-метки. Полученные метки подлежат постоянному сканированию персоналом компании при помощи ручного сканера, осуществляющего соединение с сервером. С каждым сканированием сопровождающая информация об имуществе может быть предоставлена при помощи пользовательского интерфейса сканера [9].

3. ***Происхождение и контроль качества отслеживаемой информации.*** Следующий пример показывает важность чёткого определения объекта. Грузовая логистическая компания маркирует транспортируемый товар при помощи меток RFID. Данные метки содержат идентификатор объекта транспортируемого продукта совместно с любыми другими атрибутами (производитель, дата производства и др.). Местоположение продукта записывается при прохождении пунктов считывания. В дальнейшем, данные метки могут быть повторно использованы для других продуктов с другим идентификатором объекта. Метка сама по себе также хранит собственный идентификатор метки, который используется компанией для определения происхождения информации, контроля качества меток и др.

Примером подобных идентификаторов, хранящихся на одной метке, но при этом относящихся к различным сущностям, является электронный код продукта (Electronic Product Code, EPC), а также идентификатор метки (Tag Identifier, TID), определенные международной организацией GS1 [11]. Электронный код продукта идентифицирует продукт, к которому прикреплена метка, в то время как идентификатор метки идентифицирует непосредственно метку. В отличии от идентификатора метки, который не меняется на протяжении жизни, электронный код продукта меняется с каждым новым продуктом, к которому метка прикрепляется.

1.2.2. Идентификатор приложений и услуги

Идентификаторы приложений и сервисов определяет приложения и сервисы, что также включает в себя способы взаимодействия с приложением или сервисом (например, API, RPC).

Примеры использования идентификаторов приложений и сервисов

Услуги на базе платформ Интернета вещей. Платформа Интернета Вещей может предоставлять различные сервисы, например, сервис обеспечения связи, магазин приложений, сервис управления устройствами, сервис регистрации устройств. Каждый сервис имеет уникальный идентификатор. Сервисы могут быть занесены в реестр, что позволит приложениям осуществлять поиск сервисов. Сервисы также могут быть представлены приложениям. Для федеративных платформ (как правило, функционирующих в пределах страны), в случаях, когда один и тот же сервис (к примеру, сервис регистрации) может быть предоставлен различными (к примеру, региональными) программными платформами, возможно присвоение множества уникальных идентификаторов для определенного числа услуг одного и того же типа [12].

1.2.3. Коммуникационные идентификаторы

Коммуникационные идентификаторы определяют оконечное коммуникационное оборудование (к примеру, источник или получатель), а также сессии.

Примеры использования идентификаторов связи

Основываясь на примере, описанном в документе ETSI GS LTN 002 [12] Европейского института по стандартизации в области телекоммуникаций, Энергоэффективные сети дальнего радиуса действия (Low Power Area Networks, LPWAN)[47], используют уникально присвоенные коммуникационные идентификаторы для определения оконечного оборудования в пределах каждой из сетей. Централизованные центры обслуживания обмениваются данными с оконечным оборудованием через точки доступа в обоих направлениях. Оконечное оборудование зарегистрировано в системе при помощи уникального коммуникационного идентификатора [114;115]. При установлении соединения от терминала к сервису обслуживания, оконечные устройства используют собственные коммуникационные идентификаторы в качестве адреса отправителя с целью последующей успешной обработки и маршрутизации пакета до

центрального сервиса. В случае соединения происходит запрос от сервиса обслуживания к терминалу. Конечные устройства запрашивают в сети существующие данные, используя собственный коммуникационный идентификатор в качестве адреса получателя.

1. *Ethernet/WiFi MAC адрес*. В сетях на основе технологий Ethernet/WiFi (прим. IEEE 802.3 [16]) MAC адрес является идентификатором для коммуникационного оконечного оборудования на канальном уровне. MAC-адрес устройства назначается производителям оборудования. Данный адрес состоит из 48 бит (6 байт), где первые 3 байта обозначают уникальный идентификатор организации (Organizationally Unique Identifier, OUI), назначаемый компаниям регистрирующим органом IEEE (Институт инженеров электротехники и электроники).

2. *IP-адрес.* Адреса IPv4 и IPv6 (прим. IETF RFC 4291 [15]) используются в сетях IP для логической идентификации оконечного оборудования на сетевом уровне. Размер адреса IPv4 равен 32 битам, в то время как размер адреса IPv6 равняется 128 битам. IP адрес бывает глобальный (публичный), локальный или типа Link-local, в зависимости от сферы применения и используемой сети. Более того, поддерживаются также адреса для широковещательной, многоадресной и одноадресной рассылки (broadcast, multicast, unicast). IP-адреса структурно основаны на принципе маршрутизации и состоят из сетевого префикса и идентификатора интерфейса, размер которых может варьироваться. Глобально уникальный диапазон IP-адресов распределен между пятью основными региональными интернет-регистраторами (Regional Internet Registries, RIRs), который в дальнейшем может распределяться между интернет-провайдерами на непосредственно пользовательские сети. Управление глобальным набором адресов осуществляется реестром доменов верхнего уровня (IANA), который и выделяет блоки IP-адресов региональным интернет-регистраторам [124].

3. *Телефонный номер*. Телефонные номера присваиваются конкретному устройству абонента в телефонной сети. В зависимости от сферы применения, могут использоваться глобально и локально уникальные номера. Для звонков при

помощи локального номера в глобальную сеть используется номер с расширением, предоставляющий глобальную уникальность. Глобальный телефонный номер начинается с кода страны, определенным Международным Союзом Электросвязи (прим. ITU-T E.164 [16]). Коды регионов или провайдеров назначаются регулирующими организациями в конкретной стране.

4. *Сессия HTTP.* Коммуникационной сессией можно считать обмен серией связанных между собой сообщений. Примером может являться интернет-магазин, в котором пользователь может наполнить корзину несколькими позициями и затем осуществить оплату. Веб-серверу необходимо отслеживать все действия пользователя в контексте магазина. Протокол HTTP не предоставляет механизмов по сохранению состояния, поэтому необходимо сохранять выделенный идентификатор сессий, чтобы предоставлять подобный функционал магазина. Идентификатор генерируется сервером и обычно хранится в качестве специального фрагмента данных на стороне клиента, являющийся параметром в запросах HTTP GET и POST.

1.2.4. Идентификатор пользователя

Идентификатор пользователя однозначно определяет пользователя сервиса или приложения Интернета вещей. Пользователем может быть человек, компания (юридическое лицо) или даже программное обеспечение, взаимодействующее с соответствующими приложениями Интернета вещей.

Примеры использования идентификаторов пользователей

1. *Пользователь-человек.* С целью получения определенной информации от устройства, осуществляющего управление устройствами интернета вещей, человеку необходимо авторизоваться в системе. Для этого сначала человеку необходимо идентифицировать себя в системе, например, при помощи имени пользователя, специальной чип-карты или отпечатка пальца. В зависимости от системы безопасности, возможна аутентификация дополнительными методами. Система проверяет наличие прав доступа у конкретного пользователя к объектам или сервисам Интернета вещей и производит необходимые действия. Права

пользователя зависят от группы, к которой он принадлежит. Внутри системы Интернета вещей, пользователю назначается особый идентификатор, который привязывается ко всем операциям, связанными с безопасностью, и который может отличаться от идентификатора, используемого человеком для собственных идентификаций [72].

2. Программный доступ к объектам Интернета вещей. В определенных сценариях возможно взаимодействие программного обеспечения приложений с объектом ИВ при помощи системы ИВ. Приложение представляет себя системе в виде определенного ключа. Система проверяет наличие необходимых прав у приложения для осуществления доступа к объекту ИВ и выполнения необходимых действий.

1.2.5. Идентификатор данных

Данный класс покрывает одновременно идентификацию особых видов данных и типов данных (например, метаданные, свойства, классы).

Примеры использования идентификаторов данных

1. ***Цифровой близнец.*** Цифровой близнец – это набор данных, содержащий виртуальное представление об объекте. Он связан с вещью через идентификатор объекта. Более того, в целях обращения и осуществления доступа из сервисов и приложений, сам цифровой близнец также нуждается в идентификаторе. Однако, сама вещь может иметь множество цифровых близнецов, которые, в свою очередь, могут содержать различные наборы информации.

2. ***Набор данных временного ряда.*** Сбор данных с сенсоров устройства интернета вещей происходит автоматически с постоянной частотой. Данные хранятся в качестве временного ряда непосредственно на платформе Интернета вещей для дальнейшего использования. Различные приложения могут осуществлять доступ к этим данным, например, для предиктивного обслуживания, оптимизации процессов или прогнозов. Набор данных нуждается в особом идентификаторе, который бы позволил обращаться к таким данным из приложений.

3. Типы свойств объектов. Свойства объекта, такие как вес, размеры и температура, являются стандартизированными для определенных сфер применения. Определение свойств включает значение, диапазон значения, формат конкретного свойства. Каждое из подобных определений нуждается в уникальном идентификаторе в целях однозначного обращения к таковым.

1.2.6. Идентификатор местоположения

Данный раздел рассматривает идентификацию местоположений в географических районах (координаты в пространстве, почтовые адреса, номера комнат) [11].

Примеры использования идентификаторов местоположения

1. ***Отслеживание продуктов***. Компания может отслеживать доставку товаров высокой стоимости. GPS-приёмник с модемом для передачи данных по сети является частью транспортировочной упаковки. GPS-координаты упаковки передаются с периодическими интервалами в облачное приложение, отслеживающее путь движения продукта.

2. ***Обслуживание недвижимости***. Руководитель объекта должен контролировать своевременное обслуживание систем нагрева, вентиляции и кондиционирования (HVAC) на больших территориях. Системы HVAC оповещает об аварийных ситуациях, что используется совместно с сервисами предиктивного обслуживания. В целях сопровождения обслуживающего персонала к необходимому месту аварии, в каждое из устройств необходимо встроить идентификатор, привязывающий устройство к определенной локации (например, здание, этаж, номер комнаты).

1.2.7. Идентификатор протокола

Идентификаторы протоколов могут информировать, к примеру, коммуникационные протоколы о протоколах вышележащих уровней, данные которых они передают с нижних уровней, или также возможно осуществить уведомление приложений о протоколах, которые лучше использовать для совершения необходимого обмена данными.

Примеры использования идентификаторов протоколов

1. ***Использование различных значений поля Ethertype.*** Высокоуровневые протоколы инкапсулируются в кадр Ethernet. Поле Ethertype в заголовке кадра Ethernet указывает, какой высокоуровневый протокол передаётся в данном пакете (см. IEEE 802.3 [13]).

2. ***Поле «Следующий заголовок» IPv6.*** Данное поле явно указывает тип протокола транспортного уровня, передаваемого при помощи протокола IP. В случае использования расширенных заголовков это поле также показывает, какое именно расширение заголовка используется (см. IETF RFC 8200 [21]).

3. ***Схема URI.*** В унифицированном идентификаторе ресурса (URI) существует поле «схема», явно указывающее на способ интерпретации самого URI (см. IETF RFC 3968 [18]). Зачастую это поле отображает, какой протокол используется для доступа к ресурсу, идентифицируемому по URI (например, HTTP, FT, NNTP).

1.3. Категории требований для идентификаторов в Интернете вещей

Ниже будет представлен список требований с различным уровнем детальности. Стоит отметить, что большинство требований не ограничены конкретным идентификатором класса или набором классов. По причине различного уровня детальности, а также разнообразия природы представленных требований, категории требований определены и детализированы в последующий главах. Пример по каждой из категории требований сведены воедино, в соответствии со стандартом AIOTI WG03 [22].

1.3.1. Уникальность

Требования по уникальности идентификаторов в зависимости от области применения сильно различаются. Многие ожидают глобальной уникальности, в то время как некоторые ограничивают уникальность до локальной, или до локальной в пределах домена продавца или производителя. Также, по причинам

конфиденциальности, в одном из вариантов использования предлагалось не использовать уникальность вовсе.

Отдельной темой для обсуждения является способ обеспечения уникальности идентификаторов, находящихся под управлением различных организаций.

1.3.2. Конфиденциальность и защита личных данных

Конфиденциальность является основой в темах, связанных с взаимодействием людей и их личных данных. Данная тема связана с идентификаторами пользователей, напрямую определяющих конкретных людей, а также идентификаторов для сущностей, которые могут быть очень близко связаны с людьми и их деятельностью. Например, автомобиль, личное оборудование, товары, местоположения, адреса для взаимодействия (коммуникации) каждого из которых принадлежит или назначена конкретному человеку (пользователю) или оборудованию в его владении[77].

1.3.3. Безопасность

Требования безопасности чаще всего связаны непосредственно с идентификатором, но в некоторых случаях также ожидается обеспечение безопасности данных, связанных с идентификатором (связанных с сущностью, идентифицируемой при помощи идентификатора). Обеспечение безопасности данных, связанных с идентификатором, не рассматривается в данном документе. Необходимо обеспечить корректную идентификацию сущности при помощи идентификатора, исключить подделку в процессе выделения сущности из множества других, во время перемещения и использования. Назначение подписи для идентификатора указывается как один из главных способов достижения данной цели. Дублирование и использование идентификатора для других сущностей должно быть предотвращено. Проверка правильности идентификатора должна быть доступна через глобальную сеть, а также в случае отсутствия Интернет-соединения.

Аутентификация идентификатора также является желаемой функцией, т.к. необходим способ доказать принадлежность идентификатора к правильной сущности. Данный функционал относится к функционалу системы управления идентификацией и не описывается в рассматриваемом документе.

1.3.4. Идентифицированные объекты

Множество разнообразных сущностей, начиная от крошечных объектов подобно таблеткам, заканчивая объектами больших размеров, подобных машинам, транспортным средствам, целым предприятиям, а также живым сущностям (люди, животные, растения) рассмотрено как потенциальные объекты для идентификации. Для этого необходимо наличие идентификации составных частей больших сущностей. Также необходима возможность идентификации нефизических вещей, например, наборы данных, организаций, потоков трафика. В общем смысле, идентификации может быть подвержено что угодно.

В ряде случаев может также потребоваться поддержка множества идентификаторов для одной и той же сущности, т.к. различные посредники или приложения могут использовать различные схемы идентификации (например, идентификатор производителя, идентификатор управления активами владельца/пользователя).

Схемы идентификатора должны чётко соответствовать конкретной схеме использования и идентифицируемой сущности (например, вещь, пользователь, приложение). К примеру, сетевой адрес (коммуникационный идентификатор) не должен использоваться в качестве идентификатора объекта, т.к. сетевой адрес объекта может измениться за время его жизни.

1.3.5. Прослеживаемость, подлинность и происхождение

Прослеживаемость указывается в отношении идентификатора и идентифицируемой сущности. Прослеживаемость идентифицируемой сущности (например, отслеживание сущности за время её жизни для предоставления безопасности пищевых товаров и устойчивости производства) является схемой использования идентификатора, и не является конкретным требованием для

процесса идентификации, прослеживаемость не должна поддерживаться в случае возможности предоставления с её помощью личных данных особенно при необходимости обеспечения конфиденциальности. Для идентификатора необходимо предусмотреть наличие возможности прослеживания идентификатора издателем.

1.3.6. Масштабируемость

Возможность масштабируемости идентификаторов остается одной из сложных и не решенных проблем. К тому же, не предоставлены конкретные числа по возможному количеству идентификаторов. Одной из первоочередных задач является решение проблем с масштабируемостью подобно вопросу ограниченности адресного пространства IPv4.

Помимо непосредственно возможности масштабирования модели идентификатора, необходимо поддержание в будущем жизненного цикла идентификатора и его обработки.

1.3.7. Совместимость и стандарты

Совместимость идентификаторов внутри и на всем множестве различных приложений, индустрий и географических регионов является важной вехой в организации единого пространства Интернета вещей. Различные признанные и только формирующиеся схемы идентификации в различных сферах должны быть учтены и включены во вновь создаваемые идентификаторы для обеспечения их совместимости.

В случае использования множества идентификаторов для одной и той же сущности, необходимо включить поддержку отображения между данными идентификаторами. Отображение между различными идентификаторами связанных сущностей (например, идентификатор объекта и коммуникационный идентификатор сетевого интерфейса данного объекта) должно также поддерживаться.

Одной из ожидаемых функций является возможность использования базовых схем идентификации на передаче через различные зоны, в которых применяется

собственная идентификация. Всемирные стандарты являются одним из способов достижения данного функционала. Использование решений сообществ в сфере международной стандартизации, а также решений сообщества с открытым исходным кодом являются одним из возможных решений проблемы совместимости идентификаторов.

1.3.8. Постоянство и повторное использование

В настоящее время существует потребность в необходимости использования постоянных идентификаторов на всём времени жизни сущности (вещи). Вероятно, что стоит предусмотреть ситуацию, согласно которой идентификаторы могут быть изменены, к примеру, если меняется владелец сущности, а также возможность отзыва и замены идентификаторов. Кроме этого, целесообразно рассмотреть невозможность повторного использования идентификаторов после окончания жизненного цикла сущности (а в ряде случаев предусмотреть повторное использование).

1.3.9. Распределение, регистрация и разрешение

Назначение идентификаторов должно быть организовано таким образом, чтобы организации могли назначать их личный набор идентификаторов без конфликтов с другими организациями (федеративный подход). Орган, ответственный за выдачу идентификаторов, должен вести учёт идентификаторов (индивидуальных или целых наборов), назначенных организациям.

Должна существовать возможность регистрации идентификаторов в глобальной базе данных, которая может хранить информацию о идентифицируемой сущности и способах доступа к ней. Должны поддерживаться различные схемы идентификации.

Информация, связанная с идентифицируемой сущностью должна быть доступна путем использования идентификатора. Это может быть достигнуто при предоставлении сущностью ссылки на другое местоположение. В зависимости от применения, информация может быть доступной не только при наличии соединения к сети Интернет, но и без него.

1.4. Стандарты идентификаторов

В настоящий момент проводится множество работ по стандартизации идентификаторов, одновременно с уже существующим множеством стандартов в этой области разрабатываются всё новые и новые документы, в которых учитывается специфика устройств и приложений Интернета вещей. Большинство из них применяемы только для определенных сфер деятельности или сценариев применения. Стандарты идентификации зачастую применимы к более чем одному классу идентификаторов.

В рамках диссертационной работы невозможно учесть все имеющиеся на сегодняшний день идентификаторы, но будет приведен перечень стандартов и рекомендаций так или иначе имеющих отношение к Интернету вещей. Полный перечень был бы полезен лишь в том случае, если бы мы могли использовать эти стандарты для обеспечения совместимости и актуальности в решениях Интернета вещей для каждого стандарта. Однако, это не представляется возможным в связи с огромным количеством пересекающихся стандартов, с ограниченным доступом, а также из-за объема работы, который необходимо проделать для детального анализа. Вместо этого, для каждой категории идентификаторов в контексте Интернета вещей представлены примеры стандартов. Также стоит отметить, что это не означает, что выбранные стандарты являются предпочтительными или обязательными к применению.

Стоит отметить, что помимо стандартов идентификации определенных организаций по разработке стандартов, государственными органами определены идентификаторы для определенных сфер применения, например, номера социального страхования и номера автомобилей. Также, компании могут иметь собственные «реализации» понятия «идентификатор», подобно серийным номерам у продуктов.

1.4.1. Стандарты идентификации вещей

Существует большое количество стандартов для идентификации вещей. Зачастую они определены для специфичных сфер или специфичных типов

сущностей, но некоторые из них используются во множестве сфер применения и для различных типов и классов сущностей. Некоторые стандарты предоставляют механизмы для схем множественной идентификации, позволяющие реализовать межсетевое взаимодействие внутри одного и того же приложения Интернета Вещей. Данный механизм относится к схемам мета-идентификации.

Примеры

1. ***VIN-номер автомобиля*** (Vehicle Identification Number), ISO 3779 [6], определяет универсальную систему номерной идентификации для автомобилей.

2. ***Кодирование грузовых контейнеров***, идентификация и маркировка определены в стандарте ISO 6346 [7]. Предоставляет систему идентификации с обязательной маркировкой для визуальной интерпретации и опциональные возможности для автоматизированной идентификации и электронного обмена данными, а также система кодирования для данных, зависимая от типа и размера контейнера.

3. ***Идентификация животных*** на базе радиочастотных меток определена в стандарте ISO 11784 [8], независимо от протокола передачи между меткой и считывателем.

4. ***Идентификация RFID-меток*** при помощи системы присвоения уникальных идентификаторов определена в стандарте ISO/IEC 15963 [23]. Идентификатор метки (Tag ID, TID) может использоваться для отслеживания и контроля состояния самой метки. Также может быть использован для отслеживания объекта, к которому метка прикреплена. Считается хорошей практикой идентифицировать объекты независимо от технологии передачи данных.

5. ***Юридические лица*** также могут быть идентифицированы уникальным глобальным номером при помощи уникального идентификатора юридического лица (Legal entity identifier, LEI), определённом в стандарте ISO 17442 [13]. Стандарт разрабатывался для применения в контексте услуг финансового сектора. Тем не менее, он также может быть использован для любых случаев, где необходимо ссылаться на юридическое лицо.

6. ***Уникальный идентификатор продукта*** для единиц логистики, продуктов, возвращаемых товаров описан в серии стандартов ISO/IEC 15459 [10]. Стандарт позволяет соответствующим учреждениям определять по идентификатору тип поставщика. Данные учреждения могут использовать существующие идентификаторы без конфликтов с аналогичными идентификаторами в приложениях Интернета вещей ввиду применения этих идентификаторов внутри локальных зон. Примером использования данного стандарта является GS1 [25].

7. ***Цифровой идентификатор объекта*** (Digital Object Identifier, DOI) определен в ISO 26324 [26, 133] в целях идентификации сущностей в Интернете и используется в основном для предоставления доступа к объекту, сообществу или для управления интеллектуальной собственностью [15]. Система DOI изначально была спроектирована для обеспечения совместимости и работы с существующими системами идентификации, а также схемами метаданных. В диссертационной работе рассматриваются модели и методы идентификации устройств и приложений Интернета вещей на базе архитектура цифровых объектов. Далее более подробно будет описано взаимодействие основных элементов этой архитектуры.

1.4.2. Стандарты идентификаторов приложений и услуг

Идентификаторы приложений и сервисов обычно определены в контексте конкретных платформ (сервисной платформы, операционной системы), с которой они предоставляются. Подобные идентификаторы могут быть как стандартными, так и проприетарными (закрытыми). В случае, если платформа стандартизована, то приложение и сервис также являются стандартизованными.

Примеры

1. ***Идентификаторы приложений и сервисом OneM2M.*** Стандарт OneM2M TS-0001 [27] определяет различные идентификаторы, используемые в решениях на основе Интернета вещей от организации OneM2M. Включает в себя идентификаторы для приложений, сущностей в приложениях и общих сущностях сервисов.

2. *Идентификатор ресурсов REST.* Передача состояния представления (Representational State Transfer, REST) является парадигмой программирования для встраиваемых систем. Представляет из себя способ предоставления сервисов от одного устройства к другому при помощи универсального и предопределенного набора операций без статуса. Ресурсы данных сервисов идентифицируются по URI. Формат URI определен в IETF RFC 3968 [29].

1.4.3. Стандарты идентификаторов в сетях передачи данных

Коммуникационные идентификаторы являются неотъемлемой частью коммуникационных протоколов и влияют на их функциональность (прим. маршрутизация и коммутация). Обычно, схема идентификации не может быть изменена без внесения существенных изменений в сам протокол. По этой причине, идентификаторы определены как часть определенного стандарта коммуникационного протокола.

Примеры

1. *Адрес IPv6.* Стандарт IETF RFC 4291 [15] определяет архитектуру адресации для IPv6. IPv6 адреса представляют из себя 128-битный идентификатор для интерфейсов (unicast) и набора интерфейсов (anycast и multicast).

2. *MAC-адрес.* Стандарт IEEE 802[28] определяет MAC-адрес, сетевой адрес для большинства сетевых технологий IEEE 802, подобных Ethernet или беспроводным локальным сетям (WLAN) на базе технологии WiFi. Они могут использовать 48- или 64-битные номера, но большинство стандартов IEEE 802 используют исключительно 48-битные MAC-адреса. MAC-адреса могут быть глобально или локально администрируемы. Универсальные MAC-адреса администрируются глобально и являются уникальными, могут быть 48 или 64 бита. Является расширением уникального идентификатора (Extended Unique Identifier, EUI-48/64). В качестве первых 24/28/36 бит данные адреса содержат уникальный идентификатор организации, выдаваемый регистрирующим органом IEEE в качестве идентификатора организации (OUI). Оставшиеся биты в наборе находятся в зоне ответственности конкретной организации, получившей данный OUI.

Идентификатор EUI также используется другими коммуникационными технологиями, например, Bluetooth Low Energy.

3. *Телефонные номера.* Стандарт ITU-T E.164 [16] определяет набор номеров для всемирной общественной телефонной сети (включая наземные линии и мобильные сети). Номера E.164 имеют максимум 15 символов. От одного до трёх первых символов могут быть кодом страны, выдаваемых Международным союзом электросвязи.

1.4.4. Стандарты идентификации пользователей

Форматы идентификаторов пользователя обычно определяются конкретной системой, к которой необходимо обеспечить доступ пользователям. Подобные идентификаторы могут предоставляться человеком (пользователем) и проверяться системой на уникальность или даже присваиваться самой системой. Зачастую в качестве идентификатора человека (пользователя) используется электронный адрес. Государственные организации чаще всего имеют собственные требования для идентификаторов человека и организации [36].

Примеры

1. *Адрес электронной почты*. Стандарт IETF RFC 5322 [29] определяет формат адреса электронной почты в Интернете. Он содержит строку, сопровождаемую символом принадлежности (“@”), сопровождаемый доменным именем.

2. *Идентификатор организации*. Стандарт ISO/IEC 6523-1 [23] определяет структуру для уникального определения организации и её подразделений. Обычно реализуется в иерархическом виде, начинается с идентификатора регистрирующей организации (максимум 4 символа), идентификатора организации (максимум 35 символов), выдаваемый регистрирующим органом, опциональный номер подразделения организации, в свою очередь выдаваемый самой организацией или третьими лицами, и опциональный индикатор источника подразделения организации (одна заглавная или строчная буква).

1.4.5. Стандарты идентификаторов данных

Существуют различные стандарты идентификаторов для наборов данных, файлов, потоковых данных, метаданных, типов данных и других элементов данных [37]. Некоторые стандартизированные решения предоставляют поддержку множества существующих идентификационных схем с целью предоставления схем, основанных на доменах и контексте.

Примеры

1. ***Идентификаторы метаданных.*** Стандарт ISO/IEC 11179-6 [18] описывает процедуры, по которым метаданным может быть назначен международный уникальный идентификатор, сопровождаемый регистрацией в реестре метаданных, обслуживаемом одной или более регистрирующей организацией. Он поддерживает множественные схемы идентификации и гарантирует уникальность идентификации путем определения пространства имен для каждой из схем. Данный стандарт не обязует использовать конкретные схемы, но предоставляет приложение, в котором описана структура для идентификатора в случае, если используемая схема идентификации определена стандартом ISO/IEC 6523-1 [22]. Подобный идентификатор является иерархически структурированным, содержит идентификатор регистрирующей организации, уникальный в пределах регистрирующей организации идентификатор, идентификатор версии для объекта данных.

2. ***Идентификатор данных (типов).*** Стандарт ISO/IEC 15418 [17] определяет использование идентификаторов приложений GS1 и идентификаторов данных ASC MH10 для целей идентификации кодированных данных. Данные идентификаторы являются буквенно-цифровыми префиксами, используются в носителях данных, подобных штрих-кодам и RFID-меткам, которые определяют значение и формат кодированных элементов данных (прим. номер товарной позиции, серийный номер, вес, дата производства) [32].

3. ***Унифицированный идентификатор ресурса URI.*** Стандарт IETF RFC 3968 [18] определяет синтаксис идентификаторов URI. URI используются для идентификации ресурсов, доступ к которым возможен через сеть, обычно

глобальную сеть Интернет. Подобные ресурсы обычно являются элементами данных (документы, программы, наборы данных) в различных форматах. URI поддерживают различные схемы идентификации путем указания идентификатора схемы в начале каждого URI. Примерами являются схемы типа «epc» для электронных кодов продуктов, обозначаемых как «urn:epc» [33].

4. ***Характеристики электрических товаров.*** Стандарт ISO 61360-1 [20] является основополагающим для явного и недвусмысленного описания характерных свойств всех элементов электротехнических систем, начиная от базовых компонентов, заканчивая системой в полусобранном и полностью собранном виде. Стандарт также является основой для словаря IEC Common Data Dictionary (IEC CDD) [24] – распространенное хранилище концептов для всех решений в сфере электротехники. Идентификатор типа элементов данных должен содержать комбинацию из шести символов под названием «Тип элемента данных», сопровождаемый дефисом, за которым следует комбинация из трёх символов, обозначающие номер версии типа элемента данных. В соответствии с стандартами ISO/IEC 6523-1 [30] и ISO 13584-26 [24], чтобы данный код был глобально уникальным, используется расширение данного кода при помощи идентификатора регистрирующего органа (прим. «0112/2///61360_4»).

5. Ключи базы данных. Универсальные уникальные идентификаторы (UUID) определены стандартом IETF RFC 4122 [26], зачастую используются в качестве уникальных ключей в базах данных. Размер UUID равен 128-ми битам и может быть локально сгенерирован без необходимости в централизованном органе администрирования. Генерация может происходить различными способами, включая псевдослучайную генерацию и генерацию при помощи алгоритмов на базе текущего времени, других локально уникальных идентификаторах или именах.

1.4.6. Стандарты идентификации местоположения

Идентификация местоположения является ключевой в приложениях Интернета вещей. Стандарты существуют в виде названия объектов и географических положений. Данная информация зачастую хранится в приложении

ИВ для отслеживания места происхождения события, для фиксации в памяти где объект (вещь) должна находиться или куда должна пойти [34].

Примеры

1. ***Стандарт ISO 6709*** [27] описывает представление географического местоположения через координаты, который включает в себя широту и долготу, используется при обмене данными. Также стандарт уточняет представление точек на плоскости при помощи координат других типов, отличных от широты и долготы. Он также конкретизирует представления высоты и глубины, которые могут быть ассоциированы с координатами на плоскости. Представление включает в себя единицы измерения и очередность координат.

2. ***Идентификатор местоположения международной организации воздушного транспорта (IATA)*** является уникальным трёхбуквенным кодом, используемый в авиации для идентификации местоположения аэропортов по всему миру. IATA также предоставляет коды для железнодорожных станций и обслуживающих аэропортов. Данные коды администрируются ассоциацией IATA и регулируются резолюцией IATA 763 [27].

3. ***Идентификатор местоположения международной системы классификации географических объектов*** [27] используется большинством транспортных компаний, экспедиторами и производственной индустрией по всему миру. Он также применяется национальными государствами, в торговой деятельности, используется Европейским Союзом для статистики, всемирным почтовым союзом для определенных почтовых услуг и др. Каждый элемент кода содержит пять символов, где два первых отображают страну (в соответствии со стандартом ISO-3166-1 [28]), а три оставшихся отображают имя места.

1.4.7. Стандарты идентификации протоколов

Подобно коммуникационным идентификаторам, идентификаторы протоколов обычно определяются в качестве части протокола, который его использует [35].

Примеры

1. *Поле Ethertype*. Стандарт IEEE 802.3 [13] определяет поле Ethertype в качестве двухбайтного значения, отображающего протокол MAC на стороне клиента. Значения поля Ethertype назначаются регистрирующим органом IEEE. Поле Ethertype передаётся в поле двойного назначения кадра Ethernet, предназначенного для отображения длины или типа. Если значения поля больше или равно десятичному числу 1536 (в шестнадцатеричном представлении 0600), то поле тип/длина отображает значение Ethertype [39].

2. *IPv6 следующий заголовок*. Стандарт RFC 8200 [19] определяет формат пакета IPv6. Поле «Следующий заголовок» является частью заголовка пакета и определяет тип заголовка, следующий непосредственно за заголовком IPv6. Он может также являться расширением заголовка или заголовком вышележащего протокола, подобно TCP, UDP или ICMP [95].

3. Идентификатор формата содержимого CoAP. Стандарт IETF RFC 7252 [41] определяет протокол CoAP. Идентификатор формата содержимого является опциональной частью протокола, отображающий формат представления полезной нагрузки сообщения. Представлен в виде численного идентификатора в диапазоне 0-65535. Используется в качестве краткой формы отображения типов интернет медиа-форматов, например, «text/plain», «application/XML» или «charset=utf-8».

Таким образом, из приведённого анализа различных систем идентификации можно сделать вывод, что для Интернета вещей необходима система идентификации, которая бы была совместима с существующими схемами идентификации и имела возможность подключения и учёта всего множества устройств и приложений ИВ. Это больше, чем просто идентификация людей и управление их доступом к различным типам данных (то есть, конфиденциальные данные, служебные данные, данные устройств и т. д.) [38; 98].

Идентификация объектов и ее услуг признаны одной из основных проблем на пути развития глобализации Интернета вещей. В диссертационной работе основной фокус сделан на исследование и модернизацию архитектуры цифровых объектов, предложив соответствующие методы и модели. Это позволит создать

основу идентификации устройств (объектов) в сетях Интернета вещей. Сравнение существующих систем идентификации, применяемых в настоящее время для Интернета вещей (DNS, EPC, ucode, HIP и др) и архитектуры цифровых объектов показало высокую эффективность и универсальность применения данной архитектуры для всех типов рассмотренных устройств интернета вещей [59]. Таким образом, можно предположить, что идентификация будущего ИВ будут основаны на DOA [40;46].

1.5. Аналитический обзор по исследованиям, проводимым в мире по идентификации Интернета вещей

Обзор публикаций показывает, что исследуемая проблема малоизучена ввиду ряда факторов. Во-первых, исследованием данной проблемы занимается узкий круг специалистов, которые недостаточно взаимодействуют между собой (возможно, это вызвано закрытостью значительной части проводимых исследований). Во-вторых, значительные исследовательские усилия были сосредоточены на легковесных протоколах связи и криптографических / традиционных механизмах, а метод идентификации, который объединяет все типы и требования идентификаторов практически нигде не рассматривается. В-третьих, в России и за рубежом уделяют большое внимание постановке натурных испытаний и получению практических результатов, однако при этом авторы неглубоко вникают в обмен служебными сообщениями между всеми компонентами различных систем идентификации, что порой, значительно загружает сеть [42]. Наблюдается стремление к количественным результатам, но не к качественному пониманию сути процессов.

Постараемся разобраться во всем подробнее, исходя из материалов исследований идентичности в интернете вещей доступных автору.

В статье [119] рассматривается понятие «Идентичность в Интернете вещей» и вводится аббревиатура IDoT. Кроме этого, проводится параллель и анализ на предмет: почему она настолько уникальна по сравнению с понятием «идентичность пользователей» (IDoU) в традиционных сетях и системах связи.

Используя идеи «Идентичности» пользователя (IDoU) из традиционных систем и сетей был предложен стек для «идентичности» в интернете вещей (Рисунок 1.3). В представленном информационном стеке имеются четыре категории: наследование, ассоциация, знания и контекст.

Рисунок 1.3 – Информационный стек для идентификации в IoT (IDoT)

Как отмечено самими авторами статьи, использование предложенного стека для определения IDIoT действительно является новой парадигмой по сравнению с IDoU. Из-за ограниченной доступности информации в средних категориях (т.е. «Ассоциации» и «Знания»), а также негибкости категории «Наследование» и неточности категории «Контекст», аутентификация на основе риска [16] с использованием нескольких факторы, безусловно, будут предпочтительным вариантом для Интернета вещей [99].

Помимо проблем, связанных с использованием нескольких факторов из предложенного стека для определения и построения IDoT, в ИВ есть как минимум две дополнительные проблемы, которые еще больше усложняют управление IDoT. Первая проблема связана с отношением владения и идентификации пользователя объекта ИВ. В любой момент времени *t* каждый объект ИВ должен иметь владельца (одного или нескольких пользователей). Вторая проблема связана с управлением идентификаторами и пространством имен объектов ИВ. Каждый ресурс имеет URI (унифицированный идентификатор ресурса) в Интернете. Существует также DNS

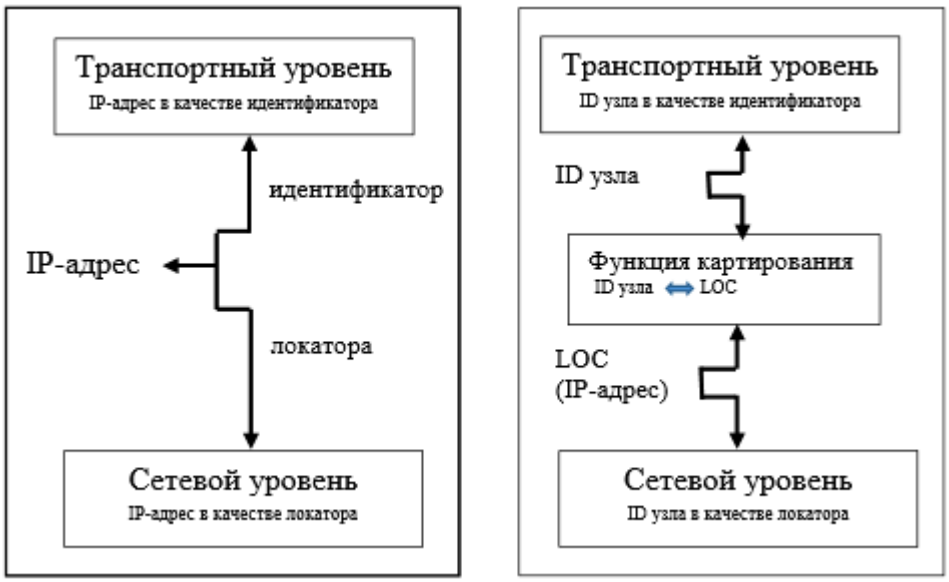
(система доменных имен), которая отображает URI на свой текущий IP-адрес ресурса; и этот DNS находится в ведении организации Internet Assigned Numbers Authority (IANA) [141]. С помощью этого пространства имен и структуры сопоставления идентификаторов динамика идентификаторов, таких как IP-адрес URI, может быть скрыта, и связь между URI становится намного проще.

Установлено, что одной из основных проблем безопасности в ИВ является отсутствие строгого понятия «идентичность» в Интернете вещей (IDIoT), а предложенный стек позволяет сосредоточить внимание на ситуативной информации, которая, как ожидается, будет неточной и «зашумленной».

В работе [43] рассмотрен проект «IDENTITY/IDENTIFIER-ENABLED NETWORKS IDEAS», который призван служить механизмом или плоскостью управления общим идентификатором (Identity and Identifier) для будущих сетей, которые могут быть адаптированы для Интернета вещей в нескольких измерениях. На Рисунке 1.4 представлено взаимодействие ИВ на базе различных технологий передачи данных с доменами идентичности/идентификации.

Рисунок 1.4 – Взаимодействие ИВ с доменом Идентификации/Идентификаторами

В статье [92] представлены исследования, базирующиеся на новых схемах ILS (Identifier and Locator Split). Рисунок 1.5 иллюстрирует абстрактную концепцию традиционного идентификатора (IDf) и разделение локатора [43].



а) б)

Примечание – а) традиционная архитектура IP, б) архитектура разделения ID/LOC

Рисунок 1.5 – Пример разделения идентификатора и локатора (ITU-T Y.2015)

По утверждению авторов, во всех существующих схемах ILS идентичность явно не связана с соответствующим идентификатором (-ами), в то время как идентичность вещи (или объекта) может фактически выполнять многие функции в будущей сетевой архитектуре, обеспечивая возможность разделения идентификатора (IIS). Поэтому авторы предложили новую схему IIS вместе с парадигмой ILS, которая продвигается в рамках единой структуры с различными потенциальными услугами с добавленной стоимостью.

Проведенный анализ результатов российских и зарубежных исследований, доступных автору, показал, что существуют недостатки, представленных в работах авторов, среди которых можно выделить:

– для идентификации цифровых объектов на данный момент есть два типа уникальных глобальных идентификаторов, которые считаются критически важными Интернет-ресурсами. Это IP-адрес и доменное имя, но вопрос: кто их контролирует и как?

– все механизмы и методы идентификации цифровых объектов, которые были исследованы привязаны к местоположению. Что происходит, когда объект перемещается? Теряется ли уникальность совокупности метаданных об идентичности того или иного объекта?

– большинство методов идентификации в Интернете вещей не обеспечивают уникальность и несменяемость присвоенного ранее адреса;

– способность поддерживать идентификаторы с большим объемом метаданных о уникальных параметрах цифровых объектов – каким образом обеспечить для простых устройств интернета вещей?

– отсутствие методов, которые поддерживают все типов языков и имеют децентрализацию систем регистрации цифровых объектов в Интернете.

Из материалов проведенного обзора можно сделать заключение, что вопрос идентификации в Интернете вещей очень уязвим. С одной стороны, в настоящее время всё как-то работает, но если задуматься о перспективе на горизонте 10–15 лет, то необходимо уже сейчас найти методы и модели, которые позволят управлять огромным объемом информации, содержащейся в более чем 9 миллиардах подключенных устройств [117; 87].

В настоящее время отсутствуют исследования, в которых была бы подробно представлена архитектура цифровых объектов как новый механизм для идентификации устройств и приложений интернета вещей. Работы по стандартизации методов идентификации устройств и борьбы с контрафактом на базе архитектуры цифровых объектов в настоящее время ведутся в исследовательской комиссии МСЭ-Т. Так, в декабре 2018 г. на процедуру согласия была представлена Рекомендация «Архитектура взаимодействия устройств интернета вещей на базе архитектуры цифровых объектов», а в 2019 году планируется принятие Рекомендацию «Структура решений по борьбе с контрафактными устройствами интернета вещей на базе архитектуры цифровых объектов».

Проведенный анализ показал, что перспектива повсеместного применения архитектуры цифровых объектов создаст уникальные условия для транснациональной единой системы идентификации, которую уже сегодня необходимо внедрять во вновь создаваемые устройства и приложения Интернета вещей.

1.6. Общая концепция архитектуры цифровых объектов

Как было показано в предыдущих параграфах, существующие системы идентификации и управления информацией в сети основаны на классической клиент-серверной архитектуре. Сервер в такой системе представляется местом хранения информации и обработки запросов от клиентов на работу с данной информацией. DOA, в отличии от такого подхода стремится решить вопрос не о локализации, а о контексте цифрового объекта [1, 125].

Цифровой объект в данной архитектуре характеризуется не только информацией о своем расположении. Помимо этого, существует возможность получать различные сведения о самом объекте: требования к доступу, аутентификации, информацию об авторе и прочее [9]. Вся эта информация вносится самим создателем цифрового объекта. Для этого в архитектуру DOA интегрирована специальная инфраструктура, обеспечивающая необходимое шифрование и верификацию доступа.

Основными структурными элементами DOA являются цифровой объект, система резолюции идентификатора (Handle System) и репозиторий и реестр цифровых объектов. Остановимся на принципах система резолюции подробнее.

Каждому цифровому объекту в описываемой архитектуре ставится в соответствие уникальный идентификатор – DOI (*от англ.* Digital Object Identificator). Данный идентификатор чем-то напоминает URL, на базе которого построен современный Интернет. Однако, в отличии от последнего, присваиваемые идентификаторы остаются постоянными и не зависят от состояния цифрового объекта. Именно система резолюции связывает идентификатор с информацией о текущем статусе цифрового объекта (местонахождение, доступ, информация об аутентичности).

В классической архитектуре DOA система резолюции является двухуровневой [42; 43]. Первым уровнем резолюции является глобальный реестр (GHR, *от англ.* Global Handle Registry); вторым уровнем – набор локальных реестров (LHR, *от англ.* Local Handle Registry) или локальных сервисов (LHS, *от англ.* Local Handle Service). Для разрешения идентификатора в данной подсистеме,

вначале идет обращение к глобальному реестру GHR, который сообщает информацию о локальном реестре LHR, в котором содержится необходимая информация о цифровом объекте. Схематически данный процесс представлен на Рисунке 1.6.

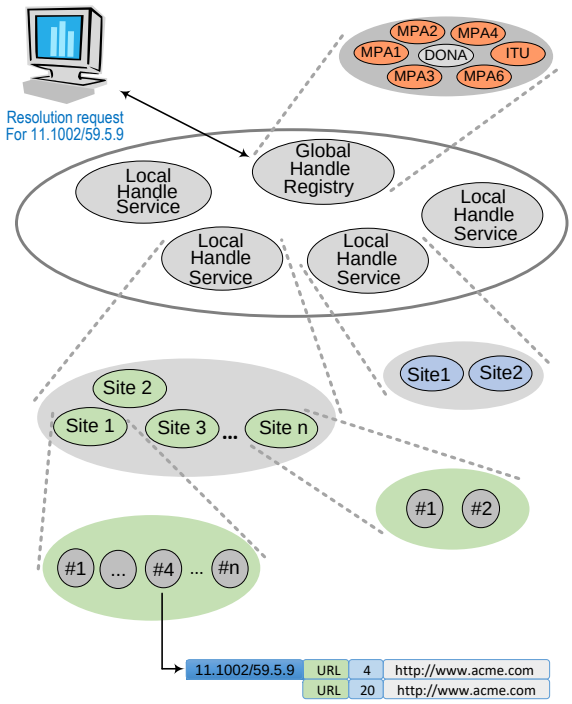


Рисунок 1.6 – Структура системы резолюции (Handle system)

Сама структура идентификатора DOA также соответствует двухуровневой системе. Например, рассмотрим идентификатор: 10.1000/123abc. Первая часть, расположенная до «/», носит названия префикса; вторая часть – суффикса. Префикс позволяет установить сведения о локальном реестре цифрового объекта LHR. Данное соответствие префикса и информации об администраторе хранится в глобальном реестре GHR. Суффикс же уже однозначно идентифицирует конкретный объект, и данная информация, связывающая суффикс с конкретным объектом хранится в локальном реестре LHR [82;128].

Выводы по главе 1

1. В первой главе были проанализированы различные системы идентификации, их архитектура, структура идентификаторов и примеры их использования в повседневной жизни.

2. Проведенный обзор международной деятельности по исследованиям идентификации в концепции Интернета вещей показал, что в настоящее время отсутствуют прикладные исследования, посвященные идентификации устройств и приложений Интернета вещей на базе архитектуры цифровых объектов. В связи с этим, в диссертационной работе особое внимание насущно необходимо уделить разработке методов и моделей идентификации устройств и приложений Интернета вещей.

3. В аналитическом обзоре показано, что до настоящего времени отсутствовали работы, в которых была бы подробно проанализирована и исследована архитектура цифровых объектов как метод идентификации устройств и приложений Интернета вещей.

4. Технология DOA позволяет осуществлять однозначную персистентную идентификацию объектов, в которой заинтересованы правообладатели этих объектов. Это делают целесообразным развитие применения технологии DOA как транснациональную систему идентификации с равными правами для всех членов.

Глава 2. МЕТОД ПОСТРОЕНИЯ СЕТЕВОЙ АРХИТЕКТУРЫ ЦИФРОВЫХ ОБЪЕКТОВ ЗА СЧЕТ ВВЕДЕНИЯ ПРОМЕЖУТОЧНОГО УРОВНЯ ВЗАИМОДЕЙСТВИЯ

В данном разделе будет проведен анализ построения сетевой архитектуры цифровых объектов и предложены методы модернизации архитектуры построения DOA. Модернизация будет заключаться в введении дополнительного уровня взаимодействия с целью уменьшения сетевой задержки при разрешении запросов.

2.1. Анализ системы идентификации архитектуры цифровых объектов

Архитектура цифровых объектов (Digital Object Architecture – DOA) и связанная с ней система резолюций Handle System были разработаны корпорацией национальных исследовательских инициатив (CNRI) в начале 1990-х годов, основываясь на работах над цифровыми библиотеками для Управления перспективных исследовательских проектов Министерства обороны США (DARPA)[133]. Одним из первоначальных мотивов создания DOA была необходимость идентификации и получения информации об объекте в течение длительного периода времени (порядка десятков или сотен лет).

Разработка архитектуры цифровых объектов стала попыткой перехода от представления данных в Интернете с помощью наборов узлов и транспорта к обнаружению и доставке информации в виде цифровых объектов [48].

Цель создания архитектуры цифровых объектов – решение следующих проблем управления цифровой информацией:

– обеспечение стандартного доступа к разрозненной информации (идентификация, поиск информации и предоставление данных, обеспечение безопасности, типизация);

– взаимодействие с разнообразными информационными системами;

– независимость от конкретных базовых технологий, которые используются для размещения и обслуживания информации;

– взаимодействие в течение длительных периодов времени;

– активное управление системами, на которых распространяется информация;

– обеспечение большого уровня масштабируемости;

– распределенная архитектура;

– открытая архитектура;

– стандартные протоколы и процедуры взаимодействия компонентов системы.

Архитектура цифровых объектов – архитектура распределенной системы хранения, определения местоположения и поиска информации в Интернете[45]. К фундаментальным компонентам архитектуры цифровых объектов относятся отображенные на Рисунке 2.1.

Рисунок 2.1 – Фундаментальные компоненты архитектуры цифровых объектов

Рассмотрим более подробно каждый из компонентов DOA.

1. Цифровые объекты

Каждый объект в целом обладает набором признаков, определяющих его сущность и, благодаря этому, выделяющих его из множества других. Таким образом, различные признаки будут являться своего рода уникальными идентификаторами.

Идентификация необходима для решения таких задач, как:

– однозначное определение объекта;

– распознавание объекта по его свойствам;

– группирование объектов по определенным признакам;

– выделение объекта из множества подобных.

Согласно Рекомендации МСЭ-Т X.1255 Цифровая объект – это «структура обнаружения информации по управлению определением идентичности», общепринятая структура данных, состоящая из одного или нескольких элементов, благодаря которой обеспечивается функциональная совместимость информационных систем в Интернете.

По факту, цифровой объект – это объект, состоящий из структурированной последовательности бит, имеющий название, уникальный идентификатор и атрибуты, описывающие его свойства. Однако, в идентификации цифровых объектов возникает сложность, т.к. человек воспринимает не сами биты, а их отображение с помощью программного обеспечения. Таким образом, чем больше в этом процессе задействуется сенсорный аппарат человека, тем субъективнее вопрос, что именно считать сущностью цифрового объекта и рассматривать способы его идентификации.

В контексте DOA, цифровой объект – данные, которые не зависят от платформы. Для управления цифровыми объектами используются три архитектурных компонента. Каждый из компонентов может использоваться самостоятельно, но в комбинации они обеспечивают распределенную и масштабируемую систему управления информацией в Интернете. Таковыми компонентами являются [136]:

1) масштабируемая и распределенная система идентификаторов и резолюций цифровых объектов;

2) репозитории доступа и управления цифровыми объектами;

3) реестры для поиска и обнаружения объектов.

Система резолюции связывает идентификаторы с информацией о состоянии цифровых объектов. К примеру, такая информация может содержать местонахождение данного объекта в Интернете или требования к доступу, информацию об аутентификации и т.п. Создатель объекта или авторизованный администратор предоставляет эту информацию с использованием инфраструктуры публичных ключей, которая интегрирована в DOA. Технология публичных ключей предполагает использование двух ключей для шифрования – публичного и частного.

Цифровые объекты – ключевой элемент, вокруг которого выстроены другие компоненты и сервисы. Цифровые объекты не заменяют существующие форматы и структуры данных, но обеспечивают общепринятые способы представления этих форматов и структур. Это позволяет их однозначно интерпретировать. и перемещать между различными гетерогенными информационными системами в ходе изменений в системах по прошествии времени [48].

В контексте Рекомендаций МСЭ-Т объектами могут быть сети, сервисы, документы, права доступа, транспортная информация, устройства или отдельные чипы, авторские произведения или любая другая информация, представленная в виде структуры данных и связанных с ними метаданных, то есть в виде цифрового объекта. Таким образом, в качестве цифрового объекта можно представить и любой объект реального мира.

Из нескольких рассмотренных выше определений можно сделать вывод что цифровой объект – структурированная запись, содержащая данные, информацию о состоянии данных и метаданных. Цифровой объект может содержать указатели на места, где может быть найдена соответствующая информация. Все цифровые объекты доступны с использованием протокола цифровых объектов, независимо от основных технических систем. Каждый цифровой объект описывает себя и свое

содержимое. Каждый цифровой объект содержит в себе описание своих собственных правил управления доступом.

2. Репозитории

Репозитории цифровых объектов (DO Repository) используется для хранения и доступа к цифровым объектам. Количество репозиториев в системе не ограничено. Идентификаторы – набор идентификаторов, называемых хэнделами, для цифровых объектов, которые являются уникальными, постоянными и независимыми от базовой физической или логической системы [49].

3. Реестр объектов

Реестры определяют коллекции объектов, доступных в репозиториях. DO Registry используется для определения коллекций цифровых объектов и регистрации объектов, которые могут располагаться в одном или нескольких открытых репозиториях. Реестр также используется для извлечения сведений о ранее зарегистрированном объекте (место, свойства, авторы, обладатели прав и т.п.) по его идентификатору. Хотя первоначально DOA была создана так, что позволяла использовать различные системы идентификации, однако со временем практически полностью был осуществлен переход к использованию системы хэнделов.

4. Система резолюций

Система резолюций (Handle System) – это система, используемая для выделения хэнделов в информации и сведений о местоположении информации. Остановимся на ее принципах функционирования более подробнее [50; 51].

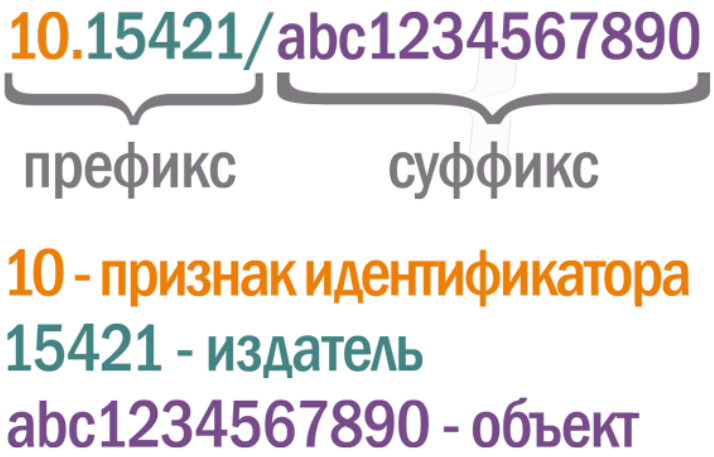
2.1.1. Система резолюция

Система резолюции (Handle System) была создана, чтобы преодолеть ограничения функциональности существующих систем идентификации объектов в Интернете.

Резолюция – это процесс, в котором идентификатор является запросом к сетевому сервису на получение актуальной информации (данных о состоянии), относящейся к определяемой сущности (чаще всего речь идет о местоположении).

Система резолюции поддерживает множественную резолюцию, т.е. ответом на запрос может быть местоположение различных экземпляров объекта, связанные сервисы и любая другая информация, указанная в метаданных объекта. Возвращаемая информация необязательно должна указывать на экземпляр объекта: например, это может быть описание или состояние объекта, некие индикаторы или измерения, отношения с другими сущностями и т.д.

Хэндел – глобально уникальный и разрешимый идентификатор. Он представляется следующим образом [52; 60; 61]: «префикс/суффикс», где префикс уникален в пределах Системы резолюции. Пример хэндела изображен на Рисунке 2.2.



Примечание – «10» – признак идентификатора – всегда остается неизменным; «15421» – уникальное цифровое обозначение издателя. Каждый издатель (например, издательские дом, отдельное учреждение, отдельный человек) должен иметь свое уникальное обозначение (префикс), которое можно получить у одного из официальных агентств по регистрации DOI или от сторонних лиц, которые имеют действующий контракт с данными агентствами; «abc1234567890» – (суффикс) – непосредственно идентификатор конкретного объекта, например, научной статьи или сборника. Суффикс тоже должен быть уникальным – повторения исключены. Суффикс может содержать как цифры и буквы, так и отдельные знаки

Рисунок 2.2 – Пример хэндела

В настоящее время префикс может состоять только из числовых значений, тогда как для суффикса ограничений нет. Ограничений по длине имени не существует [129].

Префикс обработки структурирован иерархически. Менеджер префиксов может выделять дочерним организациям вспомогательные префиксы, разделенные точкой. Это похоже на систему доменных имен, но иерархия системы Handle пишется слева направо (xxx.yyy), тогда как иерархия DNS записывается справа

налево (yyy.xxx), где xxx – верхний уровень. По префиксу можно однозначно установить организацию, занятую его обслуживанием (администратора).

Суффикс однозначно идентифицирует конкретный объект.

Система резолюции предоставляет доступ клиента к местоположению цифрового объекта. Для этого используется иерархическая модель обслуживания, состоящая из глобального регистратора хэнделов (GHR) и локальной системы обработки хэнделов (LHS). Каждая служба локальной дескрипции может содержать свою собственную иерархию хэндел сервисов: GHR содержит информацию о сопоставлении префикса хэндела для LHS, который обслуживает хэндел для данного префикса [53].

На Рисунке 2.3 изображен пример идентификатора «bar.foo/1234». Префикс верхнего уровня «bar», служебная информация которого содержится в LHS A.

Клиент отправляет запрос Handle в GHR для 0.NA/bar.foo. GHR возвращает служебную информацию для 0.NA/bar.foo, указывающую на LHS A управляющую префиксом идентификатора. Клиент запрашивает LHR A для bar.foo/1234. LHS A идентифицирует сервер для bar.foo/1234. В этом примере он обращается к цифровому объекту и возвращает запрашиваемую информацию клиенту. После чего клиент обрабатывает возвращенную информацию соответствующим образом.

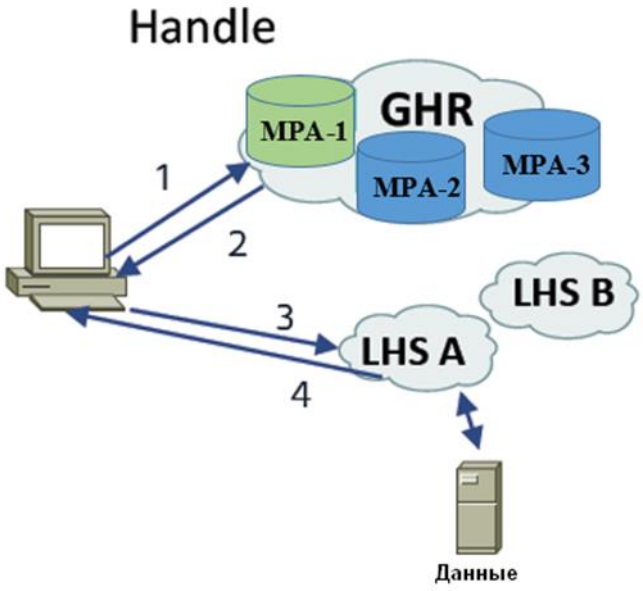


Рисунок 2.3 – Резолюция идентификаторов

GHR отвечает за управление корнем иерархии дескрипторов системы, выделяя уникальные префиксы и предоставляя глобальную службу привязки префиксов к LHS этого префикса.

Каждому объекту в системе резолюции приписан ряд обязательных атрибутов:

– числовой идентификатор данных (индекс);

– тип данных. Тип данных представляет собой ссылку на запись в системе резолюции;

– время последнего изменения данных;

– время жизни записей в системе TTL (Time To Live);

– права доступа для администратора записи и для не авторизованного пользователя;

– ссылка на администратора данной записи об объекте;

– ссылки на другие хэнделы;

– данные.

В протоколе HSP – Handle System Protocol предусмотрен ряд возможностей для уточнения получаемого набора данных:

– выбор типа возвращаемых данных;

– выбор индекса или диапазона индексов;

– указание на то, можно ли ответить кешированными данными.

Таким образом, система Handle System:

– обеспечивает базовую систему идентификаторов в Интернет;

– идентификатор содержит данные о текущем состоянии объекта;

– идентификатор сохраняется, даже если местоположение и другие атрибуты объекта изменяются;

– обладает высокой масштабируемостью;

– связывает одно или несколько типизированных значений, например, IP-адрес, открытый ключ, URL-адрес каждого идентификатора;

– обладает открытым, четко определенный протоколом взаимодействия и моделью данных;

– обеспечивает инфраструктуру для доменов приложений, например, цифровых библиотек и публикаций, электронных исследований и т.д.

Основными качествами, заложенными при ее создании, были [72;73]:

1) уникальность – каждый хэндел уникален в рамках глобальной системы;

2) постоянство – хэнделы могут использоваться в качестве постоянных идентификаторов для объектов в Интернете. При этом хэндел не зависит от объекта, который он именует, их единственная связь – в самой системе;

3) множественные экземпляры – хэндел может указывать на различные экземпляры ресурса, расположенные по различным сетевым адресам;

4) множественные атрибуты – хэндел может указывать на различные атрибуты ресурса, включая связанные сервисы, расположенные по различным сетевым адресам;

5) расширяемое пространство имен – локальное пространство имен можно присоединить к глобальному, получив статус регистратора и уникальный префикс, чтобы избежать конфликтов с существующими именами;

6) модель безопасности – система резолюции позволяет осуществлять безопасную резолюцию и администрирование;

7) распределенный административный сервис – для каждого хэндела в системе можно определить собственного администратора (владельца);

8) эффективный сервис резолюции – протокол системы спроектирован с учетом множественных одновременных обращений.

Возвращаемая информация не обязательно должна указывать собственно на экземпляр объекта: например, это может быть описание или состояние объекта, некие индикаторы или измерения, отношения с другими сущностями и т.д.

Для обработки запроса через протокол http система использует прокси-серверы, которые обрабатывают хэнделы, представленные в виде URL. Варианты резолюции при этом хранятся в хэндл-записи в виде xml-файла. Таким образом, прокси-серверы, являются надстройкой над системой резолюции.

2.1.2. Модель данных

Объекты в архитектуре DOA описываются как цифровые. Лимита на длину самого имени или элементов суффикса, или префикса не существует. Имена могут содержать любые печатные Unicode-символы и не зависят от регистра. Комбинация уникального префикса, а также уникального суффикса, выдаваемого конкретным регистратором, очевидно, уникальна сама по себе, что и позволяет децентрализовано регистрировать идентификаторы [55].

Хотя не рекомендуется содержать значимые данные в именах, необходимо указывать их в метаданных, суффиксы часто применяют для включения идентификатора из другой системы, используемой для идентификации объекта.

Объект однозначно описывается метаданными, основанными на расширяемой модели данных. При назначение имени требуется заполнение минимального набора метаданных, это необходимо для того, чтобы отличить объект от других в системе [56].

Набор метаданных может расширяться регистрационными агентствами включением элементов, необходимых для работы соответствующих приложений.

2.1.3. Идентификатор цифровых объектов

Идентификатор цифровых объектов (Digital Object Identifier – DOI) – цифровой идентификатор объекта, частный случай системы идентификации DOA.

Создание системы DOI стало результатом инициативы нескольких издателей в 1990-х годах. Они заявили о необходимости создания уникальной системы идентификации контента, вместо того, чтобы ссылаться на него по его местоположению [74].

В результате в 1998 году была основана компания International DOI Foundation для разработки системы на основе существующих технологий и стандартов.

Наиболее известной сферой применения DOI является сервис Cross-ref – сервис цитирования, который позволяет ученым ссылаться в работах непосредственно на источник цитаты, вне зависимости от издателя.

Синтаксис DOI в 2010 году стал международным стандартом ISO 26324. DOI является зарегистрированной частью схемы URI (RFC 4452), также идентификаторы DOI можно записывать как URL, резолюция при этом будет использовать сервисы HTTP Proxy.

Модель данных в DOI. В системе DOI идентификатор может быть назначен любой объекту: физическому, цифровому или абстрактному. В архитектуре же DOA объекты описываются как цифровые. Между двумя подходами нет конфликта, так как любая сущность может рассматриваться с точки зрения ее цифрового представления [57]. Синтаксис идентификаторов DOA и DOI одинаковый. Имена могут содержать любые печатные Unicode-символы и не зависят от регистра [76; 131].

Описание метаданных системы DOI. Основные элементы метаданных и их структура представлены в Таблице 2.1.

Таблица 2.1 – Описательные элементы метаданных и их структура

Элемент	*Кол-во запросов*	*Описание*
DOI name	1	Идентификатор DOI
referentIdentifier(s)	0-n	Идентификаторы из других систем идентификации (ISAN, ISBN, ISRC, ISSN, ISTC, ISNI...). Для некоторых категорий идентифицируемых объектов это открытый список
referentName(s)	0-n	Наименование идентифицируемого объекта. В этом элементе присутствует указание на язык согласно стандарту ISO 639-2 и на категорию в зависимости от природы идентифицируемого объекта
primaryReferentType	1	Основная категория объекта. Открытый список. Среди стандартных допустимых значений: creation, party, event
structuralType	1	Структурная категория объекта. Доступные варианты зависят от значения основной категории объекта. Для стандартных категорий эти списки ограничены. Так, для категории creation возможны 4 значения: physical, digital, performance, abstraction. Для категории party допу
Mode	0-n	Этот атрибут представляет собой закрытый список форм восприятия, на которые рассчитан идентифицируемый объект. Атрибут предназначен только для категории creation. Допустимые значения: audio, visual, tangible, olfactory, tasteable, none
Character	0-n	Этот атрибут представляет собой закрытый список форм, в которых выражен контент идентифицируемого объекта. Атрибут предназначен только для категории creation. Допустимые значения: music, language, image, other

Продолжение таблицы 2.1

Элемент	*Кол-во запросов*	*Описание*
referentType	0-n	Роль по отношению к связанному объекту. Это открытый список со значениями, зависящими от основной категории объекта. Так, для типа party возможны, например, следующие значения, представленные атрибутом associatedPartyRole: Composer, Author, BookPublisher, JournalPublisher. Для типа creation обычно указывается формат, в котором представлен объект, описываемый атрибутом creationType. Например, возможны значения: audio file, scientific journal, musical composition, dataset, serial article, eBook, PDF
linkedCreation	0-n	Производные от оригинальных объектов. Этот элемент предназначен только для категории creation. В этом элементе могут быть указан атрибут creationRoleToCreation, характеризующий отношение производного объекта к оригинальному (например, экранизация книги)
linkedParty	0-n	Это элемент, применяемый только для категории party, и описывающий производные объекты категории party. В этом элементе может быть указан атрибут partyRoleToParty, уточняющий категорию производного элемента по отношению к исходному
principalAgent	0-n	Это элемент, применяемый только для категории creation. Он предназначен для перечисления сторон, связанных с созданием или публикацией объекта. Это открытый список. Этот элемент содержит атрибут agentRole, специфицирующий конкретную роль стороны по отношению к объекту (например: Creator, Author, BookPublisher)
dateOfBirthOrFormation	0-1	Это элемент, применяемый только для категории party и содержащий дату рождения (для человека или животного) или создания (для организации)
dateOfDeathOrDissolution	0-1	Это элемент, применяемый только для категории party и содержащий дату смерти (для человека или животного) или прекращения существования (для организации)
associatedTerritory	0-n	Это элемент, применяемый только для категории party, и содержащий список территорий, с которыми ассоциирован данный объект, согласно стандарту ISO 3166a2

2.2. Протоколы сигнализации в архитектуре DOA

Архитектура DO определяет два основных протокола и три основных компонент между которыми происходит обмен служебными сообщениями. Как уже отмечалось, тремя компонентами являются система резолюции, система репозиториев и система реестра [57]. На практике компоненты репозитория и реестра являются модульными и могут при необходимости комбинироваться.

Первый служебный протокол DOA, который будет рассмотрен ниже является протоколом идентификатора/резолюции Identifier/Resolution Protocol (IRP). В более ранней версии протокол идентификатора/резолюции назывался как протокол

системы резолюции, который использовался для создания, обновления, удаления и разрешения идентификаторов цифровых объектов. Как указано в описании протокола IRP, каждый идентификатор непосредственно связан с записью идентификатора, содержащей актуальную «информацию о состоянии», к которой могут обращаться клиенты. Все идентификаторы имеют вид префикса/суффикса, где по умолчанию префикс может сначала быть разрешен для определения местоположения конкретного сервиса идентификатора/разрешения, а суффикс может быть любой битовой последовательностью. Таким образом, любая организация может запустить систему резолюций для своего собственного набора идентификаторов, присвоив ей префикс, а также любой существующий идентификатор может быть преобразован в идентификатор цифрового объекта, обработав его как суффикс и добавив свой выделенный префикс.

Второй протокол DOA, который был анонсирован в конце 2018 года – это протокол интерфейса цифровых объектов (DOIP) [88]. Согласно технической документации о порядке взаимодействия архитектуры цифровых объектов этот протокол определен для использования сервисами цифровых объектов при которых системы репозитория и реестра представлена конкретными экземплярами. Более ранняя версия этого протокола, основанная на протоколе доступа к репозиторию (Repository Access Protocol – RAP) первоначально была описана выложена в открытый доступ Робертом Каном и Р. Виленски в 2009 году.

Протокол интерфейса цифровых объектов (DOIP) вер. 2.0 определяет стандартный способ взаимодействия клиентов с цифровыми объектами (DO)[94]. Предполагается, что такими DO управляют сервисы DO, которые часто называются сервисами DOIP, реализация протокола является частью этих сервисов. В этом контексте сама услуга DOIP считается цифровым объектом. Протокол предназначен для обеспечения взаимодействия между одним или несколькими объектами между которыми поддерживается взаимодействие на базе этого протокола, что обеспечивает поддержку конкретной формы межпроцессного взаимодействия в сетевой среде.

DOIP использует IRP для связи идентификаторов с различными элементами протокола. Максимальный размер идентификатора будет меняться со временем, но изначально максимальный размер идентификаторов, указанный в DOIP, составляет 4096 битов.

DOIP позволяет обеспечить безопасность с использованием инфраструктуры открытых ключей (Public Key Infrastructure – PKI) для проверки цифровых объектов, в том числе для аутентификации службы/клиента, а также для обеспечения целостности через подписи [62; 90]. Встроенная поддержка PKI также поможет клиентам и службам использовать шифрование. Протоколом предполагается управление доступом для DO с использованием идентификаторов для определения утвержденного списка контроля доступа и тестового ответа на запрос PKI. В описании определен набор основных операций, которые клиенты могут вызывать в сервисах и протокол по своей природе поддерживает добавление новых операций к уже имеющимся операций [126].

DOIP может быть туннелирован через любой безопасный протокол связи, а сам DOIP может использоваться для определения выбора такого протокола. Минимальным требованием является поддержка TLS для сетевых коммуникаций [132]. В дополнение к транспортной безопасности, протоколом также используется ряд других специфических функций:

1) реализация при которой возможна поддержка формата JSON [63;119];

2) возможность аутентификации системных ресурсов с использованием идентификаторов цифровых объектов;

3) в параметрах каждого DO должен быть указан тип. Типы являются расширяемыми, что позволяет создавать новые типы. Одна важная функция типов состоит в том, чтобы позволить службе DOIP идентифицировать допустимые операции. Типы – это назначенные идентификаторы и поэтому каждый тип связан с записью идентификатора, доступ к которой можно получить с помощью IRP [64]. Семантическая и другие особенности структурирования записей типов не указываются в описании DOIP. Предполагается, что группы или организации,

обладающие знаниями в данной области, возьмут на себя ответственность за создание типов в своем домене и за определение семантики и сериализации записей типов[51].

2.3. Представление системы идентификации на базе архитектуры цифровых объектов

В настоящее время в результате быстрого развития информационных технологий, роста объемов различной информации в сети связи общего пользования (ССОП)[58], а также из-за повсеместного внедрения технологий интернета вещей появилась насущная необходимость внедрения механизмов однозначной идентификации устройств и приложений интернета вещей, позволяющих отслеживать достоверность информации в сети и бороться с контрафактной ИКТ-продукцией [65; 66]. Предложения по созданию такого реестра объектов выдвигаются Министерством цифрового развития, связи и массовых коммуникаций Российской Федерации, правообладателями и обществами по управлению коллективными правами [67].

Для разработки такого сервиса сначала необходимо выбрать наиболее оптимальную систему идентификации [68]. Для идентификации можно использовать множество различных программных и аппаратных решений, например, системы аппаратной идентификации IPv6, связку IPv4+MAC, IMEI и др [114]. Однако общими недостатками этих систем являются возможность программного и аппаратного изменения идентификатора сетевого интерфейса и привязка к аппаратным идентификаторам, которая исключает возможность идентификации цифрового контента, что относится к виртуальным сущностям интернета вещей и тоже требует идентификации. Этих недостатков лишены альтернативные программные решения для идентификации, такие как DOA, URI, XRI, IRI и др., которые позволяют идентифицировать любой виртуальный или реальный объект в ССОП, независимо от наличия или отсутствия у него сетевого интерфейса [70; 71]. Эти системы так же, как и системы аппаратной идентификации, используют для аутентификации физических и цифровых объектов различные сторонние технологии [91,99]. Кроме того, необходимо

отметить, что не все существующие системы идентификации объектов отвечают развитию сетей связи в рамках концепции Интернета вещей [75].

Выбор оптимальной системы определяется следующими требованиями к технологиям идентификации, учитывающими ее использование в ССОП:

– системы идентификации должны отвечать на множественные запросы;

– для работы с идентификаторами необходимо реализовать различные уровни доступа, т.е. систему авторизации пользователей;

– база, содержащая данные, должна быть отделена от самого объекта идентификации;

– идентификаторы не должны содержать динамические элементы или метаданные.

На Рисунке 2.4 представлена общая архитектура для системы резолюции Handle System.

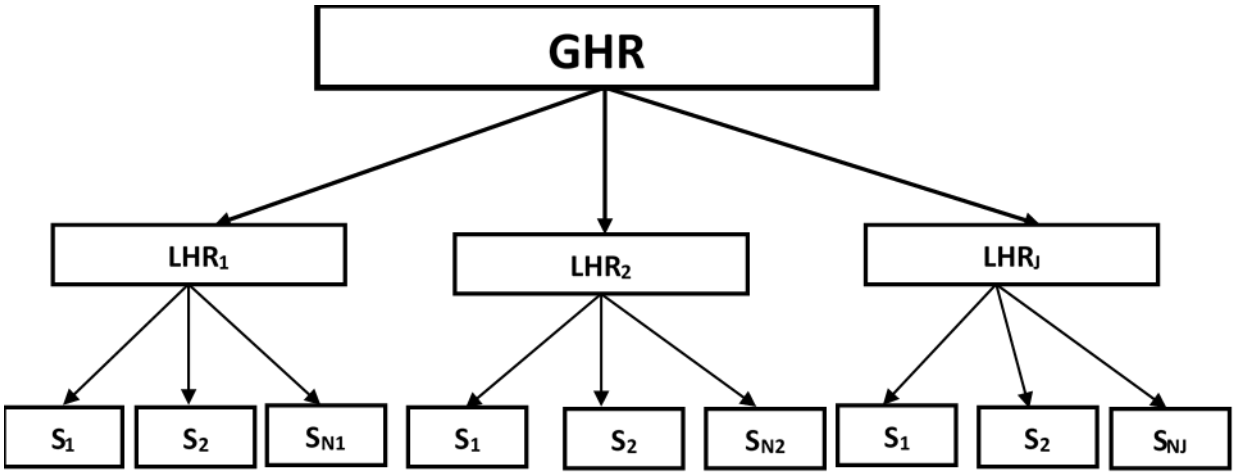


Рисунок 2.4 – Общая архитектура для системы резолюции Handle System

Подробное описание функционирование этой системы было представлено выше, однако, остановимся на ключевых этапах разрешения идентификатора в рамках архитектуры DOA. Клиент отправляет запрос Handle в GHR. GHR возвращает служебную информацию, указывающую на систему LHR, управляющую префиксом идентификатора. Клиент запрашивает LHR. LHR идентифицирует сервер, в результате чего происходит обращение к цифровому объекту, а в качестве ответа возвращается запрашиваемая клиентом информация.

После этого клиент обрабатывает полученную информацию. GHR отвечает за управление корнем иерархии дескрипторов системы, выделяя уникальные префиксы и предоставляя глобальную службу привязки префиксов к LHR [132].

Проходящая сейчас реорганизация модели управления DOA предполагает переход от модели с одним главным администратором GHR (до недавних пор им была CNRI) к модели с несколькими администраторами верхнего уровня MPA (Multi-Primary Administrator – многоцелевой первичный администратор сети цифровых идентификаторов), которых авторизует и чью деятельность координирует некоммерческая организация The DONA Foundation, зарегистрированная в 2014 г. в Женеве (Швейцария).

В сентябре 2018 г. были подписаны документы о создании национального MPA Российской Федерации, что обусловлено растущим интересом к построению среды, основанной на реализации концепции Интернета вещей. При этом ПАО «Ростелеком» были присвоены функции многоцелевого первичного администратора сети цифровых идентификаторов. Работы по стандартизации методов идентификации и борьбы с контрафактом на базе архитектуры цифровых объектов в настоящее время ведутся в 20 исследовательской комиссии МСЭ-Т. Как говорилось ранее, в декабре 2018 г. на процедуру согласия была представлена Рекомендация МСЭ-Т «Архитектура взаимодействия устройств интернета вещей на базе архитектуры цифровых объектов», а в 2019 г. планируется принять Рекомендацию МСЭ-Т «Структура решений по борьбе с контрафактными устройствами интернета вещей на базе архитектуры цифровых объектов». В этой связи совместное техническое решение, основанное на связке интернет-вещь–идентификатор DOA, может рассматриваться как эффективная технологическая цепочка. В модуль, который взаимодействует с сетевой инфраструктурой, может быть записан идентификатор DOA, в состав которого будут включены все уникальные параметры того или иного объекта (метаданные). Приложения такого решения могут быть самые разнообразные: ИКТ, фармацевтическая и автомобильная промышленности, авиастроение и т.д. В частности, они могут использоваться для борьбы с контрафактом.

2.3.1. Метод построения сетевой архитектуры цифровых объектов за счет введения промежуточного уровня взаимодействия

Взаимодействие элементов в рамках DOA предполагает коммуникации между распределенными LHR-серверами, расположенными в разных странах. Однако распределенность приводит к увеличению сетевой задержки, величина которой может оказаться неприемлемой для сервисов и приложений, требующих ультрамалых задержек на сетях связи [141].

Для минимизации сетевой задержки системы идентификации цифровых и физических объектов предлагается разбить систему резолюций, введя регистры промежуточного уровня между GHR и распределенными LHR. Эти промежуточные регистры можно также назвать средними регистрами (Middle Handle Register, MHR). Каждый MHR может быть привязан к определенному географическому региону на карте мира с учетом плотности и количества расположенных там устройств, а также плотности производителей (т.е. плотности LHR). LHR взаимодействует с ближайшим MHR вместо удаленного GHR, что уменьшает расстояние передачи данных по каналам связи и как следствие снижает сетевую задержку. Рисунок 2.5 иллюстрирует структуру системного уровня с новыми MHR. Выбор оптимального количества MHR и их географического распределения – это задача оптимизации, которая должна быть решена с точки зрения общей стоимости системы и параметров качества обслуживания сетей связи [62;63].

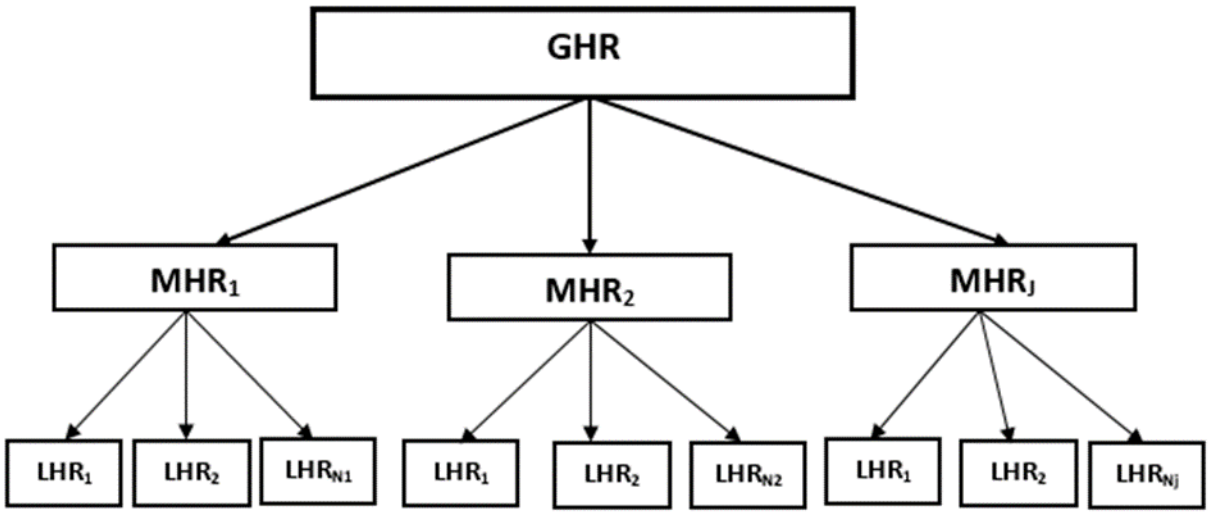


Рисунок 2.5 – Основные компоненты DOA с промежуточным уровнем взаимодействия

Как известно, глобальный регистр GHR располагается в Женеве (Швейцария), таким образом, серверы MHR могут располагаться по окружности круга радиуса r центром которого будет Женева (см. Рисунок 2.6). Стоит отметить, что r – это проектный параметр, т.е. он может быть получен путем решения задачи линейной оптимизации.

Рисунок 2.6 – Текущее местоположение GHR
и ожидаемые местоположения MHR

2.3.2. Математическая модель построения сетевой архитектуры цифровых объектов с промежуточным уровнем взаимодействия

Система имеет основной регистр (глобальный Регистр GHR), расположенный в городе Женева. Для разработки модели местоположение глобального регистра GHR обозначим G (l, h, φ, λ), где 1 и h – абсцисса и ордината местоположения GHR; φ, λ –широта и долгота GHR. GHR соединяется со всеми промежуточными (средними) регистрами MHR, развернутыми в системе. Множество MHR представим как M_j (l_j, h_j, φ_j, λ_j), j=1, 2, …, N, где l_j, h_j, φ_j и λ_j – соответственно абсцисса, ордината, широта и долгота j-го промежуточного регистра, а N – общее количество регистров MHR, развернутых в системе.

Каждый промежуточный регистр MHR соединяется с группой локальных регистров LHR и управляет ими. Регистры LHR, связанные с j-м MHR, образуют множество $L_i^j(l_i^j, h_i^j, \varphi_i^j, \lambda_i^j)$, i=1, 2, …, M_j, где l_i^j, h_i^j, φ_i^j и λ_i^j – соответственно абсцисса, ордината, широта и долгота i-го регистра LHR, связанного с j-м MHR, а

M_j – общее количество локальных регистров LHR, связанных с j-м регистром MHR, расположенным в месте, которое описывается координатами l_j, h_j, φ_j и λ_j.

Сетевая задержка L между двумя серверами прямо пропорциональна расстоянию D между передатчиком и приемником: $L \propto D$.

В предлагаемой системе сообщения передаются в основном между LHR и MHR. Таким образом, сетевая задержка для такой системы может быть рассчитана следующим образом:

$$L_i^j \propto D_i^j, \qquad (2.1)$$
$$D_i^j = \sqrt{\left(l_i^j - l_j\right)^2 + \left(h_i^j - h_j\right)^2},$$

где L_i^j – это сетевая задержка для данных, передаваемых между i-м LHR и j-м MHR;
D_i^j – расстояние между передатчиком i-го LHR и приемником j-го MHR.

Если отсутствует MHR, то регистры LHR взаимодействуют с GHR. Таким образом, сетевая задержка вычисляется между LHR и GHR, которые сохраняют свои местоположения в обеих системах (т.е. в системе с MHR и в традиционной системе без MHR). Сетевую задержку для системы без MHR можно рассчитать на основе следующего уравнения [120]:

$$L_i^{GHR} \propto D_i^{GHR}, \qquad (2.2)$$
$$D_i^{GHR} = \sqrt{\left(l_i^j - l\right)^2 + \left(h_i^j - h\right)^2},$$

где L_i^{GHR} представляет собой сетевую задержку данных, передаваемых между i-м регистром LHR и GHR;
D_i^{GHR} – это расстояние между i-м LHR и GHR.

Для сравнения между системой без MHR и новой структурой с MHR уравнения (2.2) и (2.1) представляются следующим образом:

$$\frac{L_i^{GHR}}{L_i^j} \propto \frac{D_i^{GHR}}{D_i^j}.$$

Тогда $\dfrac{L_i^{GHR}}{L_i^j} \propto \dfrac{\sqrt{\left(l_i^j - l\right)^2 + \left(h_i^j - h\right)^2}}{\sqrt{\left(l_i^j - l_j\right)^2 + \left(h_i^j - h_j\right)^2}}.$

Так как $D_i^j \square D_i^{GHR}$, то $L_i^{GHR} \square L_i^j$.

Следовательно, предлагаемая система резолюций обеспечивает более низкую сетевую задержку за счет уменьшения расстояния между серверами. Как говорилось ранее, это достигается путем развертывания промежуточного уровня регистров обработки (MHR) [53; 82].

Расстояния D_i^j и D_i^{GHR} также могут быть вычислены альтернативным способом, основанным на информации о широте φ и долготе λ местоположения регистра. Для того чтобы рассчитать кратчайшее расстояние между двумя точками в зависимости от их долготы и широты, будем использовать формулу гаверсинусов, которая применяется для оценки расстояния на поверхности сферы между двумя точками, у которых известны широта и долгота [111; 123]. Она позволяет вычислять расстояния с очень низкой погрешностью. Величина этой погрешности прямо пропорциональна расстоянию между точками, и для очень больших расстояний она не превышает 10–20 км.

$$A_i^j = \sin^2\left(\frac{\Delta\varphi_{j,i}^j}{2}\right) + \cos\varphi_j \cos\varphi_i^j \sin^2\left(\frac{\Delta\lambda_{j,i}^j}{2}\right);$$

где $\Delta\varphi_{j,i}{}^j$ – разница между долготой регистра LHR и соответствующего регистра MHR;
$\Delta\lambda_{j,i}{}^j$ – разница в широтах.

$$C_i^j = 2\arctan 2\left(\sqrt{A_i^j}, \sqrt{1-A_i^j}\right);$$

$$D_i^j = RC_i^j;$$

$$A_i^{GHR} = \sin^2\left(\frac{\Delta\varphi_{GHR,i}}{2}\right) + \cos\varphi_i \cos\varphi_{GHR} \sin^2\left(\frac{\Delta\lambda_{GHR,i}}{2}\right);$$

где $\Delta\varphi_{GHR,i}{}^{GHR}$ – разница между долготой регистра LHR и соответствующего регистра GHR;
$\Delta\lambda_{GHR,i}{}^{GHR}$ – разница в широтах.

$$C_i^{GHR} = 2\arctan 2\left(\sqrt{A_i^{GHR}}, \sqrt{1-A_i^{GHR}}\right);$$

$$D_i^{GHR} = RC_i^{GHR},$$

где R – радиус Земли.

2.4. Результаты экспериментов с моделью сетевой архитектуры цифровых объектов с промежуточным уровнем взаимодействия

Предлагаемый подход модифицированной системы регистров был протестирован на базе модельной сети для проверки производительности и уменьшения сетевой задержки по сравнению с существующей системой регистров. Для моделирования был использован программный пакет Matlab.

Предположим, что предлагаемая система MHR содержит $N = 10$ промежуточных регистров, которые расположены в разных странах и работают со всеми группами локальных регистров LHR по всему миру. Таблица 1 иллюстрирует специальные локации каждого промежуточного регистра с широтой φ_j и долготой λ_j. Кроме того, введем аппроксимированное расстояние D_j^{GHR} между каждым промежуточным регистром LHR и глобальным регистром GHR. Общее количество M_j локальных регистров LHR, соединенных с каждым промежуточным регистром MHR, представлены в Таблице 2.

Места размещения LHR, взаимодействующих с каждым MHR, приведены в Таблицах 2.2–2.13. Описания (спецификации) включают широту φ_j и долготу λ_j локации сервера LHR, аппроксимированное расстояние между каждым локальным регистром LHR и соответствующим промежуточным регистром MHR (D_i^j) и аппроксимированное расстояние D_j^{GHR} между каждым промежуточным регистром LHR и основным регистром GHR. Исходные параметры для моделирования приведены в Таблице 2.14.

Таблица 2.2 – Расположение промежуточных регистров MHR

j	*Страна*	*Город*	*Координата*		*Аппроксимированные расстояния* D_j^{GHR} *, км*
			Широта φj	Долгота λj	
1	*2*	*3*	*4*	*5*	*6*
1	Россия	Санкт-Петербург	59,9343°N	30,3351°E	2786,7
2	Египет	Каир	30,0444°N	31,2357°E	4070,0
3	Великобритания	Лондон	51,5074°N	0,1278°W	992,5
4	Испания	Мадрид	40,4168°N	3,7038°W	1384,1

Продолжение таблицы 2.2

1	*2*	*3*	*4*	*5*	*6*
5	США	Вашингтон	47,7511°N	120,7401°W	8365
6	Китай	Гуанчжоу	23,1291°N	113,2644° E	9388
7	Италия	Рим	41,9028°N	12,4964°E	887,2
8	Бразилия	Бразилия	14,2350°S	51,9253°W	8866
9	Канада	Кокран	51,2538°N	85,3232°W	6279
10	Австралия	Сидней	33,8688°S	151,2093°E	16764

Таблица 2.3 – Количество локальных регистров LHR, соединенных с каждым промежуточным регистром MHR

M_j	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}
Значение	10	8	6	4	8	10	5	4	5	4

Таблица 2.4 – Расположение M_1 локальных регистров LHR, соединенных с промежуточным регистром MHR_1

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^1, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^1	Долгота λ_i^1		
1	Россия	Москва	55,7558°N	37,6173°E	633	2418
2	Россия	Томск	56,5010°N	84,9925°E	3131	5280
3	Россия	Казань	55,8304°N	49,0661°E	1193	3128
4	Финляндия	Хельсинки	60,1699°N	24,9384°E	300,6	1980
5	Финляндия	Тампере	61,4978°N	23,7610°E	397,3	2042
6	Украина	Киев	50,4501°N	30,5234°E	1055	1854
7	Турция	Анкара	39,9334°N	32,8597°E	2231	2267
8	Россия	Волгоград	48,7080°N	44,5133°E	1544	2868
9	Иран	Тегеран	35,6892°N	51,3890°E	3091	3917
10	Россия	Сочи	43,6028°N	39,7342°E	1923	2641

Таблица 2.5 – Расположение M_2 локальных регистров LHR, соединенных с промежуточным регистром MHR_2

i	*Страна*	*Город*	*Координаты*		*Аппроксимированные расстояния* D_i^2, *км*	*Аппроксимированные расстояния* D_i^{GHR}, *км*
			Широта φ_i^2	Долгота λ_i^2		
1	Египет	Александрия	31,2001°N	29,9187°E	180	2635
2	Оман	Маскат	23,5859°N	58,4059°E	2783	5269
3	Южная Африка	Йоханнесбург	26,2041°S	28,0473°E	6264	8349
4	Саудовская Аравия	Эр-Рияд	24,7136°N	46,6753°E	1634	4312
5	ОАЭ	Дубай	25,2048°N	55,2708°E	2423	4913
6	Иордания	Амман	31,9454°N	35,9284°E	494,7	2994
7	Катар	Доха	25,2854°N	51,5310°E	2064	4626
8	Кувейт	Эль-Кувейт	29,3117°N	47,4818°E	1570	4030

Таблица 2.6 – Расположение M_3 локальных регистров LHR, соединенных с промежуточным регистром MHR_3

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^3, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^3	Долгота λ_i^3		
1	Великобритания	Ливерпуль	53,4084°N	2,9916°W	286,9	1033
2	Нидерланды	Амстердам	52,3680°N	4,9036°E	357,9	691,2
3	Норвегия	Осло	59,9139°N	10,7522°E	1154	1554
4	Ирландия	Дублин	53,3498°N	6,2603°W	463,3	1191
5	Швеция	Стокгольм	59,3293°N	18,0686°E	1433	1660
6	Франция	Париж	48,8566°N	2,3522°E	343,6	409,8

Таблица 2.7 – Расположение M_4 локальных регистров LHR, соединенных с промежуточным регистром MHR_4

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^4, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^4	Долгота λ_i^4		
1	Испания	Севилья	37,3891°N	5,9845° W	390,2	1014
2	Португалия	Лиссабон	38,7223°N	9,1393°W	502,4	1501
3	Марокко	Касабланка	33,5731°N	7,5898°W	835,3	1824
4	Алжир	Алжир	36,7538°N	3,0588°E	714,8	1082

Таблица 2.8 – Расположение M_5 локальных регистров LHR, соединенных с промежуточным регистром MHR_5

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^5, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^5	Долгота λ_i^5		
1	США	Нью-Йорк	40,7128°N	74,0060°W	3745	6216
2	США	Канзас-Сити	39,0119°N	98,4842° W	2034	7942
3	Мексика	Мехико	23,6345°N	102,5528°W	3127	9458
4	США	Чикаго	41,8781°N	87,6298°W	2670	7048
5	США	Лас-Вегас	36,1699°N	115,1398°W	1368	9139
6	США	Сан-Франциско	37,7749°N	122,4194°W	1118	9362
7	США	Майами	25,7617°N	80,1918°W	4289	7705
8	США	Новый Орлеан	29,9511°N	90,0715° W	3276	8097

Таблица 2.9 – Расположение M_6 локальных регистров LHR, соединенных с промежуточным регистром MHR_6

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^6, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^6	Долгота λ_i^6		
1	Китай	Пекин	39,9042°N	116,4074°E	1889	8205
2	Китай	Шанхай	31,2304°N	121,4737° E	1212	9234
3	Южная Корея	Сеул	37,5665°N	126,9780°E	2071	8991
4	Япония	Токио	35,6895°N	139,6917°E	2902	9792
5	Китай	Гонконг	22,3964°N	114,1095° E	118,9	9512
6	Малайзия	Куала-Лумпур	3,1390°N	101,6869°E	2548	10180
7	Бруней	Бандар-Сери-Бегаван	4,5353°N	114,7277° E	2074	11050
8	Тайвань	Гаосюн	22,6273°N	120,3014°E	723	9904
9	Индия	Мумбаи	19,0760°N	72,8777°E	4201	6714
10	Индия	Нью-Дели	28,6139°N	77,2090°E	3645	6352

Таблица 2.10 – Расположение M_7 локальных регистров LHR, соединенных с промежуточным регистром MHR_7

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^7, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^7	Долгота λ_i^7		
1	Тунис	Тунис	36,8065°N	10,1815°E	600,6	1097
2	Италия	Неаполь	40,8518°N	14,2681°E	188,4	884,2
3	Греция	Афины	37,9838°N	23,7275°E	1051	1709
4	Турция	Стамбул	41,0082°N	28,9784°E	1375	1919
5	Венгрия	Будапешт	47,4979°N	19,0402°E	808,4	990,1

Таблица 2.11 – Расположение M_8 локальных регистров LHR, соединенных с промежуточным регистром MHR_8

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^8, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^8	Долгота λ_i^8		
1	Аргентина	Росарио	32,9442°S	60,6505°W	2260	11060
2	Чили	Сантьяго	33,4489°S	70,6693°W	2853	11720
3	Бразилия	Рио-де-Жанейро	22,9068°S	43,1729°W	1334	9147
4	Боливия	Ла-Пас	16,4897°S	68,1193°W	1754	10170

Таблица 2.12 – Расположение M_9 локальных регистров LHR, соединенных с промежуточным регистром MHR_9

i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^9, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^9	Долгота λ_i^9		
1	Канада	Оттава	45,4215°N	75,6972°W	961,3	6041
2	Канада	Йеллоухед Каунти	53,9333°N	116,5765°W	2114	7639
3	Канада	Калгари	51,0486°N	114,0708°W	1992	7781
4	Канада	Торонто	43,6532°N	79,3832°W	955,2	6394
5	Канада	Террас	53,7267°N	127,6476°W	2836	8076

Таблица 2.13 – Расположение M_{10} локальных регистров LHR, соединенных с промежуточным регистром MHR_{10}

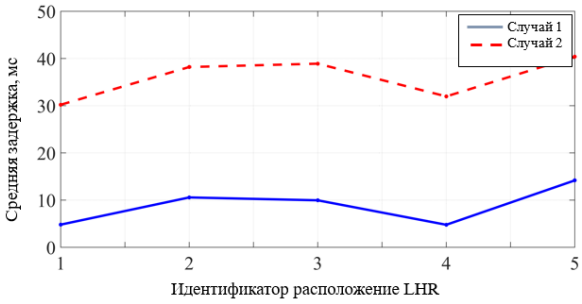
i	*Страна*	*Город*	*Координата*		*Аппроксимированное расстояние* D_i^{10}, *км*	*Аппроксимированное расстояние* D_i^{GHR}, *км*
			Широта φ_i^{10}	Долгота λ_i^{10}		
1	Австралия	Перт	31,9505°S	115,8605°E	3291	13950
2	Австралия	Мельбурн	37,8136°S	144,9631°E	713,4	16530
3	Австралия	Дарвин	12,4634°S	130,8456°E	3149	13650
4	Новая Зеландия	Веллингтон	41,2865°S	174,7762°E	2226	18950

Таблица 2.14 – Исходные параметры для моделирования

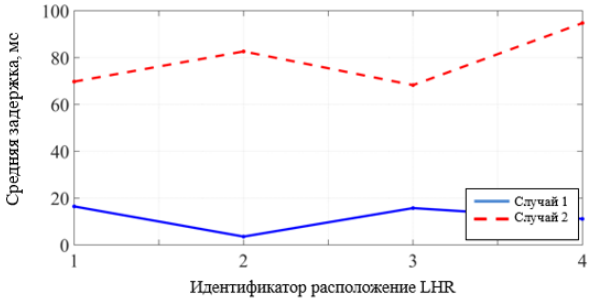
Параметр	*Приложение*	*Значение*
Скорость распространения	ν	200 м/мкс
Приблизительное местоположение GHR	φ_{GHR}, λ_{GHR}	46,2044°N; 6,1432°E
N		10

2.5 Анализ результатов математического моделирования

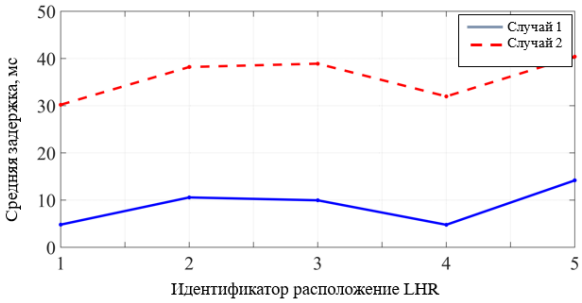
Чтобы проиллюстрировать уменьшение сетевой задержки в модифицированной архитектуре системы регистров по сравнению с существующей системой регистров, рассмотрим результаты моделирования. На Рисунке 2.7 сравниваются сетевые задержки в модифицированной и традиционной архитектурах: между каждым локальным регистром LHR и соответствующим ему промежуточным регистром MHR (Случай 1, сплошная синяя кривая) и между каждым локальным регистром LHR и глобальным регистром GHR в отсутствие MHR (Случай 2, пунктирная красная кривая). На Рисунке 2.7, а показаны задержки для всех десяти LHR, относящихся к промежуточному регистру MHR1 ($M_1 = 10$); Рисунок 2.7, б описывает восемь LHR, расположенных на территории регистра MHR_2 ($M_2 = 8$); и т.д. вплоть до Рисунка 2.7, к, где изображены задержки для четырех LHR промежуточного регистра MHR_{10} ($M_{10} = 4$).



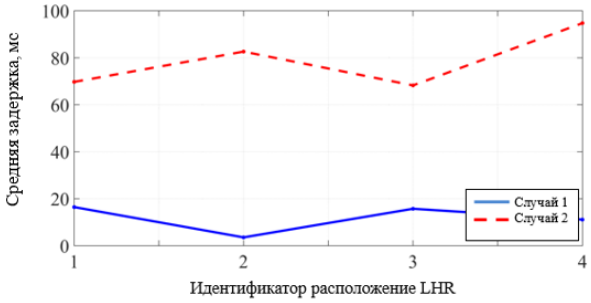
а) MHR_1, M_1=10



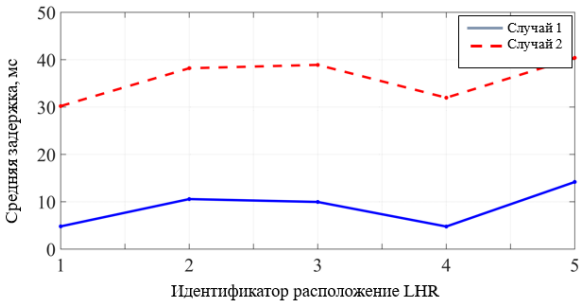
б) MHR_2, M_2=8



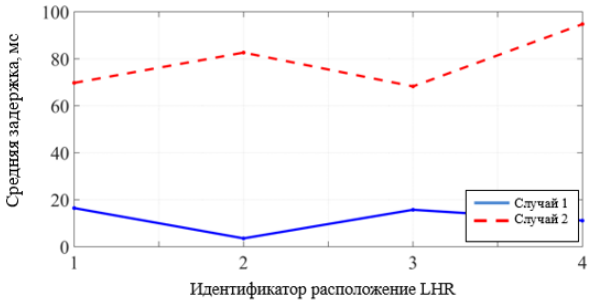
в) MHR_3, M_3=6



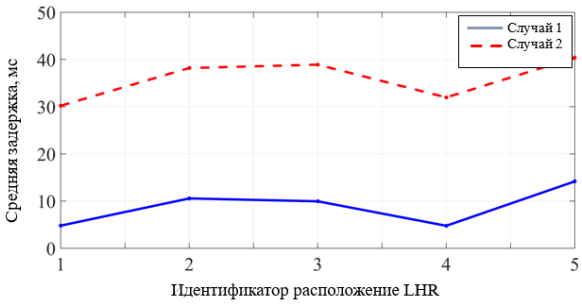
г) MHR_4, M_4=4



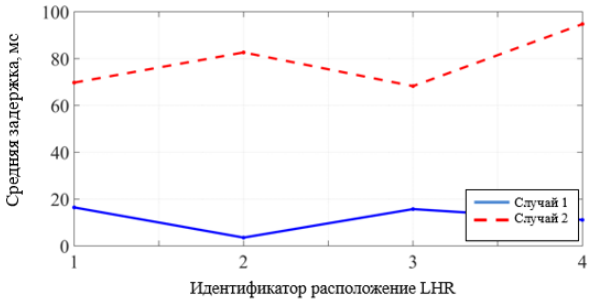
д) MHR_5, M_5=8



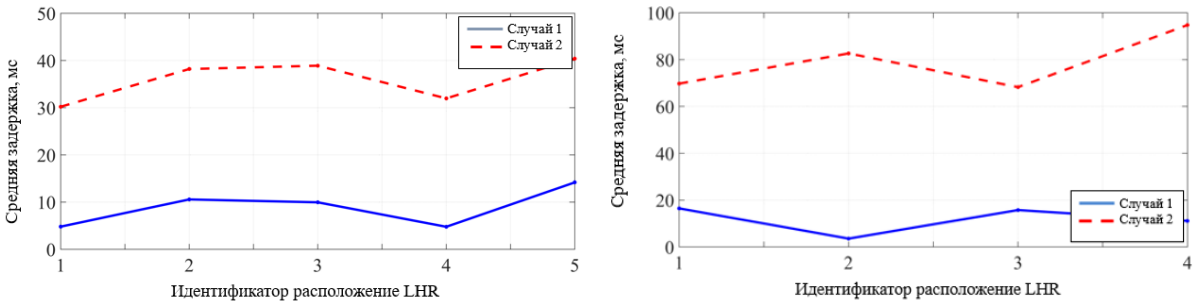
е) MHR_6, M_6=10



ж) MHR_7, M_7=5



з) MHR_8, M_8=4



и) MHR_9, M_9=5 *к*) MHR_{10}, M_{10}=4

Рисунок 2.7 – Средняя сетевая задержка для модифицированной (Случай 1) и традиционной (Случай 2) архитектуры системы резолюций Handle System для всех MHR

Как видно на графиках (см. Рисунок 2.7, *а*–*к*), средняя сетевая задержка модифицированной системы регистров (Случай 1) меньше, чем средняя сетевая задержка существующей системы регистров без MHR (Случай 2). Причем уменьшение задержки достигается для всех рассматриваемых LHR, которые распределяются случайным образом по всему миру

Таблица 2.15 – Процент уменьшения задержки при использовании MHR

	Уменьшение сетевой задержки LHR_i, %										*Среднее значение уменьшения сетевой задержки t, %*
MHR_j	LHR_1	LHR_2	LHR_3	LHR_4	LHR_5	LHR_6	LHR_7	LHR_8	LHR_9	LHR_{10}	
MHR_1	73,82	40,70	61,86	84,82	80,55	43,09	1,59	46,16	26,19	27,19	48,60
MHR_2	93,17	47,18	24,97	62,11	50,68	83,48	55,38	61,04	–	–	59,75
MHR_3	27,24	48,23	25,74	61,11	13,67	16,15	–	–	–	–	32,02
MHR_4	61,52	66,53	54,21	33,94	–	–	–	–	–	–	54,05
MHR_5	39,75	74,39	66,94	62,12	85,03	88,06	44,33	59,54	–	–	67,54
MHR_6	76,98	86,87	76,96	70,36	98,75	74,97	81,23	92,70	37,43	42,62	73,89
MHR_7	45,25	78,69	38,50	28,35	1,29	–	–	–	–	–	38,42
MHR_8	79,57	75,66	85,57	82,75	–	–	–	–	–	–	80,89
MHR_9	84,09	72,33	74,40	85,06	64,88	–	–	–	–	–	76,15
MHR_{10}	76,41	95,68	76,93	88,25	–	–	–	–	–	–	84,32
Уменьшение средней сетевой задержки всех MHR											61,56%

В Таблице 2.15 представлены данные о процентном уменьшении сетевой задержки каждого LHR при использовании модифицированной системы регистров

по сравнению с существующей системой регистров, а также средние значения уменьшения задержки для каждой группы LHR, связанных с определенным MHR. Средняя сетевая задержка всех LHR, используемых в предлагаемой системе с промежуточными MHR, на 61,56 % меньше, чем в существующей системе LHR. Таким образом, модифицированная система регистров может уменьшить сетевую задержку до 60% по сравнению с существующей системой регистров без MHR.

Выводы по главе 2

1. Проведен анализ построения сетевой архитектуры цифровых объектов. Рассмотрены основные компоненты архитектуры DOA и принципы их взаимодействия.

2. Проанализированы служебные протоколы DOA и особенности их функционирования. Показаны отличия текущих версий протоколов IRP и DOIP как по структуре, так и по функциональному назначению.

3. Предложена модель системы идентификации на базе архитектуры цифровых объектов, отличающаяся от известных тем, что для обеспечения приемлемого качества обслуживания в общей архитектуре сетей связи общего пользования, существующей сегодня, была разработана новая архитектура взаимодействия путем введения регистра промежуточного уровня (Middle Handle Register – MHR) между глобальным регистром (Global Handle Register – GHR) и локальным регистром (Local Handle Register – LHR).

4. На базе предложенной математической модели был проведен численный анализ, который показал, что предлагаемая система обеспечивает более высокую производительность с точки зрения сетевой задержки, ввиду уменьшения расстояния между серверами LHR, что достигается путем развертывания регистров промежуточного уровня обработки (MHR).

Результаты моделирования показали, что введение промежуточного уровня регистров MHR позволит снизить задержку на 60% по сравнению с существующей архитектурой.

Глава 3. МОДЕЛЬ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ АРХИТЕКТУРЫ ЦИФРОВЫХ ОБЪЕКТОВ

3.1. Имитационное моделирование как научный подход к исследованиям концепции Интернета вещей

Неоднородность возможных сценариев, возникающая в результате массового развертывания огромного количества датчиков и устройств, требует использовать сложные методы моделирования и симуляции. Фактически, моделирование ИВ представляет несколько вопросов как с количественной, так и с качественной стороны. В настоящее время существуют новые методы моделирования для повышения масштабируемости и обеспечения возможности выполнения в реальном времени массово заполненных сред ИВ (например, больших интеллектуальных городов).

Моделирование общей среды ИВ для создания эффективных и интеллектуальных услуг может быть довольно сложным из-за разнородности возможных сценариев. Таким образом, моделирование в рамках концепции Интернета вещей необходимо как с количественной, так и с качественной стороны, чтобы дать ответы на несколько вопросов: планирование емкости, моделирование и анализ «что если», упреждающее управление и поддержка многих конкретных оценок, связанных с безопасностью.

Масштаб ИВ является основной проблемой при использовании существующих инструментов моделирования. Традиционные подходы (основанные на одном процессоре) часто не могут масштабироваться до количества узлов (и уровня детализации), требуемого Интернету вещей.

В этой связи, подходы, основанные на имитации процессов, которые порой невозможно воссоздать в реальном мире позволяют оценить те или иные параметры Интернета вещей. Несомненно, детализация имитационной модели и результатов экспериментов с ней будут зависеть от уровня абстракции, заложенном на этапе создания модели.

3.2. Определение состава факторов, влияющих на идентификацию интернета вещей

Как уже отмечалось, в современном обществе значительную часть рынка технических систем занимает Интернет вещей. Данные устройства находят место во многих областях, начиная от простого бытового использования, медицины и заканчивая применением в военных целях. По приблизительным оценкам, число устройств ИВ (от англ. Internet of Things) составляет порядка 28 миллиардов и это цифра с каждым годом растет. Гигантское множество устройств интернета вещей взаимодействуют друг с другом ежедневно, что открывает широчайшие возможности по созданию приложений различного класса на базе данных «умных» систем.

Очевидно, что для обеспечения корректной и быстрой работы с огромным потоком информации от 28 миллиардов устройств требуется наличие надежной системы адресации и идентификации, в связи с чем выделяется отдельная область задач по идентификации – идентификация интернета вещей. Основной проблематикой данной области является присвоение уникальных идентификаторов и связанных с ними метаданных устройствам интернета вещей, позволяющая им обмениваться информацией с различными сущностями в сети Интернет [83–85].

Далее обобщены основные особенности идентификации для интернета вещей, а именно:

– различный жизненный цикл устройств (одни объекты ИВ могут существовать довольно длительное время, другие же – наоборот);

– взаимоотношение объектов интернета вещей с другими сущностями, не входящими в данную систему (у устройств интернета вещей в течении жизненного цикла могут изменяться владельцы и администраторы, что влияет на процессы идентификации, аутентификации и авторизации);

– особые требования к контексту, в котором работают устройства (в определенных случаях доступ объектов к одним и тем же данным может быть разрешен или ограничен в зависимости от ситуации);

– требования к обеспечению механизмов защиты (при проектировании данных механизмов стоит учитывать ограниченность устройств интернета вещей по ресурсам и производительности);

– возможность расширения системы идентификации до огромного количества устройств (свыше миллиарда);

– возможность эффективно работать для самых различных устройств (устройства в сети ИВ могут быть крайне разнородны по своим ресурсам и производительности;

– прозрачность системы адресации и независимость от сети (в отличии от классических систем адресации, применяемых, например, в сети Интернет, идентификация устройств интернета вещей должна быть независима от того, в какой сети они находятся или какому пользователю принадлежат; кроме того, следует учитывать, что устройства интернета вещей могут менять свое местоположение, но при этом быть однозначно идентифицированы в сети);

– гибкий и эффективный механизм резолюции идентификаторов (устройства интернета вещей должны быть точно определены в независимости от их местоположения; кроме того, должна присутствовать простота в подключении и настройке нового объекта IoT к существующей сети);

– безопасность и сохранность пользовательских данных (не стоит забывать, что устройства ИВ работают зачастую с огромным количеством персональных данных, что требует дополнительных мер защиты) [139].

Одним из возможных решений по проверки реализации перечисленных выше факторов при идентификации ИВ на базе архитектуры цифровых объектов является разработка имитационных моделей в пакете Anylogic. Данный пакет активно используется на кафедре Сетей связи и передачи данных в Лаборатории «Моделирования и оптимизации сетей связи им. Г.Г. Яновского» при проведении практических и лабораторных занятий, а также в научных исследованиях ввиду своей универсальности.

3.3. Описание структуры имитационной модели DOA в пакете AnyLogic

Как отмечено во второй главе диссертационной работы что, серверная часть включает в себя функционал по разрешению идентификаторов, что является обязательным условием верификации идентификатора в архитектуре цифровых объектов. При подготовке к созданию имитационной модели был введен ряд ограничений, с учетом допустимого уровня абстракции, что позволило представить серверную часть в виде двух серверов: GHR и LHS, каждый из которых имеет доступ к своей базе данных. Общая схема структуры приложения представлена на рисунке ниже [52]. На GHR хранится информация о правообладателях контента. На LHS хранится информация о продукции. Общая схема структуры приложения представлена на Рисунке 3.1.

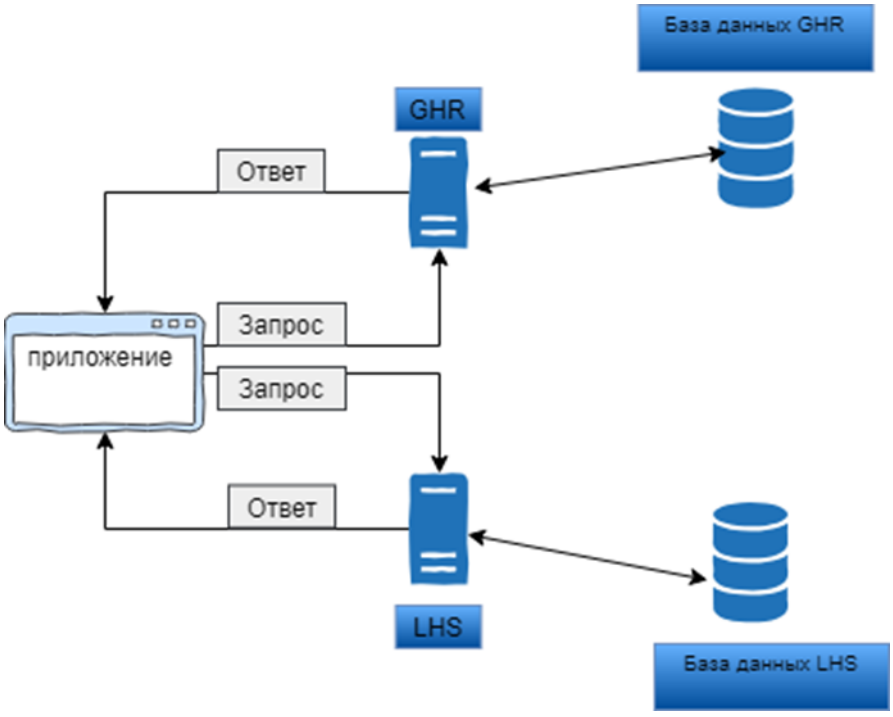


Рисунок 3.1 – Общая схема структуры идентификации

Сервер идентификации является ключевым элементом при симуляции происходящих процессов. Общий алгоритм работы сервера представлен на Рисунке 3.2.

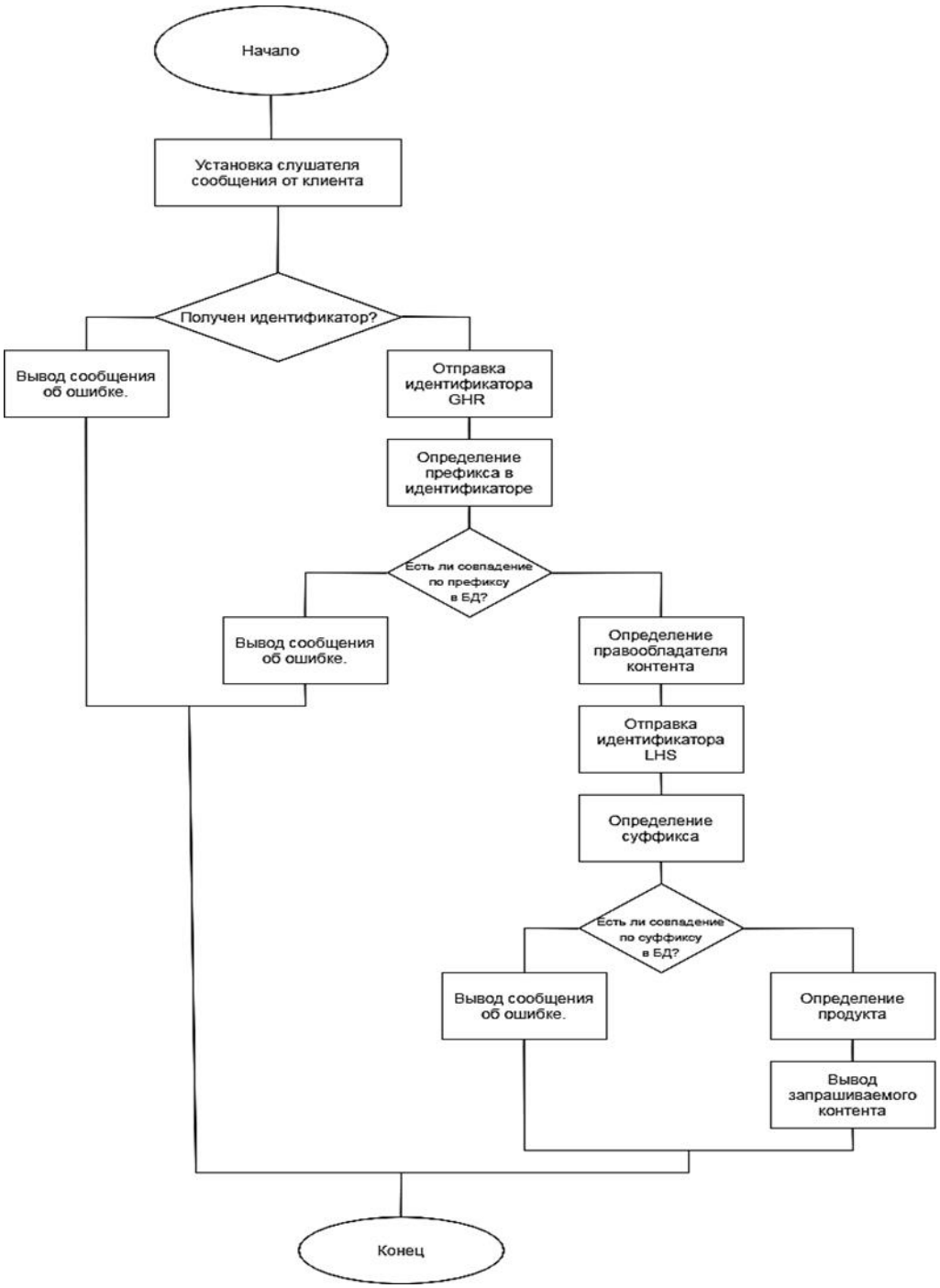


Рисунок .32 – Блок-схема алгоритма работы сервера идентификации

Для того, чтобы охарактеризовать эффективность системы резолюции идентификаторов в архитектуре DOA при применении к задачам идентификации интернета вещей, рассмотрим систему резолюции как СМО (систему массового обслуживания).

В качестве системы СМО рассматривалась модель M/M/n/m. Данная модель характеризует систему с экспоненциальным распределением времени

обслуживания заявок и экспоненциальным распределением времени между поступления заявок [87]. Кроме того, модель удовлетворяет следующим условиям:

– наличие нескольких каналов обработки. В данной модели будут рассматриваться серверы GHR как самостоятельная сущность, только лишь обрабатывающая приходящие в нее запросы [55];

– ограничение на длину буфера GHR отсутствует. Каждый запрос, поступивший в систему будет обслужен;

– нет приоритетности у поступающих запросов, каждый запрос обрабатывается в той последовательности, в которой поступил в систему.

Стоит заметить, что при анализе обработки запросов на протяжении длительного периода времени (например, суток), выбранная модель уже не будет валидной [86]. Однако на коротких промежутках времени эту модель можно использовать. В качестве времени работы системы был выбран промежуток в 200 с.

3.4. Эксперименты с имитационной моделью

Модель системы резолюции в виде СМО была реализована путем анализа существующей реализации системы резолюции [48; 86; 128]. В существующей архитектуре используется не один GHR-сервер, а несколько серверов, принадлежащих MPA (от англ. Multi-Primary Administrators), контролируемым DONA Foundation [1; 2; 8]. Каждый MPA-сервер представляет собой GHR, способный разрешать поступающие на него запросы. Путем анализа работы программного обеспечения, предоставляемого Handling.net, была установлена инфраструктура серверов глобальных регистров верхнего уровня и определена средняя задержка на разрешение запроса этими серверами. В данном программном обеспечении все MPA сервера эквивалентны между собой и запрос на разрешение поступает последовательно на все сервера и анализируется ответ, который пришел первым. При этом отсутствует учет и анализ времени задержки до сервера. По сути системы резолюции гарантирует, что если запрос на разрешение поступил в систему, то он обязательно будет обслужен, однако время, которое может потребоваться на это четко не регламентировано [78]. В Таблице 3.1 представлены

характеристики серверов MPA, используемых в качестве GHR в действующей архитектуре системы резолюции [43;100].

Таблица 3.1 – Характеристики серверов MPA

MPA	***IP адрес***	***Средняя задержка на разрешение, мс***
CNRI (Америка)	132.151.20.9; 38.100.138.153; 38.100.138.131; 132.151.20.9; 2001:550:100:6::138:153; 2001:550:100:6::4; 132.151.1.179	243.548
ITU (Швейцария)	156.106.193.160	71.33
Beijing Flash Newsletter Cas Telecommunication (Китай)	119.90.34.34	473.583
Alicloud (Китай)	47.90.103.77	410.693
ATI – Agence Tunisienne Internet (Тунис)	41.231.118.2	82.510
Gesellschaft Fuer Wissenschaftliche Datenverarbeitung Mbh Goettingen (Германия)	134.76.30.197	44.356
Communications And Information Technology Commission (Саудовская Аравия)	86.111.195.107	318.450
Liquid Telecommunications Operations Limited (Кения)	196.12.152.22	258.450

В разработанной модели рассматривался процесс разрешения идентификатора. На Рисунке 3.3 изображена основная диаграмма процесса обработки заброса системой массового обслуживания, в которой рассматривался процесс разрешения идентификатора. Имитационное моделирование системы массового обслуживания было проведено в пакете Anylogic с использованием дискретно-событийного подхода.

Элемент clients соответствует источнику заявок на разрешение идентификаторов, поступающих от устройств. Далее идет разветвление на 8 каналов, каждый из которых соответствует инфраструктуре определенного MPA. Вероятность выбора каждого из каналов в существующей системе одинакова. Каждый сервер MPA представляет собой набор из буфера заявок и сервера обработки идентификатора. При этом количество каналов в сервере обработки

соответствует количеству серверов каждого конкретного MPA, приведенных в Таблице 3.1.

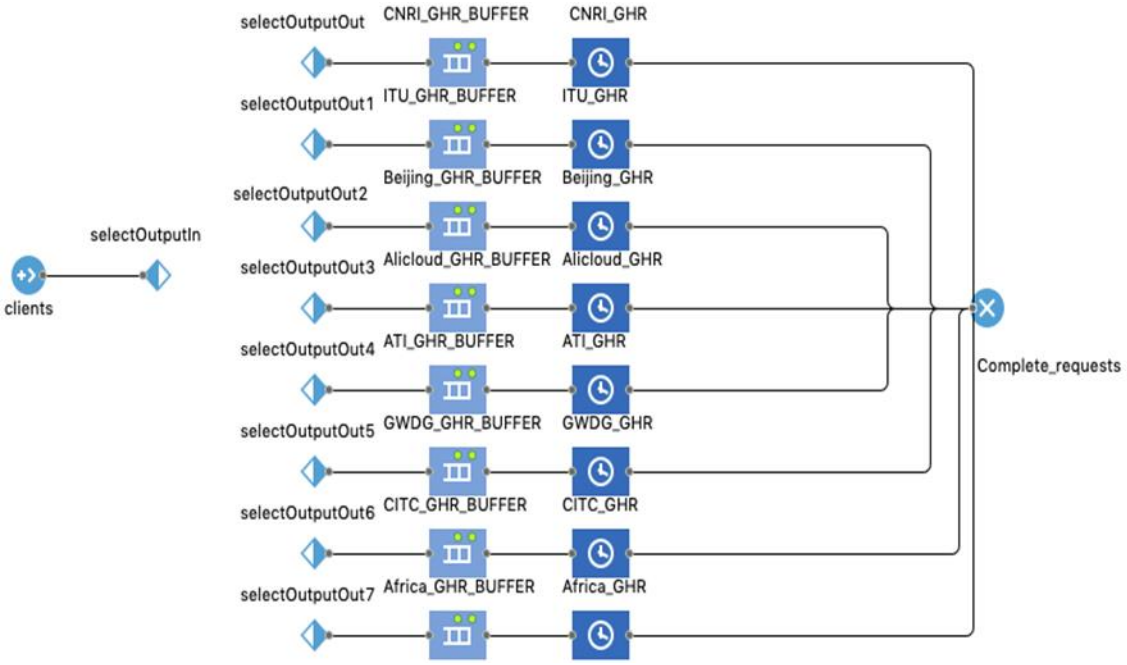


Рисунок 3.3 – Структура блоков имитационной модели разрешения заявок идентификатора в AnyLogic

Следует отметить что, в рамках исследования исследовалась статистика только на уровне GHR и не анализировался следующий уровень работы системы т.е. с LHS. Взаимодействие с локальными серверами и анализ их конфигурации должен рассматриваться отдельно в рамках конкретной решаемой задачи.

3.5. Анализ результатов имитационного моделирования

Так как DOA строится на базе сетевой архитектуры, которая уже существует на данный момент для глобальной сети Интернет, то основными параметрами, влияющими на работу, будут величина сетевой задержки для поступающего запроса, скорость обработки запроса сервером, ответственным за резолюцию, а также количество каналов обработки у каждого MPA [140].

Характеристикой системы резолюции, критичной для идентификации Интернета вещей, является среднее время обслуживания одного запроса. Данное время будет зависеть как от конфигурации системы, так и от интенсивности нагрузки.

На Рисунке 3.4 показана зависимость среднего времени разрешения идентификатора от интенсивности поступающих запросов при текущей конфигурации системы. Как видно из графика, с ростом интенсивности нагрузки увеличивается и среднее время разрешения одного идентификатора, причем при больших нагрузках это время доходит до 30 секунд, что достаточно много для реальных приложений, особенно если сравнивать с показателями системы DNS.

Рисунок 3.4 – Зависимость времени разрешения от интенсивности запросов

Используя возможности среды Anylogic проведем оптимизационный эксперимент, направленный на установление наиболее подходящей инфраструктуры GHR-серверов при текущей конфигурации временных задержек с целью снизить среднее время разрешения идентификатора. Основным параметром для оптимизации будет количество GHR-серверов, используемым каждым из MPA. В качестве целевой функции будем стремится минимизировать время разрешения запроса. Установим время разрешения не более 1 секунды. Зададимся значением интенсивности в 50 Эрл.

Результаты оптимизационного эксперимента представлены в Таблице 3.2, где: alfa – параметр интенсивности нагрузки; d1...d8 – количество серверов каждого из MPA.

Процесс оптимизации заключается в последовательном запуске модели с варьированием параметров оптимизации (количества серверов GHR) для

достижения установленной цели (времени разрешения идентификатора менее 1 сек.). В столбце «Текущее» представлены параметры оптимизируемой модели на текущем шаге итерации. Строка «Функционал» показывает значение оптимизационной функции на текущем шаге. В конце оптимизационного процесса мы получаем набор параметров (количество серверов GHR), наиболее близко обеспечивающих результат времени разрешения идентификатора не более 1 сек. Для текущей конфигурации модели количество серверов составляет 7, 10, 1, 10, 10, 10, 10, 10 для каждого MPA из Таблицы 3.2 соответственно, как показано в столбце «Лучшее».

Таблица 3.2 – Результаты оптимизационного эксперимента

Текущее		*Лучшее*
Итерация	500	60
Функционал	3,947	0,878
Параметры		
alfa	50	50
d1	7	7
d2	9	10
d3	4	1
d4	9	10
d5	8	10
d6	8	10
d7	10	10
d8	8	10

Как видно из строки «Функционал» для лучшей итерации значение времени разрешения идентификатора составило 0,878 сек. По графику на Рисунке 3.5 видно, что при конфигурации GHR-серверов, взятых по результатам оптимизационного эксперимента, разрешение идентификатора в системе происходит гораздо быстрее; прирост скорости в 15 раз достигается на максимальной интенсивности нагрузки.

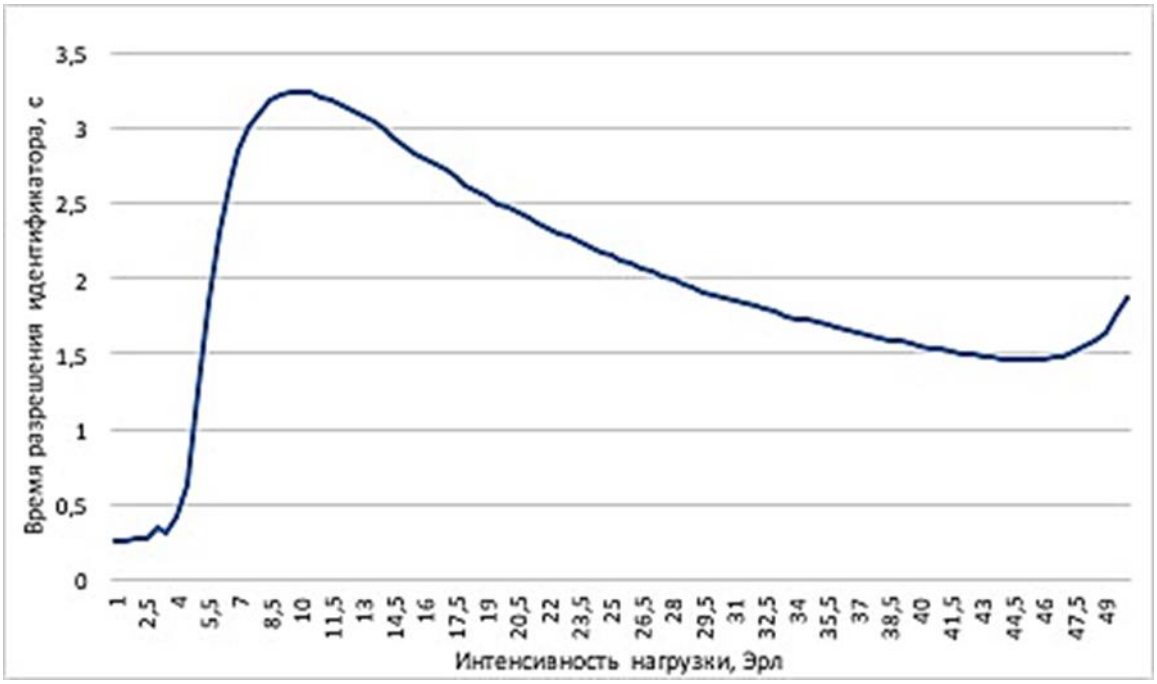


Рисунок 3.5 – Время разрешения запроса от интенсивности нагрузки при оптимальной конфигурации

Основываясь на результатах моделирования системы можно сделать выводы о том, что текущая инфраструктура системы резолюции требует дальнейшего масштабирования и распределения для того, чтобы быть способной выдерживать большие нагрузки и минимизировать время разрешения поступающих запросов. Особенно актуально это при использовании архитектуры DOA и системы резолюции в задачах, связанных с идентификацией устройств интернета вещей, количество которых исчисляется миллиардами; интенсивность запросов в системе резолюции при этом может быть сверхвысокой.

Помимо инфраструктурного расширения существующей системы доработки нужно вести и в программной части системы резолюции. Как уже было упомянуто ранее, в результате анализа открытого исходного кода библиотеки, предоставляемой Handling.net [15; 137; 135] для построения собственных клиентских решений для взаимодействия с системой резолюции, было установлено, что при отправке запроса на разрешение идентификатора к серверам GHR не производится предварительного анализа времени сетевой задержки до каждого из серверов. Каждый сервер из списка, приведенного в Таблице 1, опрашивается в случайной последовательности и анализируется первый полученный ответ. Такая реализация несомненно сказывается на общем времени

разрешения идентификатора . Поэтому требуется дальнейшая модификация исходного с целью создания функционала сортировки и приоритизации серверов GHR в зависимости от сетевой задержки от клиентского устройства.

3.6. Математическая модель системы резолюции

Как было описано выше система резолюций состоит из двух типов реестров – GHR и LHR. Пусть группа реестров GHR определяется символом *Gj*, где *j* = 1, 2, 3…*N*, где *N* – общее число реестров GHR в системе. Каждый реестр GHR объединяет и контролирует определенный набор локальных реестров. Набор локальных реестров, подсоединенных к *j*-му GHR, обозначается символом *Lji*, где *i* = 1, 2, 3...*Mj*, где *Mj* – общее количество LHR, подсоединенных к *j*-му GHR. Переданные пакеты прибывают на сервер с определенной частотой, соответствующей Пуассоновскому процессу, формируя одиночную очередь на контроллере. Такая система может быть смоделирована на основе многоканальной модели массового обслуживания (M/M/s).

Тогда среднее время ответа *Tj* реестра GHR *Gj* равно сумме времени в очереди и времени обработки, и может быть вычислено при помощи формулы Эрланга, как функция частоты поступления λ*i* запросов и частоты обслуживания μ:

$$T_j(\lambda) = \frac{f\left(s, \frac{\lambda_j}{\mu}\right)}{s\mu_j - \lambda_j} + \frac{1}{\mu}. \tag{3.1}$$

Функция $f\ (s,\ \lambda/\mu)$ определяет вероятность того, что все серверы в системе используются, и любая из поступивших заявок попадет в очередь:

$$f\left(s, \frac{\lambda}{\mu}\right) = \frac{1}{1 + \left(\frac{1}{1-\gamma}\right)\left(\frac{s!}{(s\gamma)^s}\right)\sum_{k=0}^{s-1}\frac{(s\gamma)^k}{k!}}, \tag{3.2}$$

$$\gamma = \frac{\lambda_j}{s.\mu}. \tag{3.3}$$

Функция γ показывает использование системы, что отражает также ее стабильность. Система стабильно распределена только если показатель использования системы γ меньше единицы. Данная информация может быть корректно интерпретирована при помощи диаграммы состояний многоканальной

модели M/M/s. В случае, когда число заявок в очереди больше, чем на сервере контроллера, обработка будет происходить с той же частотой μ, при этом контроллер будет предельно заполнен (Рисунок 3.6).

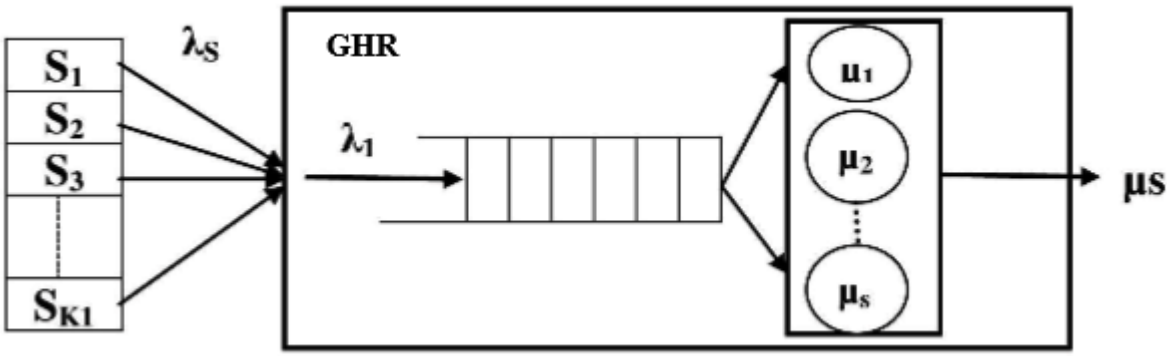


Рисунок 3.6 – Представление процесса разрешения идентификаторов на сервере GHR в виде объектов СМО

Частота поступления заявок λ_j реестра GHR G_j рассчитывается как сумма средних частота поступления заявок на локальных реестрах (L_i^j), подсоединенных к реестру G_j:

$$\lambda_j = \sum_{L_i} \lambda_l. \tag{3.4}$$

Средняя нагрузка на сервер-посредник G_j рассчитывается как среднее число поступивших и обработанных запросов. При помощи формулы Эрланга рассчитывается средняя нагрузка L_j на реестрах GHR:

$$L_j(\lambda) = s\gamma + \frac{\gamma}{1-\gamma} f\left(s, \frac{\lambda_j}{\mu}\right). \tag{3.5}$$

Таким образом, на базе полученной формулы Эрланга можно произвести численный расчет средней нагрузки *Lj* на реестрах GHR.

3.7. Апробация методов идентификации устройств интернета вещей на базе архитектуры цифровых объектов

Для апробации методов идентификации устройств интернета вещей на базе архитектуры цифровых объектов был разработан лабораторный стенд, который был основана на непосредственном взаимодействии идентифицируемого устройства с Handle-сервером через Интернет (Рисунок 3.7). В ходе эксперимента

рассматривался сценарий идентификации устройства с применением промежуточного устройства верификации.

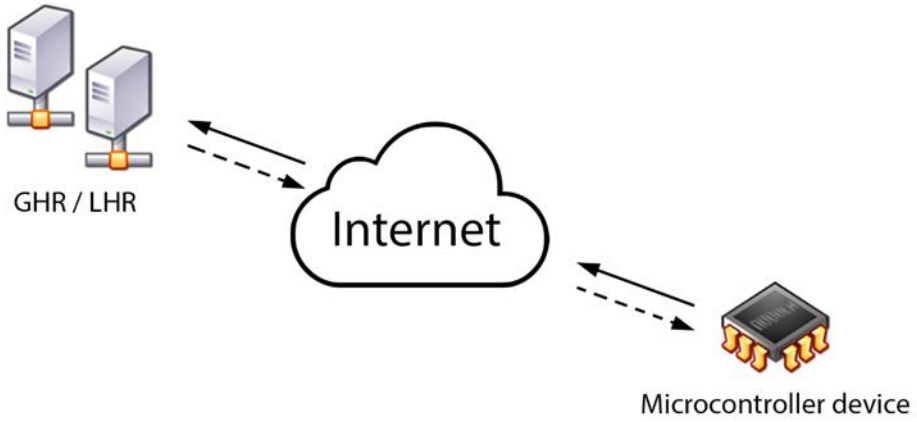


Рисунок 3.7 – Схематичное изображение взаимодействия элементов при идентификации устройств ИВ на базе DOA (традиционный подход)

Лабораторный стенд для идентификации устройств ИВ был разработан с введением нового компонента (в отличии от традиционного подхода) – уровня верификации объектов в системе DOA. Стенд состоят из следующих компонентов (Рисунок 3.8):

1) Handle-сервер, содержащий информацию об идентифицируемом устройстве;

2) сеть интернет в качестве сетевой инфраструктуры;

3) конечное устройство (устройство IoT или любой другой идентифицируемый объект);

4) дополнительный уровень верификации объектов в системе Digital Object Architecture.

Рассматривая отличия основных компонентов системы, стоит отметить объединение Global Handle Register и Local Handle Register в один объект для осуществления испытаний на лабораторном стенде, участникам исследования был предоставлен доступ к тестовой зоне DOA с префиксом “11.test”, позволяющим разместить собственные идентификаторы в существующей системе Digital Object Architecture. В перспективе, это даёт возможность оценить множество характеристик разрабатываемой системы на прикладном уровне.

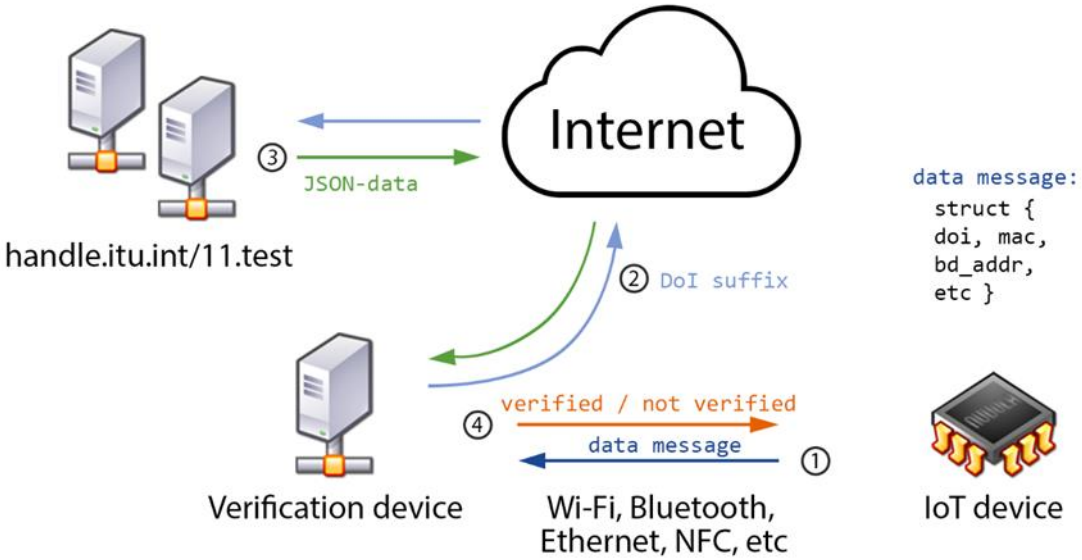


Рисунок 3.8 – Модернизированная концепция системы идентификации устройств на базе DoA с применением уровня верификации

Уровень верификации был представлен программно-аппаратным комплексом с набором сетевых интерфейсов, позволяющих подключать множество различных устройств, как путем непосредственного физического взаимодействия (технологии NFC), так и при помощи сетевого взаимодействия (BLE, WiFi).

Конечное устройство может представлять из себя как устройство интернета вещей, так и обычный объект, верификация которого необходима в каком-либо контексте.

Процесс верификации устройства с идентификатором DOA происходит поэтапно:

1) при помощи одного из доступных интерфейсов производится обращение к устройству верификации, которое включает в себя передачу массива данных, содержащего цифровой идентификатор объекта, а также данные, по которым непосредственно происходит проверка истинности объекта – MAC-адрес, BLE-адрес, дата продажи продукта, уникальный номер продукта и др.;

2) устройство верификации определяет необходимый сервер для обращения, отправляет запрос по заданному идентификатору объекта;

3) Handle-сервер отвечает на запрос JSON-массивом, содержащим необходимые поля, в том числе поля, отвечающие за проверку объекта на соответствие;

4) устройство верификации сравнивает полученные данные по заданным полям, выдаёт результат проверки (как на средство вывода информации, так и непосредственно на верифицируемое устройство).

Таким образом, устройство проходит проверку через строго заданные сервера DOA, защищенные от прямого доступа для обычных пользователей, не выдавая при этом данные по запрашиваемому идентификатору. Данный подход ограничивает возможные сценарии подделки устройств с цифровым идентификатором, одновременно разгружая конечное устройство. Полученная система в стационарном исполнении (в виде стенда) также позволяет наглядно продемонстрировать скорость процесса идентификации, маршрут следования служебного трафика и другие параметры.

Анализ результатов натурного эксперимента

Введение в традиционную схему устройства проверки позволит определить среднее время доступа системы к DOA-серверу и предоставит статус проверки. Доступ к проверочному устройству с использованием определенных технологий, таких как NFC или BLE, создает дополнительные задержки на интерфейсах, но это не целевой сценарий для данного исследования.

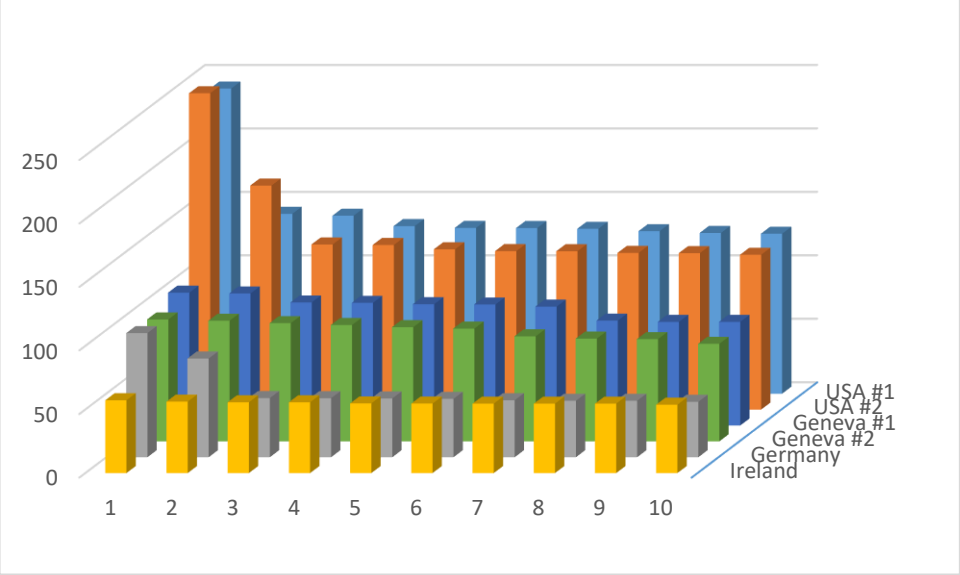


Рисунок 3.9 – График сетевой задержки при отправке запроса в разные страны

Задержки измерялись в двух случаях:

1) использование системы прокси-серверов CNRI, набор веб-серверов, которые понимают протокол обработки. Система состоит из четырех разных веб-серверов, размещенных в трех разных географических зонах;

2) использование основного сервера GHR, размещенного в Женеве.

В Таблице 3.3 содержится время задержки для каждого сервера и средняя задержка на основе десяти экспериментов. Каждый запрос выполнялся с использованием REST API, что позволяет обрабатывать запросы в формате JSON. Временные интервалы были получены с помощью программного обеспечения для захвата пакетов Wireshark [101].

Таблица 3.3 – Результаты измерения задержки с использованием разных handle-серверов

Местопо-ложение	Задержка, ms										
	1	2	3	4	5	6	7	8	9	10	Средняя
США #1	140,2	240,4	141,7	132,0	130,6	126,6	126,0	129,9	128,0	130,5	142,6
США #2	249,2	130,0	123,2	121,9	176,4	126,2	123,4	124,7	129,6	124,8	142,9
Германия	97,6	43,6	44,4	77,6	46,4	45,9	46,2	46,5	44,4	44,8	53,7
Ирланда	53,8	57,3	54,6	54,6	54,7	54,8	55,7	55,7	54,6	56,2	55,2
Женева #1	81,5	82,5	95,2	103,9	93,5	95,4	81,5	104,5	96,8	96,5	93,1
Женева #2	80,4	93,0	91,5	88,7	94,9	95,9	89,8	76,7	80,8	82,6	87,4

Как видно из Таблицы 1, наилучшее значение задержки наблюдается при обмене данными с сервером, расположенным в Германии, а худшее значение – с сервером, расположенным в США. Основываясь на этих значениях, мы можем сделать вывод, что для минимизации задержки необходимы оптимизировать маршруты для обращений к серверам GHR.

Выводы по главе 3

1. Проанализирован состав факторов, влияющих на идентификацию интернета вещей. Определены обобщены основные особенности идентификации для интернета вещей.

2. Предложена модель системы резолюции идентификаторов цифровых объектов как системы массового обслуживания, на базе которой выполнен оптимизационный эксперимент и получена конфигурация системы резолюции, позволяющая сократить время на разрешение идентификатора устройства. Система резолюций идентификаторов DOA была представлена в виде СМО.

3. Разработана имитационная модель, которая с заданным уровнем абстракции воспроизводит обмен данными между компонентами DOA. Проведенные эксперименты с имитационной моделью показали, что разрешение идентификатора в системе происходит гораздо быстрее на базе предлагаемого метода обращений к MPA. Прирост скорости в 15 раз достигается на максимальной интенсивности нагрузки сервера.

4. Разработана математическая модель системы резолюций. На базе полученной формулы Эрланга можно произвести численный расчет средней нагрузки *Lj* на реестрах GHR.

5. Проведен натурный эксперимент по исследованию задержки при передачи данных в системе архитектура цифровых объектов. Лабораторный стенд был разработан с введением нового компонента (в отличии от традиционного подхода) – уровня верификации объектов в системе DOA, что позволяет подключать множество различных устройств, как путем непосредственного физического взаимодействия (технологии NFC), так и при помощи сетевого взаимодействия (BLE, WiFi).

6. Анализ результатов натурного эксперимента показал, что наилучшее значение задержки наблюдается при обмене данными с сервером, расположенным в Германии, а худшее значение – с сервером, расположенным в США.

Глава 4. МЕТОД ИДЕНТИФИКАЦИИ УСТРОЙСТВ И ПРИЛОЖЕНИЙ ИНТЕРНЕТА ВЕЩЕЙ В ГЕТЕРОГЕННЫХ СЕТЯХ СВЯЗИ НА БАЗЕ АРХИТЕКТУРЫ ЦИФРОВЫХ ОБЪЕКТОВ

4.1. Взаимодействие устройств интернета вещей с архитектурой цифровых объектов

Одной из основных задач перед инженерами стоит предоставление доступа устройствам интернета вещей в Интернет как напрямую, так и с использованием шлюзов. Это необходимо для того, чтобы каждый объект (интернет вещь) был виртуально или физически представлен, имел адрес и был доступен через Интернет в любое время и в любом месте. В настоящее время разработка механизмов идентификации для различных классов и типов устройств продолжается как в Интернете вещей, так и в Промышленном Интернете вещей, а также является частью исследований в Международных организациях по стандартизации как государственных, так и коммерческих.

Существующие механизмы для идентификации, применяемые в различных технологиях передачи данных, были придуманы еще на рубеже веков, когда не поднимался вопрос о предполагаемом количестве подключаемых устройств ИВ. Анализ, проведенный в первой главе показал, что можно создать эффективную схему присвоения уникальных идентификаторов только для очень малого количества устройств интернета вещей. Кроме того, существующие методы идентификации в большинстве своем не поддерживают устройства ИВ, кратковременно подключаемые к сети Интернет, перемещающиеся между различными публичными и закрытыми сетями связи. Также к недостаткам данных методов можно отнести то, что в сетях связи идентификаторы могут содержать информацию с привязкой к конкретному местоположению устройств ИВ. Объекты интернета вещей должны иметь идентификаторы, не зависящие от того, в какой сети они находятся или каким пользователям принадлежат [56].

Другой вопрос, который необходимо учитывать – это концептуальная разница между идентификатором объекта и его сетевым адресом (или адресами). В самом общем случае идентификатор объекта и его адрес различны и служат для

разных целей. Первый обеспечивает уникальным идентификатором самого объекта, но сетевой адрес может меняться в зависимости от физического расположения объекта, его логического членства в одной или нескольких сетях или роли того или иного объекта. В случаях, когда идентификатор объекта и его адрес различны, идентификатор обычно структурирован по различным схемам идентификации. На Рисунке 4.1 приведена схема взаимодействия между идентификатором объекта и его сетевым адресом.

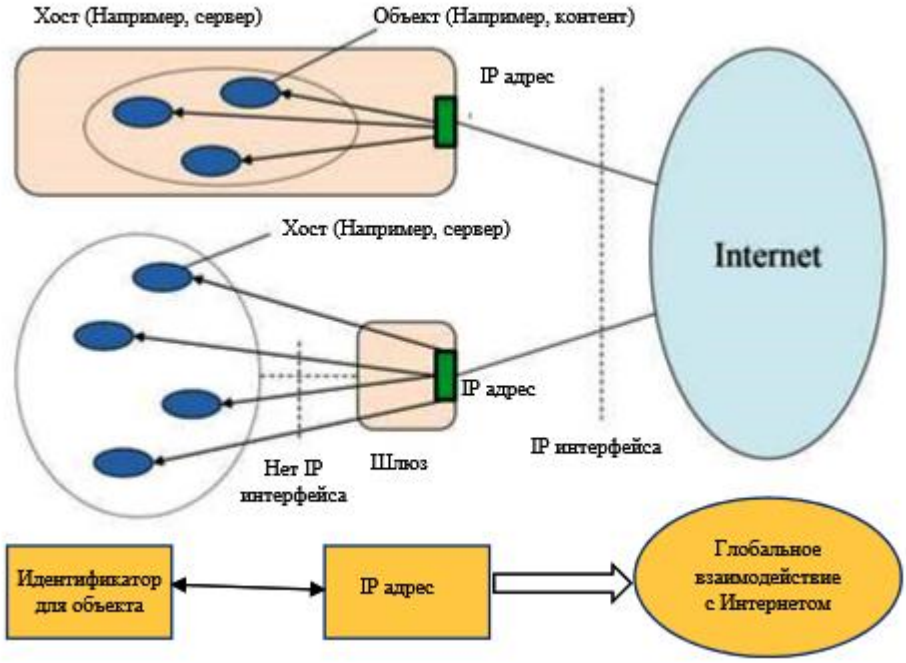


Рисунок 4.1 – Взаимодействие между идентификатором объекта и его сетевым адресом

Электронный код продукта (EPC) является одной из хорошо известных схем идентификации объектов, которые могут однозначно идентифицировать объекты, связанные с RFID-меткой. Другая схема идентификации, называемая повсеместно распространенным кодом (uCode), введенная центром uID в Японии, представляет собой другую систему кодирования, которая поддерживается исключительно в Японии и Азии. Аналогичным образом и другие схемы адресации могут быть разными. Объекты, которые в настоящее время подключены к Интернету, используют глобальную схему IP-адресации (IPv4 или IPv6). В свою очередь некоторые устройства ИВ не могут использовать глобальную IP-адресацию, ввиду того, что подключаются у Интернет через так называемый шлюз. Шлюзом,

например, может быть смартфон, который взаимодействует со смарт-часами с помощью технологии Bluetooth Low Energy. На Рисунке 4.2 представлена схема такого взаимодействия на примере сервиса фитнес трекинга.

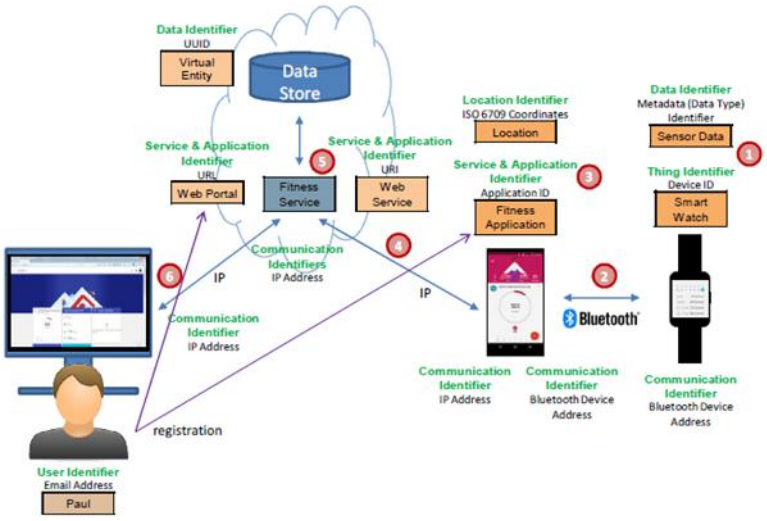


Рисунок 4.2 – Взаимодействие идентификаторов
в случае использования фитнес-трекинга

Одним из направлений обеспечения гарантированной идентификации устройств и приложений ИВ является архитектура цифровых объектов DOA.

Преимущества DOA по сравнению с имеющимися системами идентификации очевидны:

Преимуществом системы Handle по сравнению с системой DNS является более гибкая модель администрирования префиксов и лучшая масштабируемость при увеличении количества суффиксов в выбранном префиксе.

1. Уникальность – каждый хэндл уникален в рамках глобальной системы.

2. Постоянство – хэндлы могут использоваться в качестве постоянных идентификаторов для объектов в интернете. При этом хэндл не зависит от объекта, который он именует, их единственная связь – в самой системе. Это позволяет идентификатору существовать неизменным вне зависимости от изменений местоположения, владения и т.д. То есть при перемещении ресурса достаточно обновить его значение в Handle System для отражения нового местоположения.

3. Множественные экземпляры – хэндл может указывать на различные экземпляры ресурса, расположенные по различным сетевым адресам. Приложения

могут использовать это качество системы для повышения производительности и устойчивости.

4. Множественные атрибуты – хэндл может указывать на различные атрибуты ресурса, включая связанные сервисы, расположенные по различным сетевым адресам.

5. Расширяемое пространство имен – локальное пространство имен можно присоединить к глобальному, получив статус регистратора и уникальный префикс, чтобы избежать конфликтов с существующими именами. Использование регистраторов позволяет делегировать сервисы администрирования и резолюции локальным сервисам (local handle service), которым будет передавать запросы глобальный реестр (Global Handle Registry), создавая распределенную модель._

6. Модель безопасности – Handle System позволяет осуществлять безопасную резолюцию и администрирование. Протокол системы определяет стандартные механизмы клиентской и серверной аутентификации и авторизации, а также содержит функции проверки целостности данных и ограничения приватности.

7. Распределенный административный сервис – для каждого хэндла в системе можно определить собственного администратора (владельца). В комбинации с протоколом аутентификации это позволяет администратору безопасно управлять хэндлом через Интернет.

Принципом данного механизма идентификации является использование уникального идентификатора для каждого объекта интернета вещей. Причём как для уже существующих устройств ИВ, в которых используемый идентификатор может быть усилен с помощью DOA, так и новых устройств ИВ, в которых DOA-идентификатор может быть базовым для идентификации, прослеживаемости и борьбы с контрафактом.

В связи с этим одной из самых важных проблем является выбор системы идентификации для всех объектов, подключенных к сети Интернет. В качестве использования архитектуры цифровых объектов для идентификации вещей предлагается множество различных программных и аппаратных решений.

Для решения ряда задач, поставленных перед этапом повсеместного внедрения архитектуры цифровых объектов необходимо проанализировать методы интеграции и совместимости уникального DOA идентификатора в электронные устройства ИВ.

В качестве примеров базовых технологий передачи данных, применяемых для взаимодействия устройств интернета вещей с сетью Интернет (как напрямую, так и через шлюз) рассмотрим технологии: WiFi, ZigBee (IEEE 802.15.4) и LoRa (IEEE 802.15.4g). Перечисленные технологии передачи данных применяются в устройствах различного уровня: к примеру, отладочные платы устройств ИВ как правило имеют на борту радиомодуль WiFi (MAC-адрес), а технологии Fast и Gigabit Ethernet (семейство IEEE 802.3 – тоже использует MAC-адрес) в сетевых картах устройств на базе архитектуры x86 или x64, обладающих огромными вычислительными мощностями. В зависимости параметров устройств ИВ, определяется доступный функционал и методы добавления записей, однако базовые методы внесения идентификатора в устройства ИВ являются одинаковыми для всех (Рисунок 4.3).

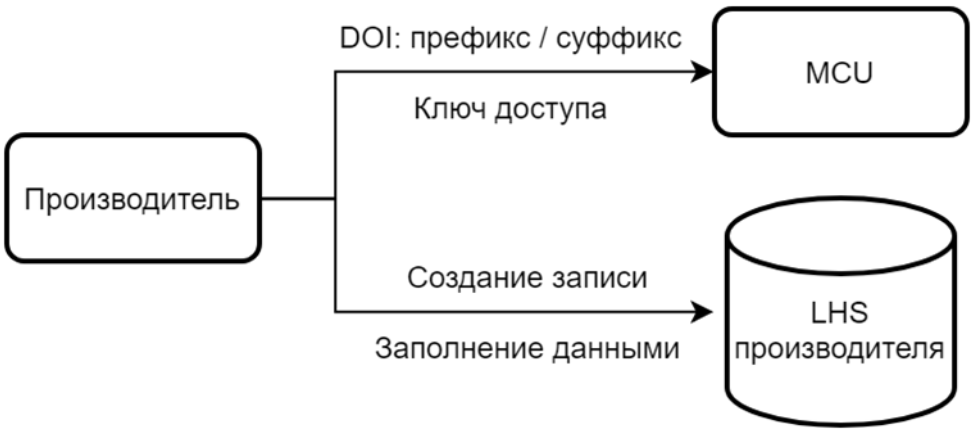


Рисунок 4.3 – Внесение базовой информации в устройство на микроконтроллере на этапе производства

В данном случае, на этапе производства каждое устройство ИВ, определяемое в глобальной системе резолюции, обязано иметь прописанный программными методами (по аналогии с существующими идентификаторами, такими как MAC или IMEI) цифровой идентификатор объекта и ключ для доступа к модификации метаданных идентификатора. Вписывание этих данных должно

сопровождаться созданием соответствующих handle-записей в LHS-базах производителя устройства ИВ. Согласно определению, идентификатор представляет собой серию цифр, букв и символов или данных в любой другой форме, используемую для идентификации абонентов, пользователей, элементов сети, функций, объектов сети, предоставляющих услуги/приложения, или других объектов (например, физические или логические предметы). Следовательно, наличие у микроконтроллера собственной цифровой копии в глобальной системе резолюции обусловлено возможностью создания универсальных методов для идентификации устройств интернета вещей. В качестве подобной информации, хранимой в уникальном для каждого выпущенного устройства домене, может выступать версия доступных протоколов устройства, привязка к иным технологиям идентификации, сопровождающая информация или даже базовые команды доступа для устройства ИВ. В общем случае, ключ доступ и DOI должны быть доступны для управляющих устройств. В случае наличия у управляющего устройства полноценной операционной системы, доступ к этим данным осуществляется при помощи драйвера ОС; если в качестве управляющего устройства выступает микроконтроллер, доступ к данным осуществляется через базовые команды микроконтроллера.

Примером использования (Рисунок 4.4) может быть ситуация, когда устройство 1 при помощи приложения осуществляет запись в доступное для устройства поле цифрового объекта «network_address» актуальный глобальный адрес в сети TCP/IP. Благодаря этому возможно взаимодействие двух различных устройств (1 и 2) без серверов-посредников, обычно оказывающих поддержку в установлении соединения. Для установления соединения достаточно иметь DOI устройства ИВ.

Указанный способ подходит для сложных устройств ИВ, содержащих как минимум два устройства на микропроцессоре, примером которых является большинство современных смартфонов. Невозможность реализации криптографических функций, необходимых для модификации данных в системе резолюции, а также отсутствие прямых методов доступа в глобальную сеть делают

невозможным реализацию подобных функций на простых устройствах (функционирующих на базе микроконтроллеров или микрочипов). Одновременно с этим, в случае использования сетей для устройств ИВ в приложениях типа Умный дом, использующих, к примеру, стек протокола ZigBee, данный пример не является реализуемым, т.к. прямого доступа к сети Интернет, и как следствие, к системе резолюции у конечных устройств нет. Реализация подобного функционала должна осуществляться через программное обеспечение шлюзов в совокупности с реализацией необходимого функционала в приложениях ZigBee.

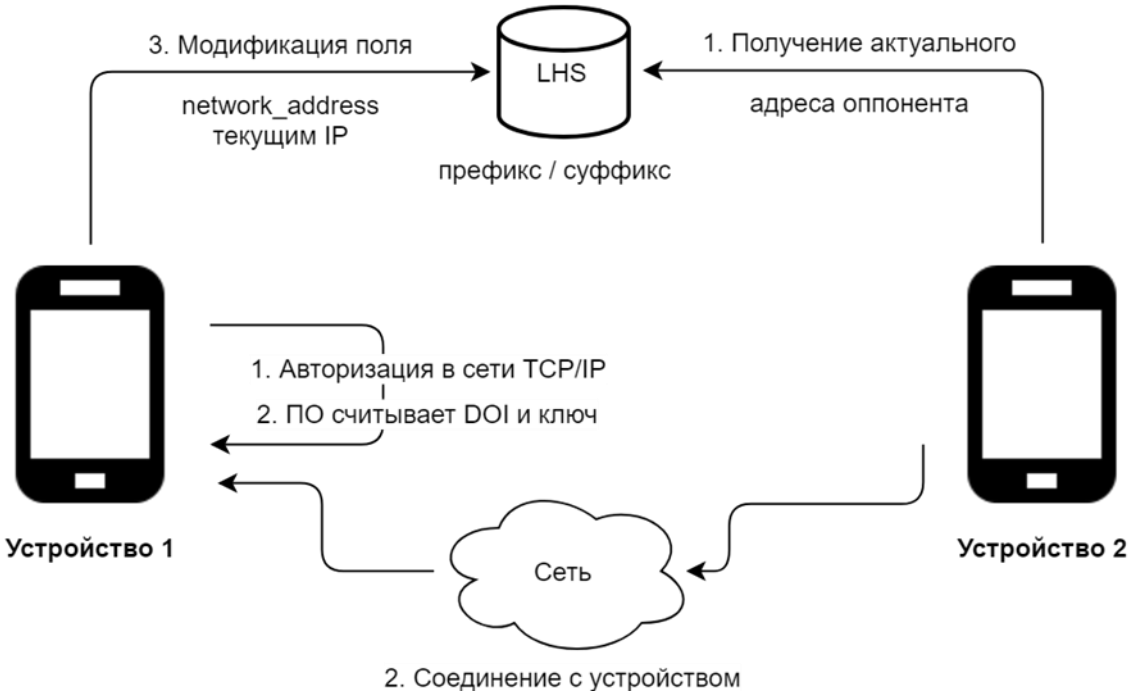


Рисунок 4.4 – Реализация полей цифрового объекта для хранения сетевого адреса устройства

Дополнительным функционалом, необходимым для реализации на устройствах ИВ одновременно с получением DOI и уникального ключа является возможность перезаписи таковых на произвольные с сохранением оригинала. Таким образом, производители устройств ИВ обязаны ограничивать максимальный объем данных, вносимый одним устройством ИВ в их домен, предполагая переназначение основного (но не исходного) идентификатора цифрового объекта и соответствующего ему ключа доступа сторонним, в целях увеличения объема метаданных и увеличения скорости доступа к данным. Стоит отметить, что важно сохранять исходный DOI, являющийся одним из возможных доказательств

подлинности устройств, с возможностью его возврата. Данный функционал может быть использован в сценариях борьбы с контрафактом устройств интернета вещей и не рассматривается в рамках диссертационной работы.

4.2. Описание лабораторного стенда для проведения натурного эксперимента

На Рисунке 4.5 изображена структура лабораторного стенда. В ходе эксперимент необходимо было протестировать возможность применения методов идентификации устройств интернета вещей на базе архитектуры цифровых объектов. Эксперимент проводился на базе модельной сети лаборатории Интернета Вещей Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича.

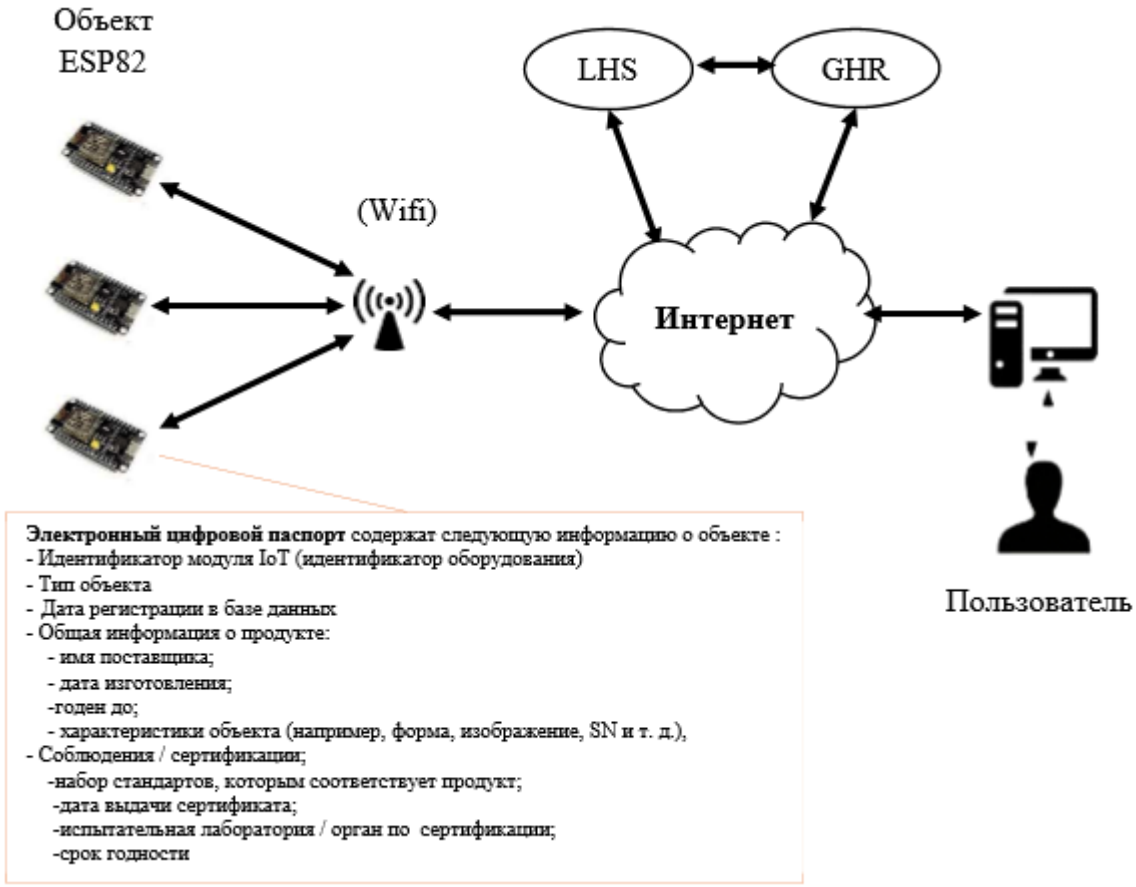


Рисунок 4.5 – Структура лабораторного стенда для тестирования применения методов идентификации устройств интернета вещей на базе архитектуры цифровых объектов

Идентификация цифрового объекта происходит следующим образом:

– пользователь (либо пользовательское устройство) производит запрос на получение информации об объекте, с помощью имеющегося открытого идентификатора DOI;

– данный запрос обрабатывается сервисом GHR, который отправляет запрос в систему реестров, затем в репозиторий, где находиться информация о LHS отвечающих за запрашиваемый цифровой объект с помощью технологии интернет вещей (в нашем случае использовалась технология WiFi);

– данная информация отправляется пользовательскому устройству, которое устанавливает защищенное соединение с сервисом LHS, при помощи алгоритмов асимметричного шифрования;

– после обработки данного запроса LHR отправляет данные об объекте пользовательскому устройству.

Идентификатор цифрового объекта используется для определения информации о состоянии самого объекта, которое может включать местоположение объекта, методы аутентификации, открытые ключи и другие соответствующие данные. Поскольку цифровой объект по существу представляет собой строку битов, которые могут быть идентифицированы однозначно, тогда часть объекта также может быть идентифицируемой. Таким образом, когда устройство интернета вещей используется как объект, его можно идентифицировать. На Рисунке 4.6 представлена структура метаданных, которые могут использоваться в архитектуре цифровых объектов для устройств интернета вещей для подтверждения оригинальности.

Как видно из Рисунка 4.6, на каждом уровне существуют данные, которые могут быть предварительно заноситься в базу данных на серверах LHS, а впоследствии использоваться для сопоставления с поступающими запросами.

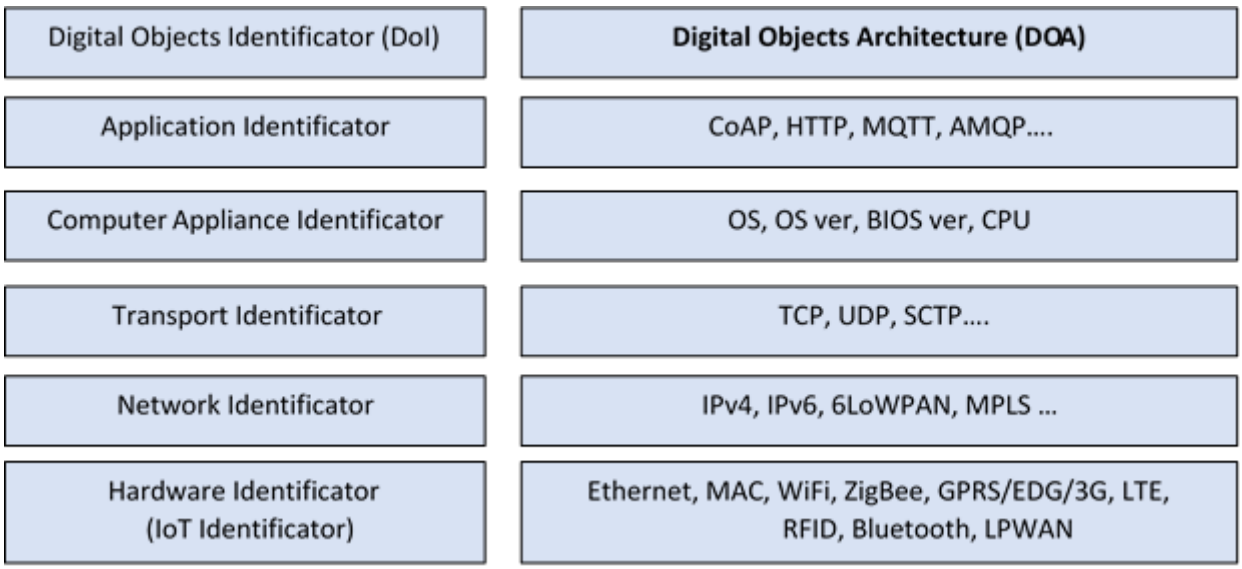


Рисунок 4.6 – Примерная структура архитектуры цифрового объекта, который основан на тегах IoT

4.3. Аспекты сетевого взаимодействия при реализации метода идентификации устройств интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов

В предложенной схеме, которая базируется на подходах, представленных в Рекомендации МСЭ-Т Y.4459 «Архитектура для взаимодействия IoT», определен минимальный набор необходимых архитектурных компонентов, протоколов и услуг, которые в совокупности формируют структуру, реализующую общие информационные функции и совместимость услуг. Ядром структуры является архитектура цифровых объектов, в котором любая информация, представленная в цифровом виде, может быть структурирована в качестве цифрового объекта, которому присваивается глобально уникальный идентификатор [4;5].

Данный идентификатор возвращает вспомогательную информацию цифрового объекта, которая существует в системе резолюции независимо от изменений, сделанных в цифровом объекте какой-либо сущностью. Данная вспомогательная информация может включать в себя местоположение, метаданные, контрольные суммы, цифровые подписи, сертификаты, публичные ключ и др. Составляющие сопровождающей информации, связанные с цифровым объектом, рассматриваются в качестве атрибутов цифрового объекта. Цифровой объект используется также в качестве представления сущностей ИВ, к примеру, поверх стека протоколов TCP/IP, в независимости от технологий передачи данных.

Представленная архитектура позволяет любой цифровой информации, предварительно структурированной в качестве цифрового объекта, быть безопасно идентифицированной, независимо от конкретной системы, сервиса или приложения, где информация создавалась или хранилась. Как известно, DOA состоит из трёх базовых фундаментальных компонентов, которые реализуют следующие услуги: сервис глобальной идентификации, сервис хранения цифровых объектов, сервис-регистр цифровых объектов. На Рисунке 4.7 показан процесс создания цифрового объекта и присвоения атрибутов.

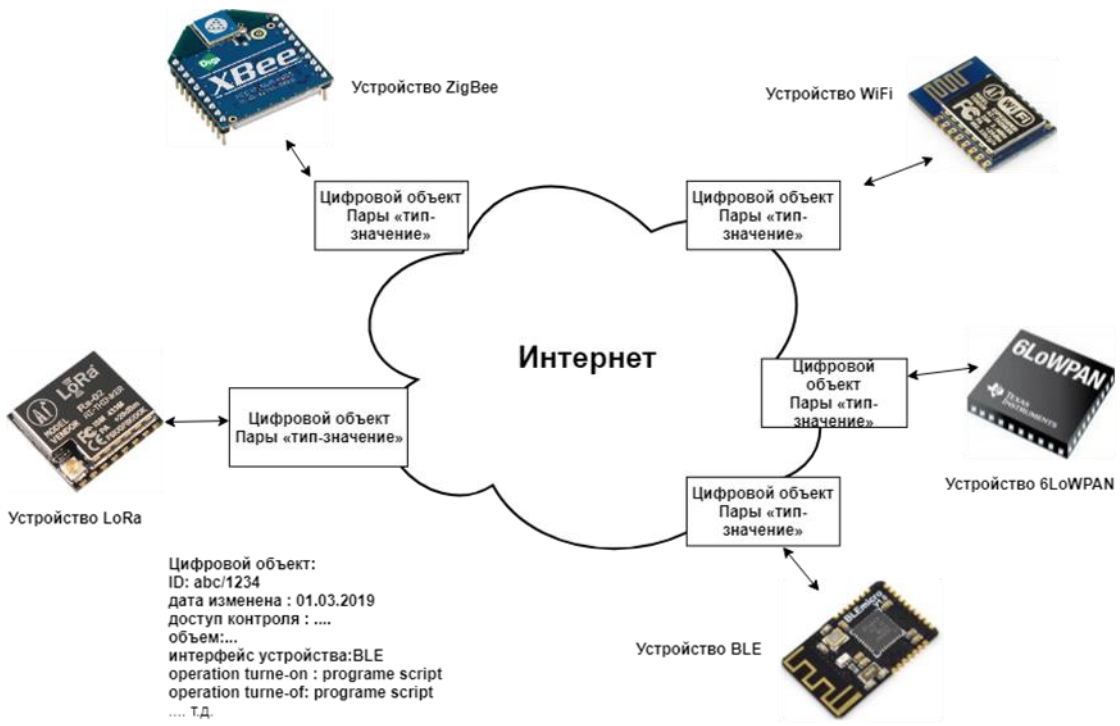


Рисунок 4.7 – Процесс создания цифрового объекта и присвоения атрибутов

Сервис глобальной идентификации позволит назначать глобальный идентификатор любому цифровому объекту. Данный сервис предоставляет протокол резолюции и администрирования, предназначенный для определения связанной с цифровым объектом вспомогательной информации: место хранения, происхождение информации, с возможностью извлечения и управления с соблюдением необходимых мер безопасности. Сервис идентификации должен являться распределенной системой с встроенными механизмами защиты для обеспечения целостности сервиса, его безотказности, целостности хранимых данных. Обязательным является также аутентификация и конфиденциальность

операций с хранимыми данными, наличие избирательного управления доступом для любых метаданных, связанных с идентификатором. Набор распределенных сервисов для хранения цифровых объектов способствует безопасному хранению, доступу и распространению объектов с использованием их идентификаторов. Само хранилище является цифровым объектом, которое может хранить внутри себя другие объекты (что не является обязательным)[54]. Цифровой объект может выполнять определенный набор действий, включая осуществление доступа к другим цифровым объектам, создание новых цифровых объектов и др. Хранилище цифровых объектов может представлять из себя набор устройств ИВ, при этом являющихся также цифровыми объектами.

Цифровой объект может иметь множество атрибутов, связанных с реальным объектом. Часть атрибутов может описывать природу устройства ИВ. В частности, объект может обладать управляющими атрибутами, которые связаны с программным обеспечением, предоставляя прямое взаимодействие с функциями устройства ИВ, например, включение или выключение системы, получение показаний температурного сенсора на устройстве. Помимо этого, цифровой объект может также иметь атрибуты, определяющие доступность основных атрибутов устройств, таким образом определяя, кто может взаимодействовать с устройством ИВ при помощи интерфейса, описанного в атрибутах объекта. На Рисунке 4.8 представлена схема взаимодействия устройств ИВ, подключаемых к сети связи с использованием различных технологий передачи данных, и компонентов архитектуры цифровых [44].

Структура цифрового объекта может быть сформирована в виде цифрового представления физического устройства ИВ. Система компонентов, а именно реестр, имеет возможности для определения способов нахождения и доступа к подобным сущностям. Критерий совместимости в терминах Интернета Вещей подразумевает наличие API, чтобы цифровые объекты могли взаимодействовать с устройствами, к которым они привязаны. Данный подход может быть использован для достижения конкретных средств управления доступом для удобства каждого хранилища. С другой стороны, хранилище может предоставлять доступ к данным,

генерируемым отдельным устройством ИВ. Архитектура цифровых объектов не ограничивает количество возможных хранилищ.

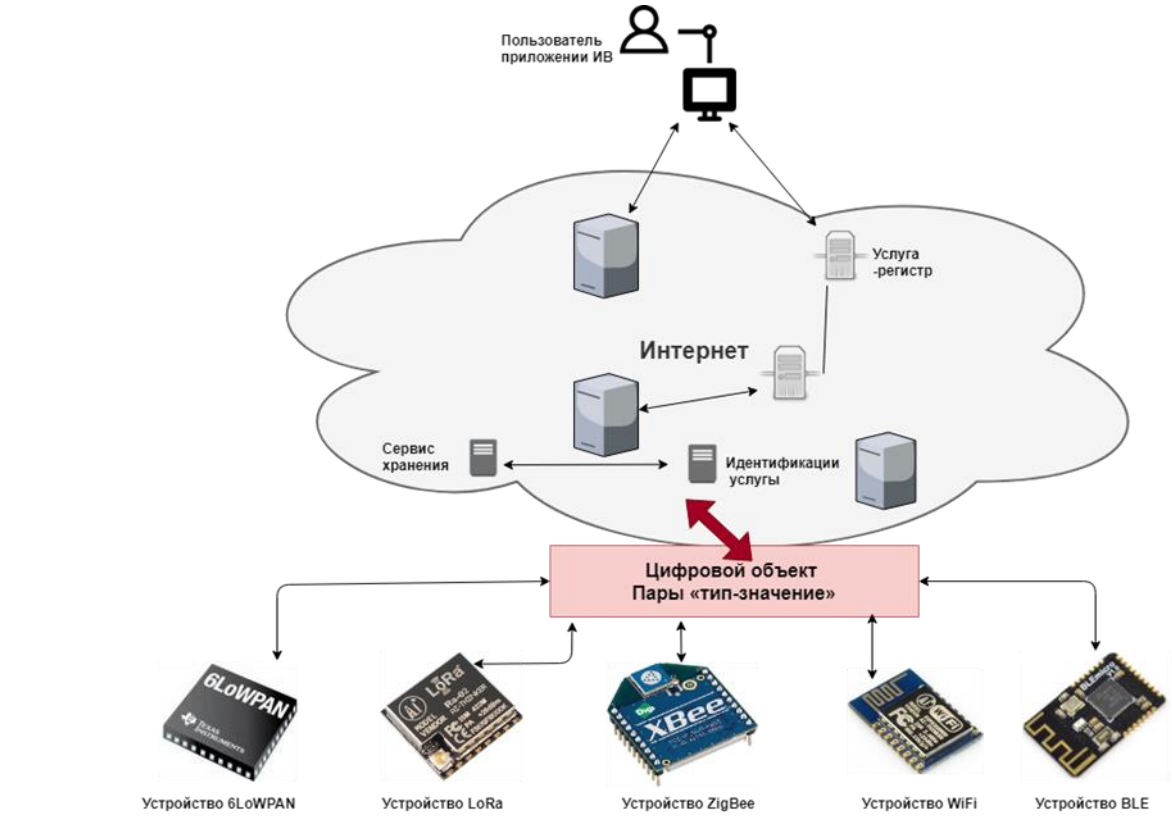


Рисунок 4.8 – Схема взаимодействия устройств ИВ и компонентов архитектуры цифровых

Набор сервисов реестров цифровых объектов в пределах одной области (входящей в состав МРА) позволяет обнаруживать любой цифровой объект. Реестр цифровых объектов может также предоставлять возможность поиска по метаданным или простым данным в цифровом объекте[118]. Использование реестра позволяет обнаруживать цифровые объекты по различным критериям, например:

1) поиск по разным типам записей метаданных цифровых объектов в пределах различных сервисов-реестров;

2) поиск по разным уровням сетевого взаимодействия в пределах различных сервисов-реестров;

3) поиск по разным видам сервисов по управлению данными;

4) поиск по различным типам безопасности и системам контроля доступа. Политика доступа, включающая в себя аутентификацию и авторизацию клиентских запросов, применяемая на множество MPA, должна быть явно определена.

4.4. Аспекты совместимости при реализации метода идентификации устройств интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов

В настоящее время продукты, подходы и инициативы, связанные с вопросом совместимости ИВ пока что находятся стадии в разработке. Пропритарный характер выпуска устройств ИВ наблюдался до 2017 года, что создало предпосылки и необходимость задуматься создание требований по совместимости устройств как для задач совместного взаимодействия, так и для задач совместной идентификации т.е. чтобы одно устройство ИВ понимало другое устройство ИВ. Таковым примером являются системы, построенные на базе общего уровня услуг и реализациях сущностей общих услуг, сервис-ориентированных архитектур, архитектур цифровых объектов и системы доменных имён [2].

Существуют различные типы совместимости, одна из которых - техническая совместимость – обычно связана с программно-аппаратными компонентами, системами и платформами, позволяющим реализовать межмашинное взаимодействие. Этот тип совместимости чаще всего основан на коммуникационных протоколах и инфраструктуре, необходимой для работы данных протоколов. Другой тип совместимости – синтаксическая совместимость – обычно связана с форматами данных. Несомненно, что все сообщения, передаваемые при помощи протоколов должны иметь явно определенный синтаксис и способы кодировки, даже если они имеют вид битовых таблиц. Третьим видом совместимости является семантическая совместимость, обычно связанная со смыслом содержания, затрагивающая человеческую интерпретацию содержимого, нежели машинную. Таким образом, совместимость на этом уровне означает общее понимание смысла обмениваемого содержимого (информации).

Большинство перечисленных подходов сконцентрированы на определении набора базовых интерфейсов в пределах различных приложений. Архитектура

цифровых объектов определяет базовый набор услуг, позволяющих произвести инкапсуляцию информации, её регистрацию и обнаружение, вне зависимости от границ применения, тем самым позволяя осуществлять обмен информацией в пределах различных приложений. Описанные подходы могут быть интегрированы в любом из ранее перечисленных аспектов совместимости (общий уровень услуг, сервис-ориентированная архитектура и др.), что позволит осуществлять обмен информацией в пределах различных приложений.

Архитектура цифровых объектов определяет минимальный набор необходимых архитектурных компонентов, протоколов и сервисов для обеспечения общей информации, и совместимости сервисов. Приведенное здесь описание является технологически нейтральным, а также может быть реализовано с существующими технологиями для задач глобальной идентификации ИВ. Это облегчит совместимость идентификации, описания, представления, доступа, хранения и безопасности устройств ИВ.

4.4.1. Описание структуры типового устройства ИВ и процесса резолюции на базе архитектуры цифровых объектов

Наличие сервиса идентификации, включающего процесс резолюции, является ключевым требованием к системам ИВ и одновременно базовым принципом DOA. Архитектура цифровых объектов, тем не менее, имеет особое требование к идентификаторам внутри архитектуры, а именно возможность резолюции идентификатора в метаданные об объекте, резолюция которого происходит, или, как в нашем случае, об объекте ИВ. Система контроля доступа, реализованная на серверах, обеспечивает доступ только к определенным значениям в метаданных [1].

Внедрение DOA для устройств Интернета Вещей подразумевает назначение каждому устройству ИВ собственного идентификатора (handle) [50].

Одной из первоочередных целей, которая может быть реализована на базе внедрения DOA в ИВ является борьба с контрафактом в продуктах и сервисах ИВ. Пользователь получает возможность проверки характеристик устройства ИВ при помощи системы Handle. Конечный пользователь может извлечь префикс

идентификатора при помощи специальной технологии, сделав запрос на сервер GHR для определения расположения сервиса LHS, располагающего непосредственной информацией о устройстве ИВ. GHR отвечает на подобный запрос сообщением, содержащим адрес запрашиваемой LHS. Устройство осуществляет новый запрос по полученному адресу, на что в ответ получает сообщение с данными по конкретному идентификатору, что включает в себя всю необходимую информацию, которая контролируется и модифицируется только производителем устройства. На Рисунке 4.9 представлена структура и процедуры проверки устройством ИВ на предмет выявления контрафакта на базе архитектуры цифровых объектов.

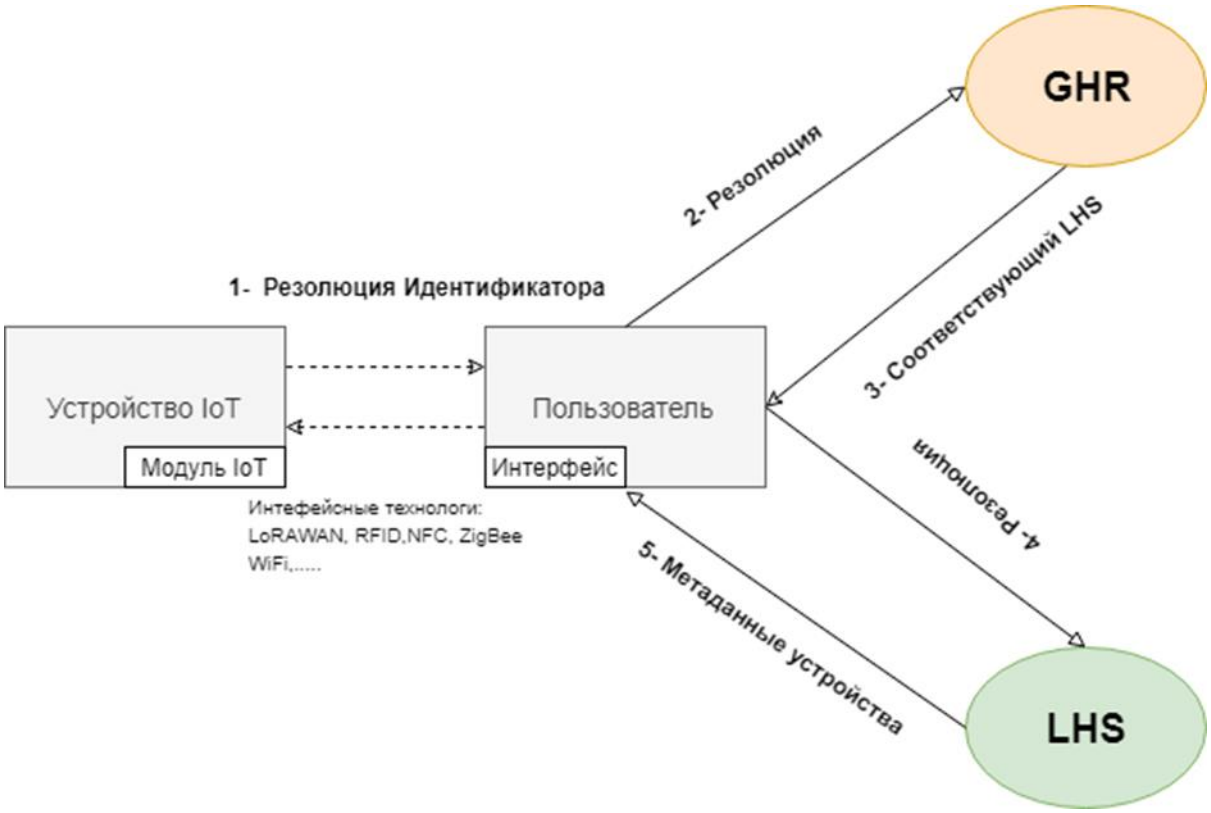


Рисунок 4.9 – Структура и процедуры проверки устройством ИВ на предмет выявления контрафакта

Результат разрешения должен быть получен в форме пары «тип – значение».

В структуре данных DOA, цифровой объект представляет из себя файл, сервис, базу данных, устройство или комбинацию перечисленных [111].

Ниже представлен перечень характеристик, описывающих системы идентификации DOA, которые могут быть использованы в контексте ИВ:

1. Возможность интеграции с существующими системами идентификации, такими как OID (X.6660) или любыми другими существующими системами идентификации, в данный момент используемыми производителями устройств ИВ для обеспечения повсеместной совместимости в процессе разрешения идентификаторов [3].

2. Верификация существования и уникальности. Для подтверждения существования конкретного идентификатора и его уникальности необходим реестр и система резолюции. Данная характеристики является ключевой в процессе подтверждения глобальной уникальности любого нового устройства ИВ.

3. Сервис идентификации структурирован таким образом, чтобы обеспечить действительный безотказный функционал, который может подтвердить источник определенного сервиса резолюции. Важнейшим является наличие механизма, позволяющего использовать методы доступа сервисам или клиентам для осуществления аутентификации устройства ИВ, с которым они взаимодействуют. Безотказность достигается наличием значений, подписанных цифровым объектом, при помощи ключа объекта, от имени устройства ИВ.

4. Возможность использования шифрования. Устройству ИВ может потребоваться наличие определенных уровней шифрования для исключения угрозы рисков от атак, подобных атакам «Человек посередине» («Man in the middle»).

5. Контроль доступа. Устройствам ИВ может потребоваться возможность контроля доступа для управления сущностями, которые могут обращаться к информации об устройстве или к непосредственно устройству.

6. Средство для ассоциации вспомогательной информации с идентификатором каждого цифрового объекта/устройства ИВ. Данная информация будет возвращена последующим запросом разрешения в виде набора пар «тип-значение». «Тип» представляет из себя непосредственно идентификатор DOA, которые возвращает набор описаний типа. Гибкость данного процесса позволяет производителям в индустриях ИВ использовать их личные системы для описания

собственных устройств ИВ, при этом позволяя всему сообществу ИВ получать информацию, необходимую для обработки определенную пару «тип-значение».

Примерами типов и/или атрибутов являются:

– тип устройства ИВ;

– описание устройства ИВ, как для понимания человеком, так и для понимания машиной;

– сопровождающая информация, включает в себя информацию о состоянии устройства ИВ (местоположение, статус и др.);

– спецификации интерфейса. Каждое устройство ИВ предоставляет идентификатор, который уникально определяет интерфейсы, используемые для взаимодействия с другими системами и устройствами ИВ. К примеру, подобная спецификация интерфейса предоставляет низкоуровневое описание физических интерфейсов, используемых для взаимодействия с устройством;

– доступ к сервисному интерфейсу. Каждое устройство ИВ, доступное в сети, может предоставлять рекомендации к сервисным интерфейсам. Для некоторых устройств это может быть URI, для других может быть простейший IP-адрес, как и любые другие варианты. Конкретное устройство ИВ может иметь различные сервисные интерфейсы;

– взаимосвязь с архивным хранилищем устройства ИВ. Множество устройств ИВ могут генерировать данные во время работы. DOA может связать устройство с хранилищем подобных данных.

4.4.2. Доступ к устройствам интернета вещей с поддержкой идентификации на базе архитектуры цифровых объектов

Для обеспечения доступности устройств ИВ им необходим стандартный интерфейс для чтения и записи данных, установки параметров, диагностики специфичных операций для конкретного устройства. Данные параметры обязательно варьируются от устройства к устройству т.к. многообразие устройств ИВ очень широко [81,102].

Посредством фундаментальных компонентов системы глобальной идентификации, хранилища цифровых объектов и сервиса реестра цифровых

объектов, архитектура цифровых объектов может обеспечить поддерживаемый доступ к устройству ИВ. Данный функционал может быть реализован посредством любого стандартного легковесного протокола, который использует понятия глобально получаемых типов для уточнения доступных операций. Результатом является возможность прямого доступа к данным или сервису при помощи особого типа запроса на управление или доступа. Не менее важно, что данный способ доступа является обнаруживаемым, а функционал понятен любому взаимодействующему клиенту в рамках архитектуры цифровых объектом.

Набор простых действий, основанных на типах, призван обеспечить поддержку всех типов устройств. Преимущества данного подхода в том, что DOA предоставляет гибкую модель данных на базе цифровых объектов, обеспечивающую базовый, полностью настраиваемый и расширяемый, подход для обеспечения доступа к данным любого устройства ИВ. DOA поддерживает доступ к устройству ИВ при помощи любого протокола, что позволяет выполнять любые устройство-зависимые операции. К примеру, устройство ИВ даёт возможность исполнения особых действий для установки точных калибровочных параметров, конфигурирования устройства, или добавление записи во внутреннюю собственную базу данных. Производитель может определить подобные действия путём создания нового цифрового объекта для представления информации об устройстве, с уникальным идентификатором и связанным описанием действия в сопровождающей информации. Когда клиент осуществляет запрос к устройству ИВ при помощи базового протокола, с целью получения списка возможных действий, выполняемых устройством, устройство ИВ возвращает тот особый тип действия, свойственный для данного производителя. При помощи системы типов, являющейся свойственной для архитектуры цифровых объектов, клиент может определить тип операций, узнать, являются ли данные действия полезными для использования. Данная свойственная расширяемость протоколов может быть использована любыми производителями устройств ИВ для разработки новых типов действий, при этом осуществляя поддержку старых типов действий.

4.5. Метод модификации архитектуры цифровых объектов для повышения сетевой безопасности

В этом разделе представлено описание метода модификации архитектуры цифровых объектов с целью повышения доверия и безопасности при обмене данными между элементами архитектуры цифровых объектов и устройствами ИВ. Целью предлагаемого метода и соответствующей математической модели является защита конфиденциальности ИВ в домене приложений.

1. Ключевые нотации:

1) $\Psi_{S,D}$ – общий ключ безопасности, передаваемый между исходным узлом *S* и узлом назначения *D*;

2) Γ_S – открытый ключ шифрования исходного узла *S*;

3) Φ_S – закрытый ключ шифрования исходного узла *S*;

4) $M_S^{\Gamma}=\{b_1, b_1, b_1, \ldots, b_n\}$ – сообщение, зашифрованное исходным узлом S с использованием его открытого ключа Γ_S;

5) $M_S^{\varphi}=\{b_1, b_1, b_1, \ldots, b_m\}$ – сообщение, зашифрованное узлом-источником *S* с использованием его закрытого ключа Φ_S;

6) $[M_S^{\varphi}]^{-1}=\{b_1, b_1, b_1, \ldots, b_m\}^{-1}$ – сообщение, дешифрованное узлом-источником с использованием его закрытого ключа Φ_S;

7) $<M_1, M_2, \ldots, M_n>$ – набор заданных сообщений $M_1, M_2, M_3, \ldots, M_n$;

8) $S \rightarrow D$: *M* – исходный сервер *S* отправляет сообщение *M* на целевой сервер *D*;

9) Rand (*i*) – генерация случайного числа со скоростью *i*;

10) R_{GHR} – скорость генерации случайных чисел, генерируемых сервером GHR и используемых для шифрования и дешифрования сообщений;

11) R_{LHR} – скорость генерации случайных чисел, генерируемых сервером локального регистра LHR (Local Handle Register) и используемых для шифрования и дешифрования сообщений;

12) ID_{LHR} – идентификационный номер LHR;

13) L_{GHR-} Буфер, используемый GHR

14) L_{LHR-} Буфер, используемый LHR

15) Q_n – номер запроса;

16) γ – ключ шифрования LHR;

17) $П_{GHR}$ – открытый ключ GHR;

18) Mess-Size(*M*) – метод расчета дайджестов сообщений и цифровых подписей (дайджест сообщения – это числовое представление фиксированного размера содержимого сообщения, вычисляемое хеш-функцией).

Предлагаемая схема повышения безопасности для системы резолюции.

Серверы LHR и GHR имеют собственные предварительно определяемые ключи $Ψ_{LHR,\ GHR}$, используемые при обмене сообщениями и в процессах аутентификации. Каждый сервер (LHR и GHR) генерирует случайный номер с предопределенной частотой R_{LHR} и R_{GHR} соответственно. Сгенерированные номера используются для шифрования и дешифрования.

$$L_{GHR} = \text{Rand}(R_{HGR}),$$

$$L_{LHR} = \text{Rand}(R_{LHR}).$$

Сервер LHR выполняет последовательность действий, которая объединяет идентификационный номер сервера LHR со случайно сгенерированным номером с предопределенной частотой и заданным номером запроса.

$$M_1^{\gamma_1} = \{\text{ID}_{LHR}, L_{LHR}, Q_n\}.$$

Сообщение M_1 шифруется при помощи публичного ключа $γ_1$. Затем сервер LHR определяет необходимый набор данных сообщений для обмена с централизованным реестром GHR:

$$LHR \rightarrow GHR : < M_1^{\gamma_1}, M_2^{\gamma_1}, M_3^{\gamma_1}, \dots, M_n^{\gamma_1} >.$$

Реестром GHR получает набор данных и расшифровывает сообщение M_1 с помощью предварительно общего ключа. Расшифрованное сообщение выглядит следующим образом:

$$[M_1^{\gamma_1}]^{-1} = \langle \text{ID}_{LHR}, L_{LHR}, Q_n \rangle.$$

Затем сервер GHR извлекает идентификатор LHR для проверки прав на доступ к серверу:

$$\text{ID}_{\text{host, app-serv}} = \text{ID}_{\text{host}} \oplus L_{GHR},$$

$$\text{ID}_{\text{IP, app-serv}} = Q_n \,||\, (\text{ID}_{\text{host, app-serv}} \oplus L_{LHR}).$$

Далее сервер GHR вычисляет хеш-сумму и цифровую подпись. Хеш-сумма, которая рассчитывается при помощи хеш-функции, является числовым представлением содержимого сообщения. Ее длина заранее определена:

$$\Pi_{\text{GHR}} = \left[\left(\text{Mess} - \text{Size}\left(\text{ID}_{\text{IP, app-serv}} \right)^{\Gamma_{\text{GHR}}} \right) \right]^{-1},$$

$$\text{Mess} - \text{Size}\left(\text{ID}_{\text{IP, app-serv}} \right) = \Pi_{\text{GHR}} .$$

Сервер GHR возвращает Π_{GHR} и $\text{ID}_{\text{IP, app-serv}}$ на сервер LHR, что является подтверждением подписей.

$$\text{ID}_{\text{IP, act}} = Q_n \,||\, \left(\text{ID}_{\text{host, app-serv}} \oplus L_{\text{LHR}} \oplus L_{\text{LHR}} \right) = Q_n \,||\, \text{ID}_{\text{host, app-serv}} \ .$$

Для решения вопросов безопасности на участке между серверами LHR и GHR в представленной системе Handle определены два типа сообщений, участвующих в обмене. Сообщение первого типа передается сервером LHR к серверу GHR и шифруется при помощи предварительно определяемого ключа. Сообщение второго типа передается от сервера GHR к серверу LHR, оно содержит цифровые подписи.

Представленная модель безопасности в предлагаемой системе Handle использует минимальное число сообщений для обеспечения процесса аутентификации. Данная схема эффективна для применения в устройствах ИВ, т.к. она позволяет сократить общий объем передаваемых данных и одновременно уменьшить сетевые задержки в процессе обеспечения безопасности.

4.6. Перспективы внедрения идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов

Как было показано в Главе 4 архитектура цифровых объектов может стать базовой архитектурой в аспектах идентификации устройств ИВ. Сущности (субъекты) в каждом из приложений ИВ, включая умные устройства, сервисы приложений, пользователи приложений, зарегистрированные в приложениях ИВ, могут быть структурированы в цифровые объекты. Каждый цифровой объект

может иметь свой глобальный уникальный идентификатор, который также может быть связан с набором атрибутов, описывающие нижележащую сущность.

К примеру, умное устройство, использующееся в ИВ, может получить свой собственный глобальный идентификатор с атрибутами, определяющими владельца, методы доступа и различные сервисные интерфейсы для осуществления связи с устройством [95;123].

Глобальный идентификатор позволяет осуществлять доступ к устройству и обнаруживать его не только в целях исходного приложения ИВ, но и другими приложениями, желающими взаимодействовать с устройством. Права владения и контроль доступа, определенные цифровым объектом, могут обеспечить безопасный доступ к устройству во множестве приложений ИВ без потери необходимых систем защиты. Интерфейс, взаимодействующий с устройством, может быть обнаружен «на лету», без ограничения для оригинального приложения.

Схожим образом, информация о сущности пользователя, зарегистрированной с множеством приложений ИВ, также может быть структурирована в качестве цифрового объекта с присвоением ему глобально уникального идентификатора. Таким образом, информация о пользователе может быть применена для множества различных приложений ИВ с целью авторизации и аутентификации пользователя в сервисе. Совместное использование личности пользователя во множестве различных приложений ИВ может упростить процесс взаимодействия пользователя с множеством отдельных приложений ИВ, при этом расширяя границы взаимодействия различных приложений ИВ между собой.

В дополнение к сказанному, сервис приложения, реализованный отдельным приложением ИВ может быть также структурирован в качестве цифрового объекта, тем самым делая его доступным для других приложений ИВ. Для сервиса приложений, цифровой объект будет иметь глобально уникальный идентификатор, который может быть использован для создания ссылки на сервис. Также, сервис будет иметь набор атрибутов, полностью его описывающий, включая административные и сервисные интерфейсы, контроль доступа над доступными действиями.

Сервисы приложений, представленные в качестве цифровых объектов, могут содействовать обмену между сервисами и интеграции между различными приложениями ИВ. Новые приложения ИВ могут быть разработаны путем интеграции различных сервисов приложений из различных существующих приложений. Подобная интеграция должна способствовать более открытому обмену данными между разными приложениями ИВ. При реализации с учетом данных рекомендации, каждое приложение ИВ может сохранить существующий опыт, одновременно позволяя сервисным данным и ресурсам быть доступными через публичный, но при этом безопасный, интерфейс, определенный и управляемый отдельным приложением [103;104].

Набор сервисов, поддерживаемых архитектурой цифровых объектов, включает в себя умные устройства, личности пользователя, сервисы приложений, представляют из себя цифровые объекты с целью обнаружения, совместного использования, получения доступа безопасными методами в пределах различных приложений ИВ. Данный подход позволяет отделить информационные сущности от границ, в пределах которых они хранятся, в которых используются и организуются, тем самым делая шаг в сторону обеспечения совместимости в приложениях ИВ [105;110].

Описанные в Рекомендации МСЭ-Т X.1255 «Структура обнаружения информации по управлению определением идентичности» открытая архитектура и представленное выше описание, которое базируется на подходах, представленных в Рекомендации МСЭ-Т Y.4459 «Архитектура для взаимодействия IoT» позволят достичь совместимости, вне зависимости от особенных реализаций, в гетерогенных сетях связи, с возможностью контроля за данными цифрового объекта в собственном информационном домене.

Провайдеры сервисов хранилищ могут развёртывать конфигурации и политики с целью определения, какие именно сущности могут осуществлять контроль над хранилищем в целях безопасности [106,107;108].

Сервис идентификации архитектуры цифрового объекта может быть использован для определения, какая информация может быть использована

совместно, с кем информацией можно делиться, и каким именно образом информация будет передана. Совместимые приложения могут быть разработаны путем взаимодействия с каждым отдельным приложением, предоставляя сервис интегрированной совместимости, основанный на данных реального времени из каждого отдельного приложения.

Выводы по главе 4

1. Представлен метод идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов, который позволяет идентифицировать устройства и приложения ИВ в глобальном масштабе.

2. Рассмотрены методы интеграции идентификатора DOA в устройства интернета вещей, поддерживающих различные технологии беспроводной передачи данных и представлена структура метаданных, которые могут использоваться в архитектуре цифровых объектов для устройств интернета вещей для подтверждения оригинальности в совокупности с традиционными идентификаторами.

3. Представлены аспекты сетевого взаимодействия и совместимости устройств и приложений интернета вещей с интегрированными идентификаторами DOA в гетерогенных сетях связи.

4. Представлено описание структуры типового устройства ИВ и процесса резолюции на базе архитектуры цифровых объектов. Рассмотрены типовые примеры реализации описанных методов и взаимодействие устройства интернета вещей с компонентами архитектуры цифровых объектов.

5. С целью повышения безопасности и конфиденциальности передаваемых данных предложен при обмене служебными сообщениями в рамках структуры цифровых объектов предложена математическая модель. Модель позволяет использовать минимальное число сообщений (за счет оптимизации процессов обмена данными) для обеспечения процесса аутентификации. Предложенная схема эффективна для применения в устройствах ИВ, т.к. она позволяет сократить общий

объем передаваемых данных и одновременно уменьшить сетевые задержки в процессе обеспечения безопасности.

6. Рассмотрены перспективы внедрения идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов. Показано, что предлагаемая интеграция должна способствовать более открытому обмену данными между разными устройствами и приложениями ИВ.

ЗАКЛЮЧЕНИЕ

На текущий момент активное развитие получила система сквозного глобального цифрового идентификатора, реализованного на базе архитектуры цифровых объектов. Как отмечается в диссертации, использование идентификатора на базе DOA позволит учитывать все существующие уникальные идентификаторы и адреса (например, MAC, IMEI, ID, IPv4/IPv6 и др.), обеспечив сквозную идентификацию устройств и приложений Интернета вещей без привязки к конкретному идентификатору. Это позволит реализовать глобальную и подлинно международную систему идентификации, так как она реализовывается при поддержке МСЭ.

Реализация DOA в системе DOI (Digital Object Identification) определена стандартом ГОСТ Р ИСО 26324-2015, поэтому ее внедрение может осуществляться быстрыми темпами.

В ходе выполнения настоящей диссертационной работы автором получены следующие основные результаты:

1. Проанализированы различные системы идентификации, их архитектура, структура идентификаторов и примеры их использования в повседневной жизни.

2. Проведенный обзор международной деятельности по исследованиям идентификации в концепции Интернета вещей показал, что в настоящее время отсутствуют прикладные исследования, посвященные идентификации устройств и приложений Интернета вещей на базе архитектуры цифровых объектов.

3. В аналитическом обзоре показано, что до настоящего времени отсутствовали работы, в которых была бы подробно проанализирована и исследована архитектура цифровых объектов как метод идентификации устройств и приложений Интернета вещей.

4. Технология DOA позволяет осуществлять однозначную персистентную идентификацию объектов, в которой заинтересованы правообладатели этих объектов. Это делают целесообразным развитие применения технологии DOA как транснациональную систему идентификации с равными правами для всех членов.

5. Проведен анализ построения сетевой архитектуры цифровых объектов. Рассмотрены основные компоненты архитектуры DOA и принципы их взаимодействия.

6. Проанализированы служебные протоколы DOA и особенности их функционирования. Показаны отличия текущих версий протоколов IRP и DOIP как по структуре, так и по функциональному назначению.

7. Предложена модель системы идентификации на базе архитектуры цифровых объектов, отличающаяся от известных тем, что для обеспечения приемлемого качества обслуживания в общей архитектуре сетей связи общего пользования, существующей сегодня, была разработана новая архитектура взаимодействия путем введения регистра промежуточного уровня (Middle Handle Register – MHR) между глобальным регистром (Global Handle Register – GHR) и локальным регистром (Local Handle Register – LHR).

8. На базе предложенной математической модели был проведен численный анализ, который показал, что предлагаемая система обеспечивает более высокую производительность с точки зрения сетевой задержки, ввиду уменьшения расстояния между серверами LHR, что достигается путем развертывания регистров промежуточного уровня обработки (MHR).

9. Результаты моделирования показали, что введение промежуточного уровня регистров MHR позволит снизить задержку на 60% по сравнению с существующей архитектурой.

10. Проанализирован состав факторов, влияющих на идентификацию интернета вещей. Определены обобщены основные особенности идентификации для интернета вещей.

11. Предложена модель системы резолюции идентификаторов цифровых объектов как системы массового обслуживания, на базе которой выполнен оптимизационный эксперимент и получена конфигурация системы резолюции, позволяющая сократить время на разрешение идентификатора устройства. Система резолюций идентификаторов DOA была представлена в виде СМО.

12. Разработана имитационная модель, которая с заданным уровнем абстракции воспроизводит обмен данными между компонентами DOA. Проведенные эксперименты с имитационной моделью показали, что разрешение идентификатора в системе происходит гораздо быстрее на базе предлагаемого метода обращений к MPA. Прирост скорости в 15 раз достигается на максимальной интенсивности нагрузки сервера.

13. Разработана математическая модель системы резолюций. На базе полученной формулы Эрланга можно произвести численный расчет средней нагрузки Lj на реестрах GHR.

14. Проведен натурный эксперимент по исследованию задержки при передачи данных в системе архитектура цифровых объектов. Лабораторный стенд был разработан с введением нового компонента (в отличии от традиционного подхода) – уровня верификации объектов в системе DOA, что позволяет подключать множество различных устройств, как путем непосредственного физического взаимодействия (технологии NFC), так и при помощи сетевого взаимодействия (BLE, WiFi) [121]. Анализ результатов натурного эксперимента показал, что наилучшее значение задержки наблюдается при обмене данными с сервером, расположенным в Германии, а худшее значение – с сервером, расположенным в США.

15. Представлен метод идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов, который позволяет идентифицировать устройства и приложения ИВ в глобальном масштабе.

16. Рассмотрены методы интеграции идентификатора DOA в устройства интернета вещей, поддерживающих различные технологии беспроводной передачи данных и представлена структура метаданных, которые могут использоваться в архитектуре цифровых объектов для устройств интернета вещей для подтверждения оригинальности в совокупности с традиционными идентификаторами.

17. Представлены аспекты сетевого взаимодействия и совместимости устройств и приложений Интернета вещей с интегрированными идентификаторами DOA в гетерогенных сетях связи.

18. Представлено описание структуры типового устройства ИВ и процесса резолюции на базе архитектуры цифровых объектов. Рассмотрены типовые примеры реализации описанных методов и взаимодействие устройства Интернета вещей с компонентами архитектуры цифровых объектов.

19. С целью повышения безопасности и конфиденциальности передаваемых данных предложен при обмене служебными сообщениями в рамках структуры цифровых объектов предложена математическая модель. Модель позволяет использовать минимальное число сообщений (за счет оптимизации процессов обмена данными) для обеспечения процесса аутентификации. Предложенная схема эффективна для применения в устройствах ИВ, т.к. она позволяет сократить общий объем передаваемых данных и одновременно уменьшить сетевые задержки в процессе обеспечения безопасности.

20. Рассмотрены перспективы внедрения идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов. Показано, что предлагаемая интеграция должна способствовать более открытому обмену данными между разными устройствами и приложениями ИВ.

Стоит отметить, что на момент написания диссертационной работы разработка Рекомендации МСЭ-Т Y.4459 «Архитектура для взаимодействия IoT» была закончена, однако её принятие откладывается в связи с отсутствием консенсуса между представителями различных государств и организаций – членов МСЭ. Несомненно, представленный подход должен удовлетворять требованиям всех заинтересованных участников рынка Интернета вещей, что позволит обеспечить совместимость устройств и приложений Интернета вещей на глобальном уровне.

В заключении хотелось бы отметить, что в Санкт-Петербургском Государственном университете телекоммуникаций им. проф. МА. Бонч-Бруевича

на кафедре Сетей связи и передачи данных, в Лаборатории Интернета вещей ведется разработка моделей и методов применения архитектуры цифровых объектов для задач идентификации устройств и приложений Интернета вещей, разработаны лабораторные стенды и собран научный коллектив, который глубоко понимает процессы взаимодействия всех компонентов DOA. Предлагаемые в диссертации методы, модели и подходы позволят модернизировать существующую инфраструктуру DOA и обеспечить лучшие параметры ее функционирования.

СПИСОК СОКРАЩЕНИЙ

ИВ – Интернет вещей

СМО – система массового обслуживания

ССОП – сеть связи общего пользования

БСС – беспроводная сенсорная сеть

МСЭ – Международный союз электросвязи

ПО – программное обеспечение

МСЭ-Т – Сектор стандартизации электросвязи Международного союза электросвязи. МЭК – Международная электротехническая комиссия

DOA – Digital Object Architecture – архитектура цифровых объектов

DTN – Delay-Tolerant Networking – сети, толерантные к задержкам

DOA– Digital Object Architecture – архитектура цифровых объектов

DOI– Digital Object Identifier - цифровой идентификатор объекта

IP– Internet Protocol – интернет протокол

TTL– Time To Live – время жизни

URI – Uniform Resource Identifier - унифицированный идентификатор ресурса

XRI – Extensible Resource Identifier - расширяемый идентификатор ресурса

GHR – Global Handle Register

LHR – Locally Handle Register

LHS – Locally Handle Service

IPv6 – Internet Protocol version 6 – интернет-протокол версии 6

IPv4 – Internet Protocol version 4 – интернет-протокол версии 4

IETF – Internet Engineering Task Force – Инженерный совет Интернета

LoRA – Long Range wide-area networks – технология беспроводной связи с большим радиусом действия

LPWAN – Low-power Wide-area Network – крупномасштабные малопотребляющие сети

QE – Qualified Equipment – устройство тестирования

RFID – Radio Frequency Identification – радиочастотная идентификация

6LoWPAN – IPv6 over Low power Wireless Personal Area Networks – стандарт взаимодействия по протоколу IPv6 поверх беспроводных персональных сетей с низким энергопотреблением стандарта IEEE 802.15.4.

СЛОВАРЬ ТЕРМИНОВ

Адрес: адрес является идентификатором определенной конечной точки и используется для маршрутизации к этой конечной точке.

Атрибут: информация, привязанная к объекту, которая определяет характеристику объекта.

Вещь: Применительно к интернету вещей означает предмет физического или информационного мира, который может быть идентифицирован и интегрирован в сети связи.

Идентификатор: последовательность битов, используемая для получения информации о состоянии идентифицируемого цифрового объекта; как правило, это делается с помощью соответствующей системы разрешения.

Интернет вещей: концепция развития сетей связи, представляющая глобальную инфраструктуру для информационного общества, которая обеспечивает возможность предоставления услуг путем соединения друг с другом (физических и виртуальных) вещей на основе существующих и развивающихся функционально совместимых информационно-коммуникационных технологий.

Инфраструктурный (или управляемый) режим работы сети: Режим взаимодействия узлов сети, при котором обмен информацией между двумя узлами всегда происходит посредством инфраструктурных элементов (точек доступа для беспроводных сетей, коммутаторов и маршрутизаторов — для беспроводных).

Метаданные: структурированная информация, которая относится к личности пользователей, систем, служб, процессов, ресурсов, информации или других объектов.

Объект: все, что имеет отдельное и отличное существование, которое может быть однозначно идентифицировано. В контексте IdM примеры объектов включают в себя абонентов, пользователей, элементы сети, сети, программные приложения, услуги и устройства. Сущность может иметь несколько идентификаторов.

Сенсор: устройство, предназначенное для сбора данных об окружающей среде.

Сенсорный узел: устройство, состоящее из сенсора, блока обработки информации, блока приема и передачи данных по беспроводному каналу, а также источника питания. Сенсорный узел является частью сенсорной сети и предназначен для фиксации состояния окружающей среды, обработки полученных данных и передачи их шлюзу сенсорной сети.

Совместимость: способность двух или более систем, или приложений обмениваться информацией и взаимно использовать полученную информацию.

Устройство: что касается Интернета вещей, это часть оборудования с обязательными возможностями связи и опциональными возможностями зондирования, активации, сбора данных, хранения данных и обработки данных.

Цифровой объект: объект, представленный или преобразованный в независимую от машины структуру данных, состоящую из одного или нескольких элементов в цифровой форме, которые могут анализироваться различными информационными системами; структура помогает обеспечить взаимодействие между различными информационными системами в Интернете. Цифровой объект иногда

называют цифровым объектом. Основным фиксированным атрибутом DE является связанный с ним уникальный постоянный идентификатор, который может быть преобразован в текущую информацию о состоянии DE, включая его местоположение (я), средства управления доступом и проверку достоверности, путем отправки запроса разрешения в систему разрешения. Примерами других внутренних атрибутов элемента DE являются: дата последнего изменения, дата создания и размер.

СПИСОК ЛИТЕРАТУРЫ

1. Recommendation ITU-T T.181203 : An architecture for IoT interoperability. – Geneva: ITU-T, 2018 – 25.

2. Recommendation ITU-T Y.2066 :Common requirements of the Internet of things. – Geneva: ITU-T, 2014 – 12.

3. Recommendation ITU-T X.1255: Framework for discovery of identity management information. – Geneva: ITU-R, April 2013. – URL: http://handle.itu.int/11.1002/1000/1195.

4. Recommendation ITU-T E.164: The international public telecommunication numbering plan. – November 2010, [Online]. URL: https://www.itu.int/rec/T-REC-E.164-201011-I.

5. Recommendation ITU-T P.10 : Vocabulary for performance and quality of service. – ITU-T, 2006. – 12.

6. ISO, ISO 3779 :Road vehicles – Vehicle identification number (VIN) Content and structure, 2009.

7. ISO, ISO 6346: Freight containers – coding, identification and marking, 1995.

8. ISO, ISO 11784: Radio frequency identification of animals. – Code structure, 1996.

9. ISO/IEC JTC1, ISO/IEC 15963: Information technology. – Radio frequency identification for item management. – Unique identification for RF tags, 2009.

10. ISO, ISO 13584: Industrial automation systems and integration. – Parts library: Logical resource: Information supplier identification, 2000. – 26.

11. ISO, ISO 6709: Standard representation of geographic point location by coordinates, 2008.

12. ISO 3166-1: Codes for the representation of names of countries and their subdivisions : Country codes, 2013. – 1.

13. ISO 17442: Financial services – Legal Entity Identifier (LEI), 2012.

14. ISO/IEC JTC1, ISO/IEC 15459: Series Information technology – Automatic identification and data capture techniques–Unique identification, 2014

15. ISO 26324: 2012 Information and Documentation-Digital Object Identifier System //BSI British Standards. – 2012.

16. IEEE 802.3: Standard for Ethernet: [Online]. Available: http://ieeexplore.ieee.org/document/7428776/ , 2015.

17. IEEE 802: Standard for Local and Metropolitan Area Networks: Overview and Architecture", [Online] Available: http://ieeexplore.ieee.org/document/6847097/ , 2014.

18. OneM2M, TS-0001: Functional Architecture“,V2.10.0, [Online] Available: http://www.onem2m.org/images/files/deliverables/Release2/TS-0001-%20Functional_ Architecture-V2_10_0.pdf – 2016

19. BEREC, Report Enabling the Internet of Things, 12.02.2016, [Online]. – URL: http://berec.europa.eu/eng/document_register/subject_matter/berec/ download/0/5755-berecreport-on-enabling-the-internet-of_0.pdf – 2017

20. Standardisation A. W. G. T. High Level Architecture (HLA) //Technical specification. – 2017.

21. China Academy of Telecommunication Research (CATR) & Research Cluster on the Internet-of Things (IERC): EU-China Joint White Paper on Internet-of-Things Identification“, 31.10.2014, [Online] Available: http://www.miit.gov.cn/newweb/n1146312/n1146909/n1146991/n1648536/c3489529/part/3489530.pdf –2017

22. AIOTI WG01:Report on Internet of Things Applications: [Online] https://aioti.eu/aioti-wg01-report-on-internet-of-things-applications – 2017

23. AIOTI WG05:Report on Smart Living Environment for Ageing Well: https://aioti.eu/aioti-wg05-report-on-smart-living-environment-for-ageing-well –2017

24. AIOTI WG06 :Report on Smart Farming and Food Safety Internet of Things Applications: [Online]: https://aioti.eu/aioti-wg06 report-on-smart-farming-and-food-safety-internetof-things-applications –2017

25. AIOTI WG07: Report on Wearables: [Online]: https://aioti.eu/aioti-wg07-report-on-wearables –2017

26. AIOTI WG08: Report on Smart Cities: [Online], Available: https://aioti.eu/aioti-wg08-report-on-smart-cities –2017

27. AIOTI WG09: Report on Smart Mobility: [Online] Available: https://aioti.eu/aiotiwg09-report-on-smart-mobility –2017

28. AIOTI WG11: Report on Smart Manufacturing:[Online]: Available: https://aioti.eu/aiotiw11-report-on-smart-manufacturing – 2015

29. GS1 EPC Tag Data Standard“, Release 1.11, September 2017, [Online], Available: https://www.gs1.org/epcrfid-epcis-id-keys/epc-rfid-tds/1-11 [Accessed 02.01.2018]

30. ETSI, GS LTN 002 V1.1.1. : Low Throughput Networks (LTN): Functional Architecture: [Online], Available: http://www.etsi.org/deliver/etsi_gs/LTN/001_099/002/01.01.01_60/gs_LTN002v010101p.pdf_ – 2014

31. IETF, RFC 791: Internet Protocol – DARPA Internet Program Protocol Specification: [Online] Available: https://tools.ietf.org/html/rfc791 –1981

32. IETF, RFC 4291: IP Version 6 Addressing Architecture”, [Online]. Available: https://tools.ietf.org/html/rfc4291 –2017

33. RFC 8200: IETF, Deering S., Hinden R. Internet protocol, version 6 (IPv6) specification. – 2017.

34. IETF, RFC 3968 „Uniform Resource Identifier (URI): Generic Syntax“, 2005, [Online] Available: https://tools.ietf.org/html/rfc3968 .

35. European Commission, General Data Protection Regulation (GDPR), [Online] Available: https://www.eugdpr.org/ .

36. IETF, RFC 5322: Internet Message Format: 2008, [Online] Available: https://tools.ietf.org/html/rfc5322

37. IETF, RFC 4122 :A Universally Unique IDentifier (UUID) URN Namespace“, 2005, [Online] Available: https://tools.ietf.org/html/rfc4122

38. UN Centre for Trade Facilitation and E-business, „UN/LOCODE United Nations Code for Trade and Transport Locations“, [Online]. Available: http://www.unece.org/cefact/locode/welcome.html

39. IETF, RFC 7252 :The Constrained Application Protocol (CoAP): 2014, [Online] Available: https://tools.ietf.org/html/rfc7252

40. GS1 : Object Name Service“, 2013, [Online] Available: https://www.gs1.org/sites/default/files/docs/epc/ons_2_0_1-standard-20130131.pdf

41. IETF RFC768: User Datagram Protocol – August 1980.

42. Аль Бахри М. С. Машинное обучение как метод для идентификации устройств IoT анализа сетевого трафика / М.С. Аль-Бахри, Р.В. Киричек // 73-я Всероссийская научно-техническая конференция, посвященная Дню радио – СПб.: СПбГЭУ «ЛЭТИ» им. В. И. Ульянова (Ленина), 2018. – С. 214–215.

43. Аль Бахри М.С. Моделирование системы идентификации устройств интернета вещей на базе архитектуры цифровых объектов / М.С. Аль-Бахри, Р.В. Киричек, Д.Д.Сазонов // Труды учебных заведений связи, 2019. Т. 5. № 1. С. 42–47.

44. Аль Бахри, М. С. Обзор внедрения технологии SigFox в государстве Оман / М. С. Аль Бахри, Р. В. Киричек // 72-я Всероссийская научно-техническая конференция, посвященная Дню радио. – СПб.: СПбГЭУ «ЛЭТИ» им. В. И. Ульянова (Ленина), 2017. – С. 172–173.

45. Аль Бахри, М. С. Обзор методов децентрализованного хранения данных для Интернета вещей / М. С. Аль Бахри, Р. В. Киричек // 71-я Всероссийская научно-техническая конференция, посвященная Дню радио. – СПб.: СПбГЭУ «ЛЭТИ» им. В. И. Ульянова (Ленина), 2016. – С. 190–191.

46. Аль Бахри, М. С. Исследование взаимодействия фрагмента беспроводной сенсорной сети с сетью связи общего пользования на базе шлюза LTE / М. С. Аль Бахри, Р. В. Киричек // Актуальные проблемы инфотелекоммуникаций в науке и образовании. IV Международная научно-техническая и научно-методическая конференция: сборник научных статей в 2 томах. – СПб.: Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича, 2015. – С. 174–178.

47. Аль Б. М. С. ЭНЕРГОЭФФЕКТИВНОСТЬ И ПОКРЫТИЕ LPWAN (SIGFOX) КАК БАЗОВЫЙ КОМПОНЕНТ ЧЕТВЁРТОЙ ПРОМЫШЛЕННОЙ РЕВОЛЮЦИИ / М.С. Аль-Бахри, Р.В. Киричек //М75 Молодежная научная школа по прикладной теории вероятностей и телекоммуникационным технологиям

(АРТСТ–2017)= 2nd International School on Applied Probability Theory & Communications Technologies. – 2017. – С. 17.

48. Аль-Бахри М.С Архитектура цифровых объектов как основа идентификации устройств Интернета Вещей в сетях 5G/IMT-2020/ М.С. Аль-Бахри, Р.В. Киричек. – INTHITEN 2017

---

49. Аль-Бахри, М.С. Архитектура цифровых объектов как основа идентификации в эпоху цифровой экономики / М.С. Аль-Бахри, Р.В. Киричек, А.С. Бородин // Электросвязь. – 2019. – No 1. – С. 12–22.

50. Аль-Бахри, М.С. Метод идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуры цифровых объектов / М.С. Аль-Бахри // Электросвязь. – 2019. – No 4. – С. 73–79.

51. Блинова, Е.А. Стенаграфический метод на основе изменения межсрочного расстояния неотображаемых символов электроннного текстового документа / Е.А. Блинова // Труды БГТУ, 2016. – с 166–169.

52. Борисенко, Б.Б. Цифровые водяные знаки в изображениях. // Журнал Информационная безопасность" #4, 2007. – с 36– 38.

53. Бородин, А.С. Сети связи пятого поколения как основа цифровой экономики / А.С. Бородин, А.Е. Кучерявый // Электросвязь. – 2017. – № 5. – С. 45–49.

54. Владимиров, С.С. Методика идентификации устройств Интернета вещей на основе принудительной деградации участка флеш-памяти / С.С. Владимиров, Р.В. Киричек // Электросвязь. – 2017. – № 2. – С. 32–35.

55. Гуда А. Н. Модели оценки параметров телекоммуникационного трафика в автоматизированных информационно-управляющих системах/ А. Н. Гуда, Бутакова, М. А. Н. А. Москат //Вопросы современной науки и практики. Университет им. ВИ Вернадского. – 2010. – №. 4–6. – С. 71–86.

56. Данилов К. Н. Методы идентификации и аутентификации устройств Интернета Вещей/ К. Н. Данилов, Р. В. Киричек, В. А Кулик // Информационные технологии и телекоммуникации. 2016. № 4. С. 49–57.

57. Данилов, К. Н. Исследование методов идентификации и аутентификации устройств Интернета вещей / К. Н. Данилов, В. А. Кулик, Р. В. Киричек // Информационные технологии и телекоммуникации. – 2016. – № 3 (4). – С. 49–57.

58. Данилов, К. Н. Методы обнаружения интернет вещей в глобальной сети / К.Н. Данилов, Р.В. Киричек, В.А. Кулик // Информационные технологии и телекоммуникации. – 2015. – № 4. – С. 48–56. – http://www.sut.ru/doci/nauka/review/4-15. pdf

59. Деарт, В.Ю. Исследование параметров качества обслуживания (QoS), определяющих качество восприятия пользователем (QoE) потокового видео при передаче через Интернет / В.Ю. Деарт, И.С. Кожухов // TComm. – 2013. –№7.

60. Иваненко В.Г., Ушаков Н.В. Встраивание цифровых водяных знаков в видеозаписи/ В.Г. Иваненко, Н.В. Ушаков // Безопасность информационных технологий, 2016 –№4, с. 21–24

61. Иваненко, В.Г. Цифровые водяные знаки в электронном документообороте/ В.Г. Иваненко, Н.В. Ушаков // Безопасность информационных технологий, 2017. – №3, с. 37–42.

62. Измерительная инфраструктура для изучения качества соединений в российском сегменте Интернет, Телекоммуникации, 2009, № 1, С. 11–16. Безопасность информационных технологий, 2016 – №1, с.50–52

63. Карташевский, В.Г. Влияние механизмов управления QOS на показатели качества обслуживания мультемедийного трафика сети Internet / В.Г. Карташевский, М.А. Буранова // T-Comm. – 2013. –№8. – с. 54–6

64. Киричек, Р. В. Модельные сети для Интернета вещей и программируемых сетей / Р. В. Киричек, А. Г. Владыко, М. В. Захаров, А. Е. Кучерявый // Информационные технологии и телекоммуникации. – 2015. – № 3. – С. 17–26.

65. Киричек, Р.В. Ложные облака для интернета вещей. Методы защиты / Р.В. Киричек, В.А. Кулик, А.Г. Владыко и др. // Информационные технологии и телекоммуникации. – 2015. – № 3. – С. 27–39. – http://www.sut.ru/doci/nauka/review/3-15. pdf

66. Критерией Стьюдента [Электронный ресурс]. – Режим доступа: http://neuromatix.pro/2015/04/30/potreb-neuro/.http://neostom.ru/osnovi-sanitarnoy-statistiki/koeffitsient-dostovernosti-i-ego-primenenie.html

67. Кулик, В.А. Методы аутентификации устройств Интернета вещей для локальных и домашних сетей / В.А. Кулик, Р.В. Киричек, А.Е. Кучерявый // 71-я Всероссийская научно-техническая конференция, посвященная Дню радио. – СПб.: СПбГЭТУ «ЛЭТИ», 2016. – С. 206–207.

68. Кучерявый, А.Е. Интернет Вещей / А.Е. Кучерявый // Электросвязь. – 2013. – № 1. – С. 21–24.

69. Кучерявый, А.Е. Самоорганизующиеся сети / А.Е. Кучерявый, А.В. Прокопьев, Е.А. Кучерявый. – СПб.: «Любавич», 2011.

70. Лушникова А. А. ОСОБЕННОСТИ АРХИТЕКТУРЫ МУЛЬТИСЕРВИСНОЙ СЕТИ / А. А. Лушникова, Е. Д. Бычков //Наука, образование, бизнес. – 2014. – С. 322–324.

71. Мартьянова, А.И. Обзор и сравнительный анализ методов идентификации устройств Интернета Вещей / А.И. Мартьянова, В.А. Кулик, А.С. Бородин, Р.В. Киричек // 72-я Всероссийская научно-техническая конференция, посвященная Дню радио. – СПб., 2017. – С. 219–221.

72. Медриш, М.А. Стабильность, безопасность, отказоустойчивость глобальной инфраструктуры интернета: технические и правовые вопросы. 2016. р 2–3.

73. Назаров А.Н. Модели и методы расчёта показателей качества функционирования узлового оборудования и структурно-сетевых параметров сетей связи следующего поколения / А.Н. Назаров, К.И. Сычев. – Красноярск: Полииздат, 2010. – 390 с.

74. Присвоение DOI. [Электронный ресурс]. – Режим доступа: https://www.ru-science.com/ru/blog/publikaciya-nauchnyh-statej.../prisvoenie-doi

75. Руководство программиста JavaBeans / Inprise Application Server VERSION 4.0

76. Сагайдак Д.А. Способ формирования цифрового водяного знака для физических и электронных документов/ Д.А. Сагайдак, Р.Т. Файзуллин //Компьютерная оптика, 2014. Том 38, №1

77. Самойлов, М.С. Анализ трафика в современных телекоммуникационных сетях / М.С. Самойлов // Труды XII МНТК «Проблемы техники и технологий телекоммуникаций». – Казань, 2011. – с.214-215.

78. Султанов Т.Г. Полукаров, Измерения сетевой полосы пропускания по данным о задержке пакетов// Т.Г. Султанов, А.М. Сухов // Измерительная инфраструктура для изучения качества соединений в российском сегменте Интернет, Телекоммуникации, 2009, № 1, С. 11–16.

79. Султанов Т.Г. Критерий качества сетевого обслуживания на основе измерений доступной пропускной способности/ Т.Г. Султанов, А.М. Сухов // Известия Самарского научного центра Российской академии наук, т.16, №4(2), 2014. – с 450–453.

80. Тельтевская, В.А. Идентификация устройств интернета вещей с помощью технологий дополненной реальности / В.А. Тельтевская, В.В. Зеленов, Н.И. Шустов и др. // Информационные технологии и телекоммуникации. – 2017. – Т. 5, № 4. – С. 64–70.

81. Цифровая идентификация объектов: технология и не только; под ред. М.А. Медриша. – М.: Научное обозрение, 2016. – 228 с.

82. Цифровой идентификатор объекта [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Цифровой_идентификатор_объекта

83. Что такое DOI и почему это важно – Интернаука. [Электронный ресурс]. – Режим доступа: https://www.inter-nauka.com/poleznaya-informatsiya/doi/

84. Al-Bahri, M. Combating Counterfeit for IoT System based on DOA / M. Al-Bahri, A.A. Ateya, A. Muthanna, et al. // Proceedings of the 2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT) 2018, St. Petersburg, Russia, November 5-9, 2018. – IEEE, 2018. – P. 338–342.

85. Al-Bahri, M. Smart System Based on DOA and IoT for Products Monitoring and Anti-counterfeiting / M. Al-Bahri, A. Yankovsky, A. Borodin, R. Kirichek // 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC). – IEEE, 2019. – P. 25–31.

86. Al-Bahri, M. Testbed for Identify IoT Devices Based on Digital Object Architecture / M. Al-Bahri, A. Yankovsky, A. Borodin, R. Kirichek // Internet of Things, Smart Spaces, and Next Generation Networks and Systems. Proceedings of 18th International Conference, NEW2AN 2018, and 11th Conference, ruSMART 2018, St. Petersburg, Russia, August 27–29, 2018. – Cham: Springer, 2018. – P. 129–137.

87. Albreem M. A. M. et al. Green internet of things (IoT): An overview/ Albreem, M. A., El-Saleh, A. A., Isa, M., Salah, W., Jusoh, M., Azizan, M. M., & Ali, A. //2017 IEEE 4th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA). – IEEE, 2017. – C. 1–6.

88. Aluthge N. et al. IoT device fingerprinting with sequence-based features. – 2018.

89. Balanis, C.A. Antenna Theory: Analysis & Design / C.A. Balanis. – 2nd Edition. – New York: John Wiley & Sons, Inc., 1997. – 941 p.

90. Berners-Lee, T. RFC 3986. Uniform Resource Identifier (URL): Generic Syntax / T. Berners-Lee, R. Fielding, L. Masinter // https://www.ietf.org/rfc/rfc3986.txt .

91. Bude, Cristian, and Andreas Kervefors Bergstrand. "Internet of Things: Exploring and Securing a Future Concept." (2015).

92. Cooper D. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile/ S.Santesson, S.Farrell, S.Boeyem, R. Housley and W.Polk// RFC 5280 (2008), at https://tools.ietf.org/html/rfc5280

93. Da B. Identity/identifier-enabled networks (IDEAS) for Internet of Things (IoT) //2018 IEEE 4th World Forum on Internet of Things (WF-IoT). – IEEE, 2018. – C. 412–415.

94. digital object architecture for iot // http://www.wileyconnect.com/ home/2016/11/8/what-governmentsdecided- on-digital-object-architecture-foriot

95. Digital Object Protocol Specification, version 1.0, CNRI (November 12, 2009), at http://dorepository.org/documentation/Protocol_Specification.pdf.

96. Dudhe, P. V. Internet of Things (IOT): An overview and its applications/ N. V Kadam, R. M. Hushangabade , & Deshmukh, M. S. // 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS) (pp. 2650–2653). IEEE.

97. Duke M./A Roadmap for Transmission Control Protocol (TCP) Specification Documents // R.Braden, W.Eddy, E.Blanton, and A.Zimmermann/ RFC 7414 (2015), at http://tools.ietf.org/html/rfc7414

98. Evans, D. The Internet of Things How the Next Evolution of the Internet Is Changing Everything / D. Evans // CISCO White Papers, 2011.

99. F.Yergeau, “UTF-8, a transformation format of ISO 10646,” RFC 2279 (1998), at https://www.ietf.org/rfc/rfc2279.txt

100. Gneiting, Tilmann; Schlather, Martin. "Stochastic Models That Separate Fractal Dimension and the Hurst Effect". SIAM Review. 46: 269–282

101. Handle.Net Registry. URL: http://www.handle.net/index.html (дата обращения 22.01.2019)

102. hart API, документация [Электронный ресурс]. – Режим доступа: http://www.jfree.org/jfreechar

103. Hendriks, S. (2016). Internet of Things: how the world will be connected in 2025 (Master's thesis). https://tools.ietf.org/html/rfc7797

104. Ignatova, L. Analysis of the Internet of Things devices integration in 5G networks / A. Khakimov, A. Mahmood, & A .Muthanna. // 2017 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SINKHROINFO) (pp. 1–4). IEEE.

105. Inter Com, Мобильные телекоммуникации [Электронный ресурс]. – Режим доступа: http://www.mobilecomm.ru/wpcontent/uploads/pdf.magazine/2011/mtk_07-2011.pdf.

106. Internet Engineering Task Force, RFC 6455, WebSocket [Электронный ресурс]. – Режим доступа: https://tools.ietf.org/html/rfc6455/.

107. Jena, A.K. A. Modeling and Evaluation of Network Applications and Services / A.K. Jena, P. Pruthi, A. Popescu // Proceedings of the RVK 99 Conference. – Ronneby, Sweden. – June 1999

108. Kahn R.E. The Digital Library Project (Volume 1): The World of Knowbots/ R.E.Kahn and V.Cerf // [Draft],” CNRI (1988), at http://www.cnri.reston.va.us/kahn-cerf-88.pdf

109. Kahn R.E., "The organization of computer resources into a packet radio network," Managing Requirements Knowledge, International Workshop (1975), at https://www.computer.org/csdl/proceedings/afips/1975/5083/00/50830177.pdf

110. Kahn, R. A framework for distributed digital object services / R. Kahn, R. Wilensky // International Journal on Digital Libraries. – 2006. – Vol. 6. – Issue 2. – P. 115–123.

111. Kahn R. The Role of Architecture in Internet Defense : America's Cyber Future: Security and Prosperity in the Information Age, Center for a New American Security (CNAS)// Volume II, Chapter XII, May 2011

112. Karlen, D. Using projections and correlations to approximate probability distributions / D. Karlen // Comput.Phys. – 1998. – Vol.12, № 4. – p. 380–384

113. Kirichek, R. False clouds for Internet of Things and methods of protection / R. Kirichek, V. Kulik, A. Koucheryavy // 18th International Conference on Advanced Communication Technology (ICACT). – 2016. – P. 201–205.

114. Kirichek, R. Internet of things laboratory test bed / R. Kirichek, A. Koucheryavy // Lecture Notes in Electrical Engineering (LNEE). – 2016. – Vol. 348. – P. 485–494.

115. Kirichek, R. Model networks for Internet of Things and SDN / R. Kirichek, A. Vladyko, M. Zakharov, A. Koucheryavy // 18th International Conference on Advanced Communication Technology (ICACT), 2016. – IEEE, 2016. – P. 76–79.

116. Kirichek, R. Transfer of multimedia data via LoRa / R. Kirichek, V. D. Pham, A. Kolechkin, M. Al-Bahri, A. Paramonov // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). – 2017. – Vol. 10531. – P. 708–720.

117. Koo, J. Interoperability of device identification in heterogeneous IoT platforms / J. Koo, Y.G. Kim // 2017 13th International Computer Engineering Conference (ICENCO). – IEEE, 2017. – P. 26-29.

118. Lam K. Y., Chi C. H. Identity in the Internet-of-Things (IoT): New challenges and opportunities //International Conference on Information and Communications Security. – Springer, Cham, 2016. – С. 18–26.

119. Lund, D. Worldwide and Regional Internet of Things (IoT) 2014–2020. Forecast: A Virtuous Circle of Proven Value and Demand / D. Lund, C. MacGillivray, V. Turner, M. Morales // International Data Corporation (IDC), Tech. Rep., 2014.

120. M.Jones, J.Bradley and N.Sakimura, "JSON Web Signature (JWS)," RFC 7515 (2015), at https://tools.ietf.org/html/rfc7515; M.Jones, JWS,

121. Madakam S.Internet of Things (IoT): A literature review/, R. Ramaswamy, S. Tripathi //Journal of Computer and Communications. – 2015. – Т. 3. – №. 05. – С. 164.

122. Madhow, U. Fundamentals of digital communication / U. Madhow. – New York: Cambridge University Press, 2008. – 518 p.

123. Marker Tracking and HMD Calibrationfor a Video-based Augmented Reality Conferencing System / H. Kato, M. Billinghurst. // Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality. – 1999. – P.85–94

124. Muthanna, A. Delay Tolerant Network model based on D2D communication/ M. S. A.Muthanna, , K. Abdukodir, A. A Ateya, & M. Al-Bahri, //2019 4th MEC International Conference on Big Data and Smart City (ICBDSC) (pp. 1–5). IEEE.

125. N.Freed and N.Borenstein, "Multipurpose Internet Mail Extensions (MIME)," RFCs 2045 & 2046 (1996), at https://tools.ietf.org/html/rfc2045 and https://tools.ietf.org/html/rfc2046; K.Moore, MIME, RFC 2047 (1996), at https://www.ietf.org/rfc/rfc2047.txt; N.Freed and J.Klensin, MIME, "Media Type Specifications and Registration Procedures," IETF, RFCs

126. Overview of the Digital Object Architecture (DOA). [Электронный ресурс]. – Режим доступа: https://www.internetsociety.org/resources/doc/2016/overview-of-the-digital-object-architecture-doa/

127. PKI-Public Key Infrastructure, see https://www.ssh.com/pki

128. Proceedings of the World Telecommunication Standardization Assembly, 2016. – https://www.itu.int/pub/TREG- LIV.1-2016/en

129. See “System for uniquely and persistently identifying, managing and tracking digital objects,” U.S. Patent No. 6,135,646 (now expired).

130. Shortle J.F., Thompson J., Gross D., Harris C.M. Fundamentals of Queueing Theory. Hoboken: John Wiley & Sons., 2018. 576 c.

131. Skarmeta, A. A required security and privacy framework for smart objects / A. Skarmeta, J.L. Hernández-Ramos, J.B. Bernabe // ITU Kaleidoscope: Trust in the Information Society (K-2015). – IEEE, 2015. – P. 1–7.

132. T.Bray, Ed., “The JavaScript Object Notation Format,” RFC 8259 (Dec. 2017), at https://tools.ietf.org/html/rfc8259

133. T.Dierks and C.Allen, “The Transport Layer Security (TLS) Protocol,” RFC 2246 (1999), at https://www.ietf.org/rfc/rfc2246.txt; T.Dierks and E.Rescorla, TLS, RFCs 4346 & 5246 (2006 & 2008), at https://www.ietf.org/rfc/rfc4346.txt and https://tools.ietf.org/html/rfc5246

134. The DOI® System. [Электронный ресурс]. – Режим доступа: https://www.doi.org/

135. The Handle System // The DONA Foundation. URL: https://www.dona.net/handle-system (дата обращения 22.03.2019)

136. The Unicode Standard, http://www.unicode.org/standard/standard.html

137. Tinkermode, Internet of things platform, documentation [Электронный ресурс]. – Режим доступа: http://dev.tinkermode.com/tutorials/overview.html.

138. Unencoded Payload Option. RFC 7787 (2016), at 4288 & 4289 (2005), at https://tools.ietf.org/html/rfc4288 and https://tools.ietf.org/html/rfc4289

139. Wang, Ya. A privacy enhanced DNS scheme for the Internet of Things / Ya. Wang, Q. Wen // International Conference on Communication Technology and Application (ICCTA 2011). – 2011. – P. 699-702.

140. Weber R. H. Internet of Things–New security and privacy challenges //Computer law & security review. – 2010. – Т. 26. – №. 1. – С. 23–30.

141. What Governments Decided on Digital Object Architecture for IoT. [Электронный ресурс]. – Режим доступа: https://www.wileyconnect.com/home/2016/11/8/what-governments-decided-on-digital-object-architecture-for-iot

142. Тхаи Н.З. Удаленные вычисления через Web-сервер MATLAB как система массового обслуживания // Вестник Иркутского государственного технического университета. – 2012. – № 4 (63). – С. 25–32.

Приложение А. ИСХОДНЫЙ КОД ПРОЦЕССА ПРОВЕРКИ ОБЪЕКТА В АРХИТЕКТУРЕ ЦИФРОВЫХ ОБЪЕКТОВ

```
<script>
  var log = function(msg) {
    $('<span/>').html(msg + '\n').appendTo('#result')
  };

  log('Calling /rpc/Config.Get ...');
  $.ajax({
    url: '/rpc/Config.Get',
    success: function(data) {
      log('Result: ' + JSON.stringify(data));
      $('#ssid').val(data.wifi.sta.ssid);
      $('#pass').val(data.wifi.sta.pass);
    },
  });

  log('Calling /rpc/Config.Get ...');
  $.ajax({
    url: '/rpc/Config.Get',
    success: function(data) {
      log('Result: ' + JSON.stringify(data));
      $('#ssid').val(data.wifi.sta.ssid);
      $('#pass').val(data.wifi.sta.pass);
    },
  });

  $('#save').on('click', function() {
    log('Calling /rpc/Config.Set  ...');
    $.ajax({
      url: '/rpc/Config.Set',
      data: JSON.stringify({config: {wifi: {sta: {enable: true, ssid: $('#ssid').val(), pass:
$('#pass').val()}}}}),
      type: 'POST',
      // contentType: 'application/json',
      success: function(data) {
        log('Success. Saving and rebooting ...');
        $.ajax({url: '/rpc/Config.Save', type: 'POST', data: JSON.stringify({"reboot": true})});
      },
    })
  });

  $('#get').on('click', function() {
    $.ajax({
      url: '/rpc/Spi.Read',
      data: JSON.stringify({config: {wifi: {sta: {enable: true, ssid: $('#ssid').val(), pass:
$('#pass').val()}}}}),
      type: 'POST',
      success: function(data) {
        //document.getElementById('doi').innerText = data.num;
          var hex  = data.num.toString();
```

```
        var str = '';
        for (var n = 0; n < hex.length; n += 2) {
          str += String.fromCharCode(parseInt(hex.substr(n, 2), 16));
        }
        document.getElementById('doi').innerText = str;

       //document.getElementById('doi').innerText = data.num.toString(16);

     },
    })
  });

   $('#verify').on('click', function() {
   $.ajax({
     url: '/rpc/Spi.Verify',
     data: JSON.stringify({config: {wifi: {sta: {enable: true, ssid: $('#ssid').val(), pass:
$('#pass').val()}}}}),
     type: 'POST',
     success: function(data) {
       document.getElementById('status').innerText = data.tex;
       //document.getElementById('doi').innerText = data.num.toString(16);

     },
    })
  });

  $('#get_fake').on('click', function() {
    $.ajax({
     url: '/rpc/Spi.Readfake',
     data: JSON.stringify({config: {wifi: {sta: {enable: true, ssid: $('#ssid').val(), pass:
$('#pass').val()}}}}),
     type: 'POST',
     success: function(data) {
       //document.getElementById('doi').innerText = data.num;
         var hex  = data.num.toString();
         var str = '';
         for (var n = 0; n < hex.length; n += 2) {
           str += String.fromCharCode(parseInt(hex.substr(n, 2), 16));
         }
         document.getElementById('doi_fake').innerText = str;
         document.getElementById('reset').innerText = "reset";

       //document.getElementById('doi').innerText = data.num.toString(16);

     },
    })
  });

  $('#verify_fake').on('click', function() {
    $.ajax({
     url: '/rpc/Spi.Verifyfake',
```

```
    data: JSON.stringify({config: {wifi: {sta: {enable: true, ssid: $('#ssid').val(), pass:
$('#pass').val()}}}}),
    type: 'POST',
    success: function(data) {
      document.getElementById('status_fake').innerText = data.tex;
      document.getElementById('reset').innerText = "reset";
      //document.getElementById('doi').innerText = data.num.toString(16);

    },
  })
});

$('#reset').on('click', function() {
  $.ajax({
    url: '/rpc/Spi.Resetfake',
    data: JSON.stringify({config: {wifi: {sta: {enable: true, ssid: $('#ssid').val(), pass:
$('#pass').val()}}}}),
    type: 'POST',
    success: function(data) {
      document.getElementById('reset').innerText = data.tex;
      //document.getElementById('doi').innerText = data.num.toString(16);

    },
  })
});

$('#lhr_request').on('click', function() {
  var xhr = new XMLHttpRequest();
  xhr.open('GET', 'http://hdl.handle.net/api/handles/4263537/4000', false);
  xhr.send();
  if (xhr.status != 200)  {
    alert( xhr.status + ': ' + xhr.statusText);
  }
  else  {
    try {
      var data_list = JSON.parse(xhr.responseText);


    }
    catch (e) {
      alert("Error: ", e.message);
    }
    document.getElementById('suffix_list').innerText = xhr.responseText;

  }
});


function fillList(data_list) {

  phones.forEach(function(data_list) {
    var li = suffix_list.appendChild(document.createElement('li'));
```

```
    li.innerHTML = data_list.name;
  });

 }

</script>
```

ИСХОДНЫЙ КОД КЛАССА АРХИТЕКТУРЫ ЦИФРОВЫХ ОБЪЕКТОВ

```
package ru.sut.model.identificator;

public class Doi {

   private String identificator;
   private String preffix;
   private String suffix;
   private int consignmentId;
   private int manufactureId;
   private int productId;

   private Doi(){
   }

   public Doi(String identificator) throws Exception {
      this.identificator = identificator;
      init();
   }

   private void init() throws Exception {
      String[] i = identificator.split("/");

      if (i.length != 2) throw new Exception("Invalid identificator!");
      else {
         preffix = i[0];
         suffix = i[1];
      }

      String[] s = suffix.split("\\.");
      if (s.length != 3) throw new Exception("Invalid suffix!");
      else {
         productId = Integer.parseInt(s[0]);
         manufactureId = Integer.parseInt(s[1]);
         consignmentId = Integer.parseInt(s[2]);
      }
   }

   public String getPreffix() {
      return preffix;
   }

   public String getSuffix() {
```

```
        return suffix;
    }

    public int getConsignmentId() {
        return consignmentId;
    }

    public int getManufactureId() {
        return manufactureId;
    }

    public int getProductId() {
        return productId;
    }

    public String getIdentificator() {
        return identificator;
    }
}
```

2- Код класса Product.

```
package ru.sut.model.tables;

import com.fasterxml.jackson.annotation.JsonIgnore;

import javax.persistence.*;
import java.util.List;

@Entity
public class Product {

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int id;

    private String name;

    private String info;

    private byte[] photo;

    @JsonIgnore
    @OneToMany(mappedBy = "product")
    private List<Consignment> consignments;

    public Product() {
    }

    public Product(String name, String info, byte[] photo) {
        this.name = name;
        this.info = info;
        this.photo = photo;
```

```
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getInfo() {
        return info;
    }

    public void setInfo(String info) {
        this.info = info;
    }

    public byte[] getPhoto() {
        return photo;
    }

    public void setPhoto(byte[] photo) {
        this.photo = photo;
    }

    public List<Consignment> getConsignments() {
        return consignments;
    }

    public void setConsignments(List<Consignment> consignments) {
        this.consignments = consignments;
    }
}
```

КОД КЛАССА PRODUCT

```
package ru.sut.model.tables;

import com.fasterxml.jackson.annotation.JsonIgnore;

import javax.persistence.*;
import java.util.List;
```

```
@Entity
public class Product {

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int id;

    private String name;

    private String info;

    private byte[] photo;

    @JsonIgnore
    @OneToMany(mappedBy = "product")
    private List<Consignment> consignments;

    public Product() {
    }

    public Product(String name, String info, byte[] photo) {
        this.name = name;
        this.info = info;
        this.photo = photo;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getInfo() {
        return info;
    }

    public void setInfo(String info) {
        this.info = info;
    }

    public byte[] getPhoto() {
        return photo;
```

```
    }

    public void setPhoto(byte[] photo) {
        this.photo = photo;
    }

    public List<Consignment> getConsignments() {
        return consignments;
    }

    public void setConsignments(List<Consignment> consignments) {
        this.consignments = consignments;
    }
}
```

КОД КЛАССА LHSCOMPANY

```
package ru.sut.model;

import javax.persistence.*;
import javax.validation.constraints.NotNull;

@Entity
public class LhsCompany {

    @Id
    @GeneratedValue(strategy = GenerationType.AUTO)
    private int id;

    @NotNull
    private String name;

    @NotNull
    private String address;

    @ManyToOne
    @JoinColumn(name = "MPA_COMPANY_ID")
    private MpaCompany mpaCompany;

    public LhsCompany() {
    }

    public LhsCompany(String name, String address, MpaCompany mpaCompany) {
        this.name = name;
        this.address = address;
        this.mpaCompany = mpaCompany;
    }

    public int getId() {
        return id;
    }
```

```
    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public MpaCompany getMpaCompany() {
        return mpaCompany;
    }
    public void setMpaCompany(MpaCompany mpaCompany) {
        this.mpaCompany = mpaCompany;
    }
}
```

Приложение Б. АКТ ВНЕДРЕНИЯ РЕЗУЛЬТАТОВ ДИССЕРТАЦИОННОГО ИССЛЕДОВАНИЯ

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА» (СПбГУТ)

Юридический адрес: набережная реки Мойки, д. 61, Санкт-Петербург, 191186

Почтовый адрес: пр. Большевиков, д. 22, корп. 1, Санкт-Петербург, 193232
Тел.(812) 3263156, Факс: (812) 3263159
E-mail: rector@sut.ru
ИНН 7808004760 КПП 784001001
ОГРН 1027809197635 ОКТМО 40909000

18.02.2019 №_____
на №_____ от_____

УТВЕРЖДАЮ:
Проректор по научной работе

К.В. Дукельский

Акт

о внедрении научных результатов,

полученных Аль Бахри Махмудом Саидом Нассером в диссертационной работе

«РАЗРАБОТКА МОДЕЛЕЙ И МЕТОДОВ ИДЕНТИФИКАЦИИ УСТРОЙСТВ И ПРИЛОЖЕНИЙ ИНТЕРНЕТА ВЕЩЕЙ НА БАЗЕ АРХИТЕКТУРЫ ЦИФРОВЫХ ОБЪЕКТОВ».

Комиссия в составе декана факультета Инфокоммуникационных сетей и систем Л.Б.Бузюкова, доцента кафедры сетей связи и передачи данных М.А.Маколкиной и заведующей лабораторией кафедры сетей связи и передачи данных О.И. Ворожейкина составила настоящий акт в том, что научные результаты, полученные в диссертации «Разработка моделей и методов идентификации устройств и приложений интернета вещей на базе архитектуры цифровых объектов», использованы при чтении лекций, проведении практических занятий и лабораторных работ по курсам:

1. Интернет вещей (Рабочая программа № 02.12.15/788, утверждена Первым проректором-проректором по учебной работе Г.М. Машковым 21.09.2015), разделы Программы:

- Сети M2M. Классификация сетей M2M по видам трафика. Модели для опосредованного и псевдодетерминированного трафика. Пуассоновский, самоподобный и антиперсистентный трафик. Влияние трафика M2M на качество обслуживания традиционных услуг связи (речь, видео, данные). Способы уменьшения влияния трафика M2M.

- Ad Hoc или самоорганизующиеся сети. Приложения самоорганизующихся сетей. Всепроникающие сенсорные сети как технологическая основа внедрения концепции Интернета Вещей.

При этом используются следующие новые научные результаты, полученные Аль Бахри М.С.Н. в диссертационной работе:

- метод построения сетевой архитектуры цифровых объектов за счет введения промежуточного уровня взаимодействия
- модель повышения производительности архитектуры цифровых объектов;

2. Сети связи (Рабочая программа № 02.12.13/861, утверждена Первым проректором-проректором по учебной работе Г.М. Машковым 11.02.2016), разделы Программы:

- Управление информационными потоками в глобальных сетях, хранение информации, в т.ч. распределенное. Архитектура центров обработки данных. Распределенные облачные вычисления.

При этом используются следующие новые научные результаты, полученные Аль Бахри М.С.Н. в диссертационной работе:

- методика идентификации устройств и приложений интернета вещей в гетерогенных сетях связи на базе архитектуре цифровых объектов.

Кроме того, научные результаты, полученные Аль Бахри М.С.Н. были использованы при подготовке вкладов СПбГУТ в Сектор Стандартизации Телекоммуникаций Международного Союза Электросвязи:

- Y.IoT-DA-Counterfeit "Information management digital architecture to combat counterfeiting in IoT"
- Y.FW.IC.MDSC "Framework of identification and connectivity of moving devices in smart city"

Декан факультета ИКСС Л.Б. Бузюков

Доцент каф. ССиПД М.А. Маколкина

Зав. лабораторией кафедры ССиПД О.И. Ворожейкина