Федеральное государственное бюджетное учреждение науки
«Санкт-Петербургский Федеральный исследовательский центр
Российской академии наук»
(СПб ФИЦ РАН)

На правах рукописи

ЖЕРНОВА Ксения Николаевна

ОЦЕНИВАНИЕ ЗАЩИЩЁННОСТИ ЧЕЛОВЕКО-КОМПЬЮТЕРНЫХ ИНТЕРФЕЙСОВ, ОСНОВАННЫХ НА ТЕХНОЛОГИЯХ СЕНСОРНЫХ ЭКРАНОВ И ВИРТУАЛЬНОЙ РЕАЛЬНОСТИ

Специальность 2.3.6 — Методы и системы защиты информации,
информационная безопасность

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
кандидат технических наук, доцент
ЧЕЧУЛИН Андрей Алексеевич

Санкт-Петербург – 2022

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы исследования

В настоящее время растет важность информационной безопасности, и вместе с этим, растет количество обрабатываемых данных, сложность их обработки и сложность управления инцидентами.

При этом, многие операции (такие как аналитика, мониторинг, разведка и прочие) выполняются преимущественно вручную, несмотря на автоматизацию принятия решений, в том числе и в области информационной и компьютерной безопасности, а окончательное решение зачастую принимается оператором-экспертом. Таким образом, чтобы упростить восприятие информации, обрабатываемые данные чаще всего представляются в виде визуализации, с которой нужно взаимодействовать. По этой причине требуется создание человеко-компьютерных интерфейсов, которые позволили бы взаимодействовать с моделями визуализации более эффективно.

Человеко-компьютерные интерфейсы используются во всех областях человеческой деятельности, где применяются компьютерные системы: информационная безопасность, банковские приложения, образование, персональные компьютеры и т.д. Таким образом, применение человеко-компьютерных интерфейсов является повсеместным. При этом пользователи могут обмениваться с компьютерной системой чувствительными данными, содержащими конфиденциальную информацию. Однако в современных исследованиях мало внимания уделяется защите взаимодействия пользователя с компьютерной системой.

В настоящее время всё чаще применяются человеко-компьютерные интерфейсы, основанные на технологиях сенсорных экранов и виртуальной реальности. Многие персональные устройства пользователей (такие как смартфоны, планшеты или мониторы персональных компьютеров) имеют сенсорный экран. Также стремительно развивается технология виртуальной

реальности, которая начинает внедряться в различные области, такие как образование, медицина и военное дело. Следовательно, данные интерфейсы требуют проведения исследований с точки зрения информационной и компьютерной безопасности.

Современные исследования перспективных интерфейсов (сенсорные экраны и виртуальная реальность) сосредоточены на изучении конкретных типов уязвимостей и угроз. Небольшое количество обзоров приводит классификацию найденных уязвимостей, большая часть которых связана в большей степени с эргономикой, чем с информационной безопасностью. Однако для того, чтобы определить, насколько защищён данный человеко-компьютерный интерфейс, требуется создать методику оценивания уровня защищённости интерфейса.

Существующие системы оценки уязвимостей применимы к оценке уязвимостей сети и программного обеспечения, однако в них отсутствуют показатели, характерные для человеко-компьютерных интерфейсов, такие как канал восприятия и урон оператору. По этой причине требуется разработать методику создания подобных систем оценивания уязвимостей, пригодных для оценивания защищённости человеко-компьютерного интерфейса.

Степень разработанности темы

Исследования вопросов формирования защищённых интерфейсов, основанных на сенсорных экранах и виртуальной реальности, существуют, однако их крайне мало. При этом исследования в области безопасности интерфейсов подразделяются на две группы: решения конкретных вопросов безопасности с помощью человеко-компьютерных интерфейсов и поиски методов защиты от конкретных угроз безопасности для человеко-компьютерных интерфейсов. Такие российские учёные как Юсупов Р.М., Ронжин А.Л., Карпов А.А., В. Л. Авербух, Байдалин А.Ю. занимались проблемами человеко-компьютерного взаимодействия. Ряд зарубежных учёных (например, Roesner F., Gulhane A., George C., Khamis M.) решал

проблемы защищённости интерфейсов от конкретных уязвимостей. Также со стороны иностранных учёных были попытки классифицировать угрозы для человеко-компьютерных интерфейсов (исследования таких авторов, как Kohno T., Thalmann D., Azuma R., Behringer R.). Однако не было выявлено работ, посвящённых оцениванию уязвимостей человеко-компьютерных интерфейсов.

Таким образом, несмотря на сделанный учёными научный задел, проблема оценивания уровня защищённости интерфейсов в области информационной безопасности на данный момент не разрешена, поэтому требуется проведение новых исследований.

Цель диссертационной работы: повышение защищённости человеко-компьютерных интерфейсов.

Решаемая научная задача: разработка комплекса моделей, алгоритмов и методики оценивания человеко-компьютерных интерфейсов, повышающих их защищённость. Решение сформулированной научной задачи предусматривало:

1) разработку модели человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности и модели уязвимостей человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности.

2) разработку алгоритма оценивания общего уровня защищённости интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности.

3) разработку методики оценивания защищённости интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности.

4) разработку архитектуры системы оценивания человеко-компьютерного интерфейса и её программного прототипа, реализующего разработанные модели и алгоритмы с помощью разработанной методики оценивания защищённости интерфейсов.

Объект исследования: перспективные человеко-компьютерные интерфейсы и присущие им уязвимости.

Предмет исследования: модели человеко-компьютерных интерфейсов и их уязвимостей, а также алгоритмы и методика, используемые для оценивания уровня защищённости человеко-компьютерных интерфейсов.

Научная новизна результатов определяется тем, что:

1) предложена новая аналитическая модель уязвимостей человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, отличающаяся от известных моделей расширенным множеством учитываемых уязвимостей этих человеко-компьютерных интерфейсов и связанных с ними новых параметров (урон оператору, канал восприятия и взаимодействие), обеспечивающая возможность работы с данными, необходимыми для оценивания защищённости интерфейса, и позволяющая учесть специфику технологий сенсорных экранов и виртуальной реальности;

2) разработан оригинальный алгоритм оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, по комплексному показателю, отличающийся от аналогов новыми правилами расчёта оценки уязвимости, учитывающий характеристики участников обмена информацией в человеко-компьютерных интерфейсах, обеспечивающий повышение показателей защищённости по сравнению с предложенными ранее алгоритмами;

3) предложена методика оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, отличающаяся от аналогов комплексным применением предложенных моделей и алгоритмов как на этапе разработки, так и на этапе эксплуатации интерфейсов, обеспечивающая повышение показателей защищённости интерфейсов по сравнению с аналогами;

4) разработаны архитектура и программная реализация системы оценивания защищённости человеко-компьютерных интерфейсов,

основанных на технологиях сенсорных экранов и виртуальной реальности, отличающиеся от аналогов расширенной функциональностью по расчёту оценок уязвимостей и уровня защищённости интерфейса, обеспечивающие оператору или разработчику выбор интерфейса с минимальной уязвимостью и повышение защищённости системы в целом.

Теоретическая и практическая значимость работы обусловлена доведением полученных результатов до уровня программной реализации и возможностью их внедрения в научные проекты, НИР и ОКР, связанные с разработкой новых и совершенствованием существующих систем. Предполагается, что использование полученных в данном диссертационном исследовании результатов позволит повысить общую защищенность как разрабатываемых, так и эксплуатируемых систем, использующих виртуальную реальность и/или сенсорные экраны в качестве интерфейса взаимодействия с пользователем системы. Кроме того, полученные результаты могут быть полезны исследователям в области информационной безопасности и человеко-компьютерного взаимодействия.

Теоретическая значимость диссертации состоит в развитии теории взаимодействия оператора и компьютерных систем, использующих технологии сенсорных экранов и виртуальной реальности, и разработке алгоритмов оценивания уязвимостей и оценивания общего уровня защищённости человеко-компьютерных интерфейсов, а также методик, использующих алгоритмы оценивания человеко-компьютерных интерфейсов. В качестве практической значимости выступает тот факт, что разработанные модели, алгоритмы, программы и методики могут быть использованы также при разработке систем оценивания защищённости человеко-компьютерных интерфейсов.

Полученные результаты ориентируются на решение задач информационной безопасности и будут носить фундаментальный и исследовательский характер. Кроме того, исследование учитывает

когнитивные особенности восприятия оператора при работе с интерфейсом, что обуславливает междисциплинарность исследования.

Были получены следующие результаты: (1) модели человеко-компьютерных интерфейсов, в том числе подходящих для описания интерфейсов на основе технологий сенсорных экранов и виртуальной реальности; (2) модели уязвимостей человеко-компьютерных интерфейсов; (3) алгоритмы оценивания уязвимостей человеко-компьютерных интерфейсов, а также уровня их защищённости; (4) экспериментальный стенд, реализующий модели интерфейсов для управления данными безопасности с использованием технологии сенсорных экранов; (5) алгоритм оценивания защищённости интерфейсов взаимодействия.

По результатам выполнения диссертационного исследования опубликовано девять статей: пять статей в журналах из перечня ВАК, четыре статьи в трудах конференций, индексируемых в системах цитирования Web of Science и Scopus. Кроме того, полученные результаты представлены в 17 тезисах на международных и всероссийских научно-технических и научно-практических конференциях, в том числе “Parallel, Distributed, and Network-Based Processing” (PDP 2020), «Завалишинские чтения» (2020), “Intelligent Information Technologies for Industry” (IITI 2021), «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (АПИНО 2018, 2019, 2020, 2021).

Методология и методы исследования. В рамках диссертационного исследования разрабатывались модели интерфейсов (включающих в себя модели управления и модели визуализации для решения конкретных задач управления данными безопасности), алгоритмы взаимодействия оператора с системами информационной безопасности, алгоритмы оценивания уязвимости интерфейсов взаимодействия и уровня защищённости интерфейсов. Ниже перечислены используемые подходы и методы, которые применялись для достижения цели диссертационного исследования и реализации поставленных задач.

1. Для разработки моделей интерфейсов использовались следующие методы и подходы:

- методы и подходы к визуализации больших объемов многомерных данных;

- методы визуальной обработки информации посредством когнитивной графики;

- методы кластеризации и классификации данных;

- методы теории графов, теории принятия решений, вычислительной геометрии, и вычислительной топологии.

2. Для разработки алгоритмов взаимодействия оператора с системами информационной безопасности использовались следующие методы и подходы:

- методы и подходы к анализу гетерогенных данных;

- методы и подходы аналитического и имитационного моделирования;

- методы и подходы контроля и управления доступом;

- методы и подходы обнаружения и предотвращения утечек информации;

- методы и подходы выявления и противодействия информации в интернет пространстве и социальных сетях;

- методы и подходы компьютерной криминалистики;

- методы и подходы, использующиеся в работе операционных центров безопасности;- методы и подходы анализа рисков и выработки контрмер;

- методы и подходы анализа вредоносного программного обеспечения;

- методы и подходы обнаружения и предотвращения вторжений;

- методы и подходы разграничения потоков данных компьютерных сетей;

- методы и подходы мониторинга трафика и состояния компьютерных сетей и сетей интернета вещей;

- методы и подходы управления информацией и событиями информационной безопасности.

3. Для разработки алгоритма оценивания уязвимости интерфейсов взаимодействия, а также уровня их защищённости, использовались следующие методы и подходы:

- методы формальной оценки эффективности;
- методы и подходы экспертного анализа действий пользователя;
- методы психологической оценки эффективности визуального восприятия человека;
- методы экспериментальной оценки результатов на основе двойного рандомизированного тестирования.

4. Для разработки методики оценивания защищённости человеко-компьютерных интерфейсов использовались следующие методы и подходы:

- методы системного анализа;
- методы управления контекстом для отслеживания текущей ситуации, ее моделирования и выстраивания механизмов поведения системы в зависимости от этой модели;
- подход к онтологическому моделированию для описания знаний предметной области.

Положения, выносимые на защиту. В результате решения сформулированной задачи получены следующие научные результаты, выносимые на защиту:

1) модель уязвимостей человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности;

2) алгоритм оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, по комплексному показателю;

3) методика оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности;

4) архитектура и программная реализация системы оценивания уровня защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности.

Степень достоверности научных результатов, представленных в настоящем диссертационном исследовании, подтверждается с помощью подробного анализа современных работ и исследований в рассматриваемой области. Также обоснованность подтверждена согласованностью полученных результатов экспериментов, результаты и основные положения успешно прошли апробацию на различных научных и научно-практических конференциях всероссийского и международного уровня. Кроме того, результаты подтверждаются рядом публикаций, описывающих результаты экспериментов и раскрывающих основные положения исследования.

Апробация результатов работы. Результаты научной работы были подтверждены на следующих научно-практических конференциях:

1. 4th International Symposium on Mobile Internet Security (MobiSec 2019), (2019).
2. 28th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP-2020), (2020).
3. XV Международная конференция по электронике и робототехнике "Завалишинские чтения", (2020).
4. 5th International Scientific Conference “Intelligent Information Technologies for Industry”, Sochi, Russia, (2021).
5. Санкт-Петербургская международная конференция «Региональная информатика», г. Санкт-Петербург, СПб ФИЦ РАН, (2020).
6. Санкт-Петербургская межрегиональная конференция «Информационная безопасность регионов России», участие с докладом, (2019, 2021).
7. Международная научно-техническая и научно-методическая конференция "Актуальные проблемы инфотелекоммуникаций в науке и

образовании" (АПИНО), СПбГУТ им. проф. М.А. Бонч-Бруевича, (2018, 2019, 2020, 2021).

8. Межрегиональная научно-практическая конференция «Перспективные направления развития отечественных информационных технологий», (2019, 2020).

Результаты, полученные в ходе исследования, были использованы в следующих проектах:

1. Разработка методов поиска уязвимостей интерфейсов взаимодействия человека с искусственным интеллектом транспортной среды “умного города”, № 19-29-06099 мк, руководитель Чечулин А.А., годы проведения 2019-2022 гг.

2. Модели, алгоритмы и методики человеко-компьютерного взаимодействия в области информационной безопасности, № 20-37-90130 Аспиранты, руководитель Чечулин А.А., годы проведения 2020-2022 гг.

3. Модели, методы, методики и алгоритмы человеко-машинного взаимодействия для поддержки визуальной аналитики сетевой безопасности критических инфраструктур с использованием сенсорных мультитач-экранов, № 18-07-01488 А, руководитель Котенко И.В., годы проведения 2018-2020 гг.

Публикации. Согласно результатам, полученным в диссертационном исследовании, было опубликовано 9 статей, 5 из них были изданы в рецензируемых журналах, входящих в перечень ВАК, 4 были изданы в сборниках трудов конференций международного уровня, индексируемых в базах WoS и/или SCOPUS. Кроме того, были опубликованы 17 статей в изданиях, входящих в РИНЦ. Также было зарегистрировано 9 программ для ЭВМ.

Личный вклад. Все полученные результаты, которые были представлены в настоящей диссертационной работе, получены лично автором в процессе выполнения научно-исследовательской деятельности.

Структура и объем диссертационной работы. Диссертационное исследование включает в себя введение и заключение, три главы основного

текста, а также список использованных источников (147 наименований) и 2 приложения. Объем работы – 161 страница машинописного текста; включая 37 рисунков и 16 таблиц.

Краткое содержание работы следующее. **В первой главе** диссертации проводится подробный анализ проблем современных типов человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности. Основными проблемами данных типов интерфейсов являются проблемы (1) визуализации, (2) управления, (3) защищённости. Выявлено следующее противоречие: несмотря на необходимость формирования защищённых человеко-компьютерных интерфейсов, в настоящее время не существует единой системы оценивания уровня защищённости интерфейса. Обоснована актуальность цели проведения научной работы. Предложено использование методики оценивания человеко-компьютерного интерфейса, сочетающей в себе оценивание уровня защищённости интерфейса и уровня удобства использования. Проведённый анализ позволил поставить задачу исследования и определить критерии успешности её выполнения.

Во второй главе диссертации описывается комплекс разработанных моделей интерфейса и уязвимостей интерфейса. Модель интерфейса включает в себя три элемента: (1) описание программно-аппаратной настройки элементов интерфейса; (2) описание взаимосвязей между элементами интерфейса; (3) описание системы защиты, которая может присутствовать на различных элементах интерфейса. Модель уязвимостей определяет, насколько уязвим тот или иной элемент интерфейса, а также возможность злоумышленника атаковать данный элемент интерфейса. Также в данной главе даны описания разработанных алгоритмов оценивания уязвимостей интерфейса, а также оценивания общего уровня защищённости интерфейса. В качестве особенности описываемых алгоритмов выступает их направленность на повышение осведомленности оператора или разработчика об уровне защищённости используемого или разрабатываемого интерфейса.

Таким образом, данные алгоритмы возможно использовать как на этапе проектирования, так и на этапе эксплуатации.

В третьей главе диссертации приведено описание методики оценивания человеко-компьютерного интерфейса, которая включает в себя оценивание как защищённости, так и удобства использования. В главе выделены три этапа выполнения методики: (1) сбор информации о человеко-компьютерном интерфейсе; (2) анализ защищённости интерфейса; (3) анализ удобства использования интерфейса. Первые два этапа относятся к процессу оценивания уровня защищённости человеко-компьютерного интерфейса и основаны на разработанных алгоритмах оценивания уязвимостей и защищённости интерфейса, третий этап описывает процесс оценивания удобства использования интерфейса. Также данная глава описывает архитектуру программного прототипа, реализующего предлагаемую методику. Проведена экспериментальная оценка выполнения требований к оперативности и ресурсопотреблению разработанного прототипа. Результаты проделанных экспериментов с участием прототипа доказали, что разработанная методика позволяет выполнить задачу исследования и удовлетворяет предъявляемым к ней требованиям. Также в данную главу включены предложения по внедрению и использованию методики.

ГЛАВА 1 СИСТЕМНЫЙ АНАЛИЗ ЗАДАЧИ ОЦЕНИВАНИЯ ЗАЩИЩЕННОСТИ ЧЕЛОВЕКО-КОМПЬЮТЕРНОГО ИНТЕРФЕЙСА

1.1 Человеко-компьютерные интерфейсы

Пользователи передают огромное количество информации по сети. Многие передаваемые данные могут быть конфиденциальными, например, пароли, коды доступа и прочие данные, которые нужно защитить от третьих лиц. Существуют различные методы и алгоритмы защищённой передачи этих данных. Однако процесс передачи этих данных от человека к устройству и, наоборот, от устройства к человеку, защищён слабо. Эта проблема особенно актуальна, когда речь идёт о приложениях информационной и компьютерной безопасности, где может возникнуть необходимость управлять конфиденциальными данными. По этой причине работа оператора с интерфейсом приложения информационной безопасности должна быть защищена.

Оператор взаимодействует с компьютерной системой с помощью человеко-компьютерных интерфейсов. В зависимости от области человеческой деятельности, интерфейсу могут быть даны различные определения. Даже в одной области компьютерных наук определения интерфейса могут различаться. Большая Российская Энциклопедия разделяет интерфейсы на «интерфейсы пользователя», «интерфейсы программ», «интерфейсы программ с аппаратными средствами», «интерфейсы аппаратных средств». В данной работе исследуются интерфейсы пользователя, которые представляют собой «способ и средства взаимодействия пользователя с программами» [1]. Этот интерфейс позволяет оператору взаимодействовать с приложениями. При этом данный тип интерфейса имеет аппаратную и программную часть. Аппаратная часть включает в себя устройства, с помощью которых осуществляется передача данных от человека к машине и от машины к человеку. Программная часть –

графический вывод обработанных данных и графические элементы управления этими данными.

В данной работе под интерфейсом понимается человеко-компьютерный интерфейс, который является набором средств, обеспечивающих взаимодействие оператора и компьютерной системы [1]. Таким образом, интерфейс отображает данные приложения и даёт оператору возможность управлять этими данными, вносить изменения и новые данные. Одним из основных инструментов информационной безопасности является визуальная аналитика. Задачи визуальной аналитики: (1) презентация, (2) мониторинг, (3) расследование и (4) управление [2]. По этой причине интерфейс приложений информационной безопасности содержит в себе две большие составляющие: (1) набор моделей визуализации, отображающих данные (презентация), в целях отслеживания их оператором (мониторинг), и (2) моделей управления, которые позволяют этими данными управлять (расследование и управление). Определения этих двух составляющих.

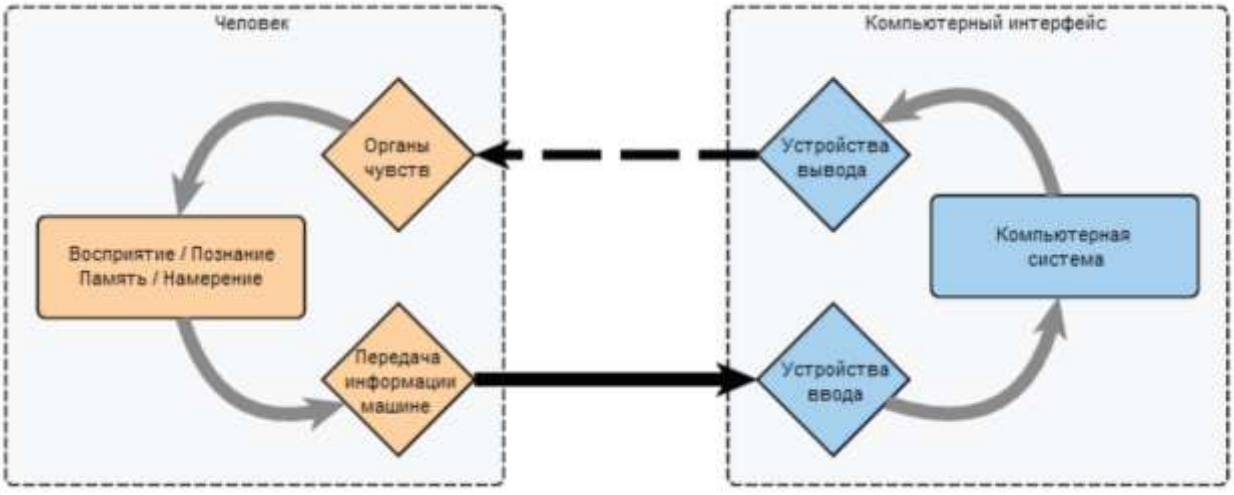


Рисунок 1 – Схема взаимодействия человека и компьютера [3]

1. Модель визуализации – базовая модель, описывающая структуру и свойства передачи данных от человека к машине. Используется как часть модели интерфейса и включает в себя совокупность графических примитивов (окружности, линии, точки, цвета, анимации и т.д.), которая формирует изображение и является графическим отображением данных. Например,

модель столбчатого графика является совокупностью прямоугольников разной длины и является отображением двумерной таблицы численных данных.

Современные интерфейсы систем безопасности имеют ряд общих черт. Как правило, классические интерфейсы включают в себя окна для ввода и для демонстрации каких-либо данных, панели с кнопками функций, также часто в интерфейсах систем безопасности присутствует визуализация.

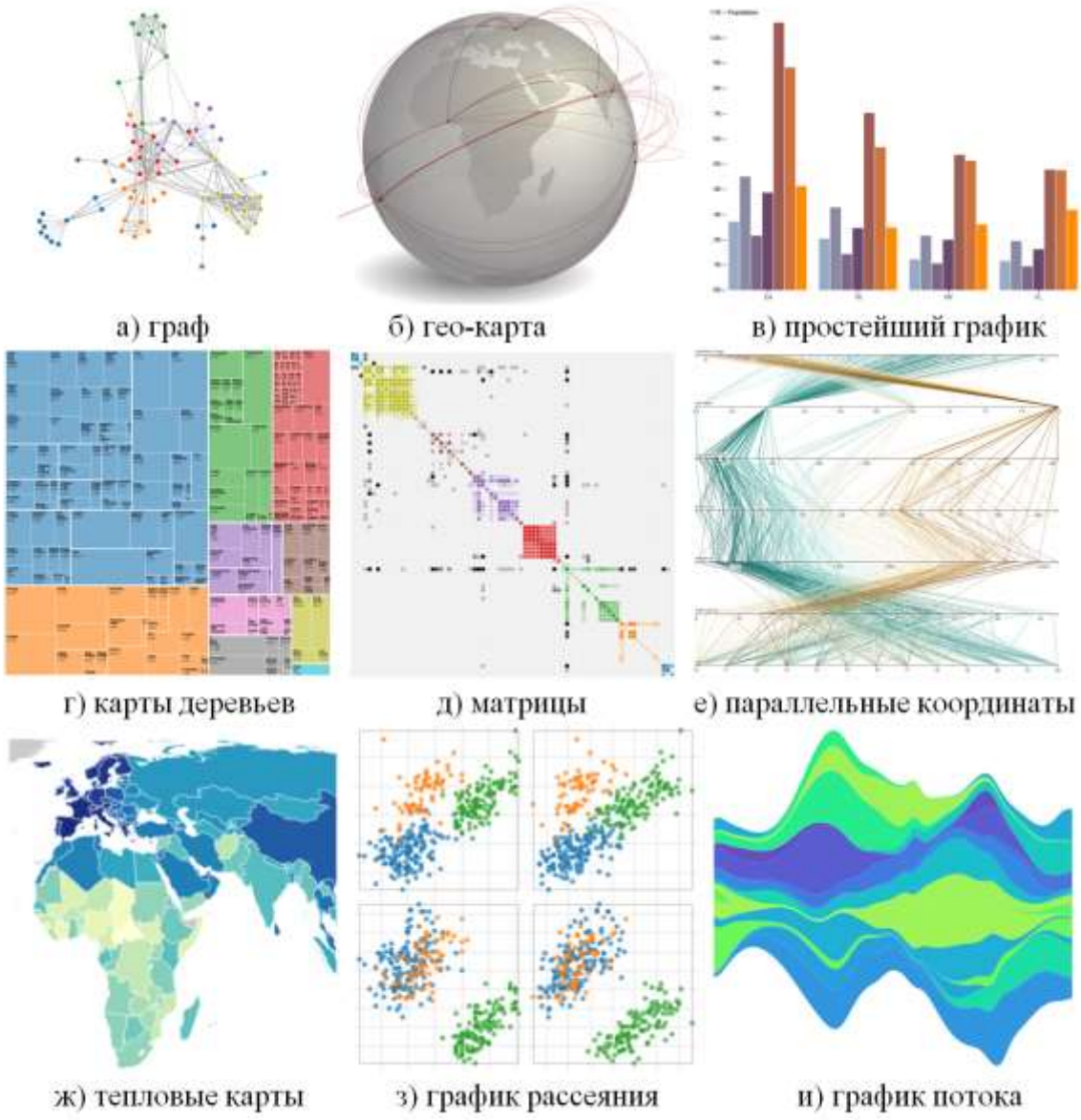


Рисунок 2 – Модели визуализации

Модели визуализации и их реализация могут быть разными в различных приложениях, так как рассчитаны на разные задачи безопасности.

Например, графы позволяют визуализировать компьютерную сеть [4], с их помощью можно отобразить результат сканирования портов [5], отобразить атаки и проследить их маршруты [6, 7, 8], также можно провести моделирование сценариев атак [9]. Однако взаимодействие с моделями визуализации, в основном сводится к взаимодействию через нажатие на кнопки и прочие стандартные инструменты управления. Кроме того, многие интерфейсы обходятся вообще без взаимодействия с визуализацией, и тогда визуализация имеет исключительно демонстрационную роль.

2. Модель управления – базовая модель, описывающая структуру и свойства передачи данных от человека к машине. Используется как часть модели интерфейса и включает в себя конкретные действия, совершаемые оператором для отправки команд. Например, нажатие кнопку, жест в воздухе, голосовая команда и т.д.

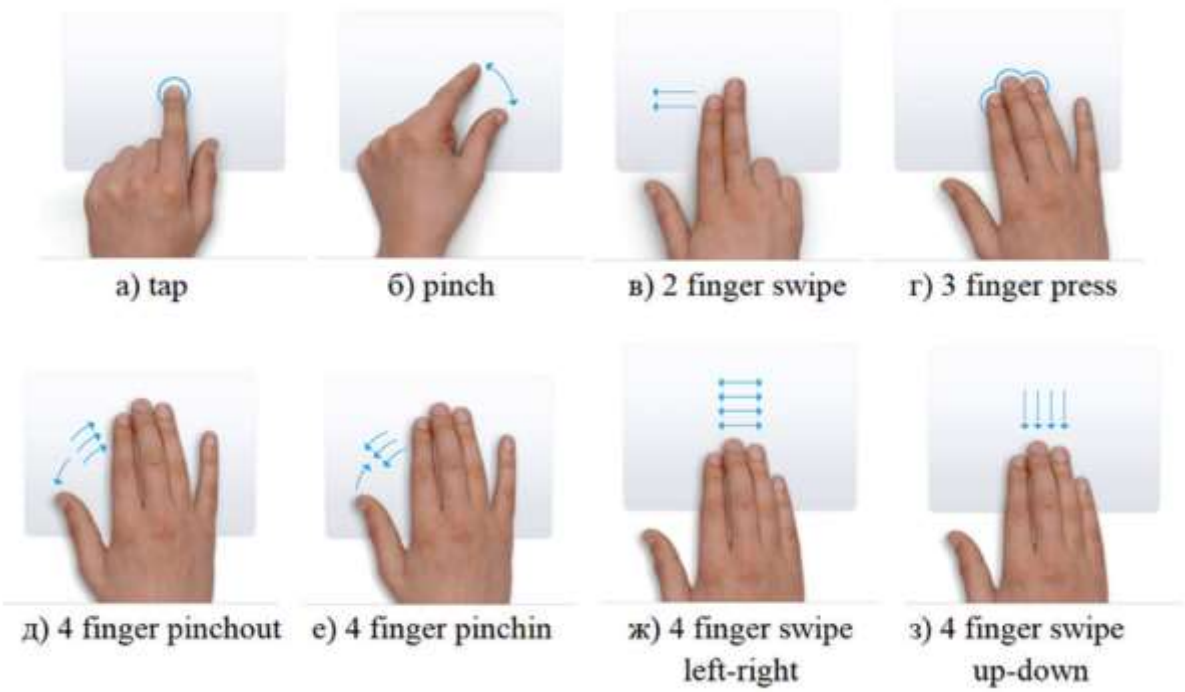


Рисунок 3 – Модели управления на примере тачпада [10]

Необходимо также отметить то, что количество обрабатываемых данных всё время возрастает, и эти данные становятся всё более разнообразными и сложными, что приводит к усложнению моделей визуализации. Такими моделями крайне сложно управлять, осуществлять

навигацию в данных и производить их анализ. Поэтому такой подход к визуализации может вести либо к недостатку информации, которую отображает модель, либо к перегруженности визуализации информацией. Таким образом, требуются более совершенные модели взаимодействия и альтернативные методы управления. Существуют инструменты, которые позволяют взаимодействовать с визуализацией так, чтобы не было необходимости её перегружать: например, «рыбий глаз», «семантическое масштабирование», «множественный взгляд» и т.п. [11]. Дополнительно повысить удобство использования интерфейсов может позволить реализация взаимодействия с системой как с реальным объектом (например, с помощью «листания» при переходе от одной визуализации к другой) [12].

Однако, подобные удобные инструменты для управления визуализацией часто сложно реализовать и использовать вместе с «традиционным» интерфейсом, управляемым посредством клавиатуры и мыши. С помощью более новых интерфейсов, например, основанных на сенсорных экранах и виртуальной/дополненной реальности, можно использовать дополнительные инструменты с большей лёгкостью, за счёт наличия у многих сенсорных устройств функции Multi-touch [13], [14], а в случае виртуальной/дополненной реальности – за счёт возможности распознавать жесты рук в воздухе [15]. Так как данные интерфейсы в настоящее время являются активно развивающейся областью, остановимся на сенсорных интерфейсах и виртуальной/дополненной реальности подробнее.

Современные типы интерфейсов.

Интерфейсы, основанные на технологии сенсорных экранов, предполагают взаимодействие пользователя с системой посредством прикосновения к экрану устройства. Таким образом, элементы управления соединены с элементами вывода визуализации информации. Управление может быть реализовано как с помощью простого касания, так и с помощью различных жестов на сенсорном экране. Жесты – это различные движения

одним или несколькими пальцами по сенсорному экрану, к которым привязаны какие-либо функции, выполняемые приложением в ответ на соответствующий жест. В настоящее время технологию сенсорных экранов поддерживают многие устройства: смартфоны, планшеты, сенсорные мониторы для компьютеров и ноутбуков.

Модели визуализации на дисплее сенсорного устройства не отличаются от моделей визуализации, используемых для «традиционных» кнопочных графических интерфейсов. Чаще всего используется двухмерная визуализация данных, так как применение третьего измерения обычно не несёт полезной информации и, вместе с тем, перегружает визуализацию, затрудняя её восприятие для пользователя. В случае использования трёхмерной визуализации, у пользователя не возникает ощущения присутствия, характерного для интерфейсов виртуальной реальности.

Ключевая особенность визуализации в виртуальной реальности состоит в том, что пользователь получает ощущение присутствия. Во время взаимодействия с объектами виртуальной реальности у пользователя создаётся впечатление, будто находящийся перед ним объект реален. Взаимодействие с объектами виртуальной реальности приближено к взаимодействию с физическими объектами: пользователь может схватить объект с помощью контроллера, перемещать виртуальные объекты вокруг себя, поворачивать их, отбросить в сторону и т.д. При этом пользователь видит себя находящимся в одном виртуальном пространстве с теми объектами, с которыми взаимодействует. Управление объектами виртуальной реальности может осуществляться с помощью различных контроллеров или с помощью бесконтактных жестов в случае наличия камеры, распознающей такие жесты.

Модели визуализации для виртуальной реальности чаще всего обладают тремя измерениями, как и объекты физической реальности. Интерфейс на основе сенсорных экранов не обладает такими же возможностями взаимодействия, что и виртуальная реальность, по этой

причине использование объёма для передачи количественных данных было бы избыточным. Однако в случае виртуальной реальности, ввиду возможности осмотреть объект со всех сторон, приблизив его к себе, или, наоборот, самостоятельно приблизившись, объём может применяться для передачи количественных данных.

Область данной работы в части интерфейсов – это современные виды интерфейсов, основанные на *сенсорных экранах* и *виртуальной реальности*. Сенсорные экраны в настоящее время очень распространены (планшеты, смартфоны), пользователи чаще всего устанавливают на смартфон множество приложений, содержащих чувствительные данные, которыми могли бы воспользоваться третьи лица. Это банковские приложения, ГосУслуги и т.п. Также для устройств с сенсорными экранами разрабатываются мобильные версии приложений безопасности. Для виртуальной реальности также начинают разрабатывать приложения информационной безопасности. Таким образом, современные типы интерфейсов (сенсорные экраны и виртуальная реальность) являются перспективным направлением для дальнейших исследований.

1.2 Место и роль человеко-компьютерных интерфейсов в области информационной безопасности

Так как современные типы интерфейсов являются новой развивающейся областью, они обладают рядом нерешённых проблем, касающихся визуализации данных, управления визуализации и защищённости интерфейса.

1.2.1 Общие проблемы визуальных человеко-компьютерных интерфейсов

Поскольку интерфейс включает в себя два основных компонента – визуализация данных и управление этими данными, проводятся многочисленные исследования, касающиеся этих направлений. Например, в настоящее время проводится множество исследований визуальной аналитики в области компьютерной безопасности. При этом выявленные исследования

проблем визуализации делятся на два типа: изучение существующих подходов к визуализации и разработка новых подходов. Однако ещё одной актуальной проблемой остаётся проблема защищённости приёма и передачи этих данных. С точки зрения человеко-компьютерного интерфейса, приём и передача данных происходит при взаимодействии оператора с интерфейсом. Таким образом, исследования в области человеко-компьютерного взаимодействия с приложениями информационной и компьютерной безопасности делятся на четыре основные группы:

(1) исследования новых подходов к визуализации данных, использующиеся для разработки новых способов визуального отображения данных безопасности (моделей визуализации);

(2) исследования существующих подходов к визуализации данных, которые можно применить для решения различных задач безопасности;

(3) исследования в области управления данными безопасности;

(4) исследования возможных способов защиты взаимодействия оператора с данными визуализации.

Для разработки успешных моделей и методов взаимодействия оператора с интерфейсом приложений информационной безопасности необходимо учитывать все эти направления исследований.

Проблемы визуализации. Исследования проблем визуализации позволяют ответить на следующие вопросы: (а) какими структурами данных приходится оперировать, (б) какие модели визуализации являются применимыми к задачам компьютерной безопасности.

(а) В области безопасности компьютерных сетей данные можно разделить на *количественные* и *категориальные*. *Количественные данные* – это такие данные, которые могут быть выражены числом, например, количество передаваемых пакетов в сети, количество запущенных процессов на одной машине, объём трафика, передаваемого от одного хоста другому, и т.п. *Категориальные данные* нельзя выразить количественно, они отражают принадлежность объекта к какой-либо группе, например, типы передаваемых

пакетов, типы устройств в сети, набор политик локальной сети, типы уязвимостей и т.п.

(б) Выбираемая модель визуализации зависит от типа данных, которыми будет оперировать специалист в области компьютерной безопасности. По этой причине модели визуализации также имеют два вида: такие модели, которые могут отображать количественные данные, и такие модели, которые отображают категориальные данные. Модели, отображающие количественные структуры: простейшие графики (гистограммы, круговые диаграммы), графики рассеяния, графики потока. Модели, отображающие категориальные структуры: графы, деревья и карты деревьев, параллельные координаты, матрицы [11]. Характерным примером использования визуализации в сетевой безопасности являются графы атак [16], которые позволяют проследить маршруты атакующего субъекта. С помощью деревьев и карт деревьев можно реализовать визуализацию древовидной структуры, например, визуализировать файловую систему. Матрицы позволяют проанализировать уязвимость портов, при этом, если объединить их с графами атак, то с их помощью можно проследить возможные маршруты атакующего с учётом известных уязвимостей [17]. Кроме того, существующие модели визуализации можно комбинировать [18], например, появляются модели, совмещённые с картами Вороного [19, 20], также различные модели часто совмещаются с гео-картами и хордовыми диаграммами [21].

Проблемы управления. Исследования, касающиеся компонента управления человеко-компьютерного интерфейса, позволяют оценить, (в) какие подходы к человеко-компьютерному взаимодействию в области компьютерной безопасности будут наиболее адекватными.

(в) Для проактивного мониторинга состояния систем сетевой безопасности требуется постоянно взаимодействовать с отображаемыми информационными объектами. В последнее время получило развитие множество аппаратных интерфейсов (сенсорные экраны, нейроинтерфейсы,

очки виртуальной реальности), а также множество методов, которые позволяют управлять визуализацией, используя тот или иной интерфейс. Простейшие примеры таких методов – перетягивание объектов (drag-and-drop) или увеличение объектов «рыбий глаз» [11] в случае сенсорных экранов. Технология отслеживания взгляда возможна для виртуальной реальности. Нейроинтерфейс управляется импульсами мозга, которые улавливаются электродами.

Несмотря на большое количество открывшихся возможностей в управлении данными визуализации, в большинстве случаев управление происходит при помощи традиционных элементов графического интерфейса: кнопки, выпадающие списки, радиальные кнопки и т.д. [22, 23, 24]. По этой причине управление данными отделено от непосредственно визуализации. Оператор по-прежнему взаимодействует с данными, а не с отображаемыми объектами, содержащими в себе эти данные. Таким образом, данные и модель визуализации всё ещё существуют отдельно друг от друга. Приложения, реализованные на основе сенсорных экранов, в основном, повторяют традиционные методы взаимодействия, где прикосновение к экрану заменяет нажатие на кнопку указателем мыши. Такой способ управления сложными моделями визуализации является проблематичным.

Закономерно, что чем сложнее модель визуализации, тем сложнее ей управлять. Таким образом, преимущество отображения большого количества разнородных данных превращается в недостаток для управления этими данными. При этом данные компьютерной безопасности чаще всего являются разнородными и обладают большими объёмами. По этой причине требуются такие методы человеко-компьютерного взаимодействия, которые смогут обеспечить управление достаточно сложной моделью визуализации, однако без необходимости запоминать большое количество команд управления, при этом команды должны соответствовать принципу прямого взаимодействия и реализовываться с помощью жестов на сенсорных экранах.

Взаимодействие с приложением компьютерной безопасности должно включать в себя методы, позволяющие реализовать шаблон информационного поиска [25], заключающийся в трёх пунктах:

1) визуализация должна содержать общую картину ситуации;

2) должны поддерживаться масштабирование и фильтрация;

3) должен осуществляться запрос деталей по требованию.

Таким образом, на основании изученных работ, можно выделить следующие виды человеко-компьютерного взаимодействия для визуального анализа в области сетевой безопасности:

1) управление положением элементов модели визуализации;

2) управление масштабом;

3) управление фильтрами;

4) управление деталями по требованию;

5) управление уровнями визуализации.

Более подробно типы классификации человеко-компьютерного взаимодействия представлены ниже.

(1) Самый простой пример взаимодействия по типу «управление положением элементов модели визуализации» – функция перетаскивания элементов визуализации drag-and-drop. С её помощью оператор может получить общую картину происходящего в системе путём перемещения и сравнения различных элементов, например, сравнить по размеру два столбца диаграммы или две вершины графа.

(2) Масштабирование может осуществляться различными способами. Например, эффект рыбьего глаза позволит рассмотреть отдельные мелкие детали большой модели визуализации. Также одна и та же информация может иметь разное представление: в зависимости от ситуации, данные с иерархической структурой могут быть представлены как дерево, карта деревьев или граф, при этом между данными представлениями можно переключаться.

(3) Фильтрация позволяет группировать элементы по определённому признаку, с возможностью скрывать «лишние» элементы и отображать их заново. Также возможна группировка с цветовым выделением, обводкой по контуру, свечением, управление прозрачностью, группировка в пространстве и т.п. Например, в матрице можно сгруппировать порты по критерию уязвимости: от более уязвимых к менее уязвимым.

(4) Детали по требованию – это информация, которая не отображается на общем виде модели визуализации, однако может быть вызвана при необходимости. Например, таким образом могут быть вызваны IP-адреса устройств сети, которые не отображаются на общем виде графа сети из соображений экономии пространства.

(5) Некоторые модели визуализации отображают данные, связанные между собой. Для таких видов визуализации можно применять управление уровнями, которое выражается в изменении связей между элементами. Например, граф компьютерной сети отображает физические и логические связи между устройствами, подключенными к этой сети, и данный тип взаимодействия позволит переключаться между этими связями.

Для того чтобы не отделять данные и визуализацию друг от друга, данные функции должны быть реализованы с помощью методов взаимодействия человека и компьютера. Это становится возможным при использовании сенсорных экранов как интерфейса управления визуализацией данных сети. В качестве примера можно рассмотреть возможное взаимодействие с визуализацией корпоративной сети. При мониторинге событий безопасности требуется переходить от одного уровня модели OSI к другому – этот переход можно осуществлять с помощью сведения/разведения нескольких пальцев одной руки. При необходимости смены физических связей внутри сети на логические можно использовать жест листания влево/вправо. При этом команды вводятся с помощью сенсорных экранов, поддерживающих множественное касание.

Принцип распознавания команды интерфейсом на основе сенсорных экранов состоит в следующем: пользователь касается экрана, приложение распознаёт количество пальцев, распознаёт координаты прикосновения пальцев к экрану, при начале движения пальцами распознаёт направления движения, далее срабатывает нужная функция, привязанная к выполняемому жесту. Основными жестами являются tap (короткое касание), press (долгое касание), swipe (листание вправо-влево или вверх-вниз), pinch (сведение/разведение нескольких пальцев на экране). Однако использование жестового интерфейса на сенсорных экранах для приложений сетевой безопасности требует дополнительного изучения.

Проблемы защищённости.

В настоящее время активно изучаются уязвимости сети и программного обеспечения, однако уязвимостям интерфейсов уделяется очень мало внимания. Поэтому данная область остаётся неприкрытой и может представлять опасность для конфиденциальной информации пользователей. Привычным типом атак является перехват данных по сети либо вредоносное программное обеспечение, которое собирает данные приложений, пароли и другие чувствительные данные. Однако есть ряд атак, типичных именно для интерфейсов. Например, эти атаки связаны с тем, чтобы нарушить конфиденциальность наблюдением со стороны или каким-либо образом зафиксировать пользовательские данные (например, с помощью фото- и видеокамеры), а также нарушить конфиденциальность, целостность или доступность, вмешиваясь в процесс передачи данных от одного элемента интерфейса к другому.

Как визуализация, так и управление могут быть уязвимы для множества атак [26]. Например, атаки посредством наблюдения за действиями оператора со стороны, атаки, направленные на временное ухудшение самочувствия оператора для того, чтобы заставить его чаще ошибаться.

1.2.2 Проблемы современных человеко-компьютерных интерфейсов

Проблемы современных типов интерфейсов также лежат в области визуализации, управления и защищённости взаимодействия оператора с данными типами интерфейсов.

Проблемы сенсорных экранов. *Визуализация.* Проблемы визуализации соответствуют общим проблемам интерфейсов, так как визуализация чаще всего двухмерная.

Управление. Целенаправленное изучение разнообразных моделей взаимодействия с системами безопасности посредством жестов на сенсорных экранах только начинает развиваться. Так, в [13], [14] предлагаются определённые жесты на сенсорных экранах под конкретные задачи безопасности.

Однако существуют работы, посвящённые исследованию взаимодействия с сенсорными интерфейсами. Поскольку сенсорные экраны используются в том числе для приложений по управлению информационной и компьютерной безопасностью, были исследованы наиболее часто встречающиеся модели управления сенсорным интерфейсом.

Был рассмотрен ряд работ, в которых проводится оценка различных типов взаимодействия пользователей с сенсорным интерфейсом. В работах исследованы следующие модели взаимодействия:

1) перетягивание указателя на экране (drag), увеличение с помощью выделения области (tap and drag) [27];

2) сведение/разведение двух пальцев на экране (pinch), используется для изменения размера объекта, увеличение с помощью выделения области (tap and drag), в работе также используется также для увеличения объекта [28];

3) печать на экранной клавиатуре (type on a keyboard) [29];

4) нажатие на объект (tap), захват и перемещение объекта из одной области в другую (drag and drop), следование пальцем на экране по определённой заданной траектории (path following) [30];

5) сведение/разведение двух пальцев на экране (pinch) в исследовании также применяется для увеличения объекта, листание сверху вниз и снизу вверх (scroll) [31];

6) нажатие на объект (tap) используется для выбора объекта, перетягивание указателя на экране (drag), поворот (rotate) используется для поворота объекта вокруг своей оси, сведение/разведение двух пальцев на экране (pinch) используется для настройки величины объекта [32].

Таким образом, выявлен следующий набор взаимодействий:

1) нажатие (tap) – нажатие на объект или область;

2) печать на экранной клавиатуре (type on a keyboard) – печать на экранной клавиатуре планшета или смартфона;

3) выделение области (tap and drag) – нажатие и проведение пальцем в другую часть экрана для выделения области или увеличения области;

4) захват и перетягивание объекта (drag and drop) – распространённое взаимодействие, которое выражается «захватом» объекта на экране путём нажатия на объект и перетаскиванием его в другую область экрана путём проведения пальцем по экрану;

5) перетягивание указателя (drag) – проведение пальцем по экрану для перемещения указателя из одного угла экрана в другой;

6) сведение/разведение пальцев на экране (pinch) – сведение/разведение двух или более пальцев, как правило, используемое для управления размером объекта на экране;

7) поворот (rotation) – поворот объекта вокруг своей оси двумя пальцами на экране;

8) следование траектории (path following) – следование какой-либо траектории пальцем на экране;

9) листание (scroll) – листание снизу вверх или сверху вниз;

Авторы работ оценивали эффективность тех или иных жестов, в основном, с помощью показателей точности (accuracy) и скорости (speed). Также могли рассматриваться другие показатели, такие как усталость руки, усталость глаз, удобство, которые оценивались испытуемыми субъективно, по шкале от 1 до 5 [27].

В [27] жесты оценивались по целому ряду показателей: скорость, точность, усталость руки, усталость глаз, удобство использования, простота использования. Оценивался метод «drag» для перетаскивания курсора и выбора объекта, а также метод «tap and drag» для увеличения выделенной области.

В [28] сравнивается увеличение объекта с помощью «pinch» и «tap and drag» в соответствии с показателями скорости и точности. Предполагалось, что пользователь выполнял действия одной рукой. «Tap and drag» оказывался более быстрым методом. «Tap and drag» менее точный (86%), чем «pinch» (88,3%), однако разница не была значительной и составляла около 2%. Также учитывалось количество предпринимаемых попыток, при условии, что чем меньше попыток, тем более эффективен метод.

В [29] оценивалась эффективность взаимодействия с экранной клавиатурой, оптимизированной для людей с расстройствами чтения. Эффективность взаимодействия также оценивалась с помощью показателей скорости и точности. Оценивалась скорость печати на экранной клавиатуре и количество ошибок.

В [30] изучалась влияние тактильного отклика (вибрация) на показатели точности и скорости типов взаимодействий «tap», «drag and drop» и «path following». Тактильный отклик не повлиял на точность и скорость при касании, увеличил точность и скорость для взаимодействий типа «drag and drop», увеличил точность для взаимодействия типа «path following» при снижении скорости.

В [31] идёт сравнение метода увеличения с помощью жеста «pinch» на сенсорном экране и с помощью поворота экрана (использование гироскопа).

Точность и скорость жеста «pinch» была ниже, но при этом разница была незначительной. Однако количество попыток при использовании гироскопа было намного ниже.

Также в [31] сравнивалось печатание на сенсорной клавиатуре и ввод текста с помощью жестов рук в воздухе. Метод ввода на сенсорной клавиатуре оказался быстрее, чем выводить буквы в воздухе. Также метод клавиатурного ввода более точный. Однако пользователи указали ввод данных жестами рук в воздухе как более удобный.

В [31] также оценивался тип взаимодействия «scroll». Сравнивалось листание с помощью сенсорного экрана и с помощью нажатия на кнопки. По скорости эти два метода практически не отличаются. Листание с помощью жеста на сенсорном экране обладало более высокой точностью. Удобство листания на сенсорном экране пользователи также оценили выше.

В [32] измерялось среднее время и точность выполнения заданий, включавших жесты «tap», «drag», «rotate», «pinch».

Таблица 1 – Использование различных показателей для моделей взаимодействия в рассматриваемых работах

Тип взаимодействия	Точность	Скорость	Количество попыток	Удобство	Усталость руки	Усталость глаз
tap	[30][32]	[30][32]				
type on a keyboard	[29][31]	[29][31]		[31]		
tap and drag	[27][28]	[27][28]	[27][28]	[27]	[27]	[27]
drag and drop	[30][32]	[30][32]				
drag	[27][32]	[27][32]	[27]	[27]	[27]	[27]
pinch	[28][31][32]	[28][31][32]	[28][31]			
rotation	[32]	[32]				
path following	[30]	[30]				
scroll	[31]	[31]		[31]		

В соответствии с таблицей, можно видеть, что малое количество исследователей оценивает такие физические показатели, как удобство использования и усталость рук или глаз.

Защищённость. Исследования, посвящённые проблеме сенсорных экранов в области компьютерной безопасности, практически не затрагивают

проблему защищённости взаимодействия оператора с моделями визуализации системы безопасности, и, в основном, посвящены проблемам безопасности как таковой. Так, в [33] рассматривается система предотвращения ложных срабатываний клавиатуры при наборе пароля. В [34] предлагается метод отслеживания динамики нажатия клавиш для увеличения надёжности аутентификации. В [35] предлагается математический анализ нажатия клавиш.

Проблемы виртуальной реальности. Несмотря на то, что технологии VR/AR появились относительно недавно, они непрерывно развиваются, и перед разработчиками встают новые проблемы. В данном разделе будут рассмотрены некоторые предыдущие обзоры, в которых освещались проблемы интерфейсов VR/AR.

Часть работ предлагает классификацию уровней погружения пользователя в VR (например, [36] и [37]). В этих работах изучаются проблемы пересечения «реальности» и «виртуальности» и вводится понятие «смешанная реальность» для областей их пересечения. При этом классификация ориентируется на три основных показателя: (1) объём знаний пользователя о виртуальной среде, (2) реалистичность происходящего в виртуальной среде, (3) степень вовлечённости пользователя в виртуальную среду [36]. Таким образом, данные работы рассматривают в основном проблемы различия реального и виртуального, а также проблемы визуализации и взаимодействия. Также в [37] разрабатываются методы, чтобы с помощью AR визуализировать объекты и явления, которые человек не способен воспринимать своими органами чувств. Набор таких методов был назван «multimediated reality» как возможность «модифицировать реальность».

Визуализация. Включает в себя проблемы при размещении виртуальных объектов в пространстве: считывание поверхностей реальной среды ([38], [39], [40]) , задержка [38], расчёт необходимого расстояния ([39], [40]), правильное расположение виртуальных объектов [41], проблема

плавного переключения между помещениями и наружной средой [41] и т.д. Проблемы согласования решаются благодаря микроэлектромеханическим сенсорам (MEMS), а также достаточно мощные процессорам, которыми обладают современные мобильные устройства [42]. Также проблемы обработки данных об окружающей среде решаются развивающимися программными методами, например, машинное обучение.

Также в данную группу входят проблемы, связанные с перегруженностью визуального интерфейса ([38], [39]), перекрытием одного объекта другим ([38], [40]), перекрытием виртуальных объектов реальными и наоборот ([38], [41]). Проблема визуализации всё ещё актуальна.

Взаимодействие. Выражается в сложности подбора удобных способов взаимодействия с виртуальными объектами [40] и отсутствии обратной связи в ответ на действия пользователя [41]. Проблема взаимодействия решается различными интерфейсами: тактильными, жестовыми, гибридными и т.д., что верно как для AR, так и для VR [42].

К проблемам взаимодействия также можно отнести проблемы, связанные непосредственно с устройствами, с которыми работает оператор. Среди таких проблем указываются прозрачность дисплеев в случае очков AR [38], непривычный обзор из-за расположения камеры вдали от глаз [38], ограничения яркости и контрастности таких дисплеев [40], ограничения угла обзора дисплеев [86], портативность шлемов VR ([39], [40]) и дисплеев [86], и т.д. Решение для проблем портативности отмечено в [42], современные мобильные устройства достаточно мощные, чтобы обладать небольшими размерами. Кроме того, современные устройства обладают качественными дисплеями, при этом для AR часто применяются смартфоны и планшеты, что позволяет решить проблему непривычного обзора из-за расположения камеры. Для VR проблема портативности всё ещё остаётся актуальной.

Защищённость. Среди проблем компьютерной безопасности, связанных с VR/AR, можно выделить две категории: (1) проблемы, которые решаются с помощью интерфейсов VR/AR; (2) проблемы безопасности самих

интерфейсов VR/AR. В данном разделе представлены примеры решения задач компьютерной безопасности из первой категории.

Виртуальная/дополненная реальность чаще используются в военной сфере и медицине [43], образовании и игровой индустрии [44], однако в области информационной и компьютерной безопасности практически не изучалась. Исследователи и разработчики интерфейсов виртуальной/дополненной реальности решали конкретные задачи, обычно не связанные с информационной безопасностью, однако результаты этих исследований можно дополнить и изучить их применимость к задачам безопасности.

Так, например, в [45] описан интерфейс для мониторинга и управления роботами. В контексте информационной безопасности подобный интерфейс можно было бы использовать для управления роем дронов, которые мониторят помещение на предмет обнаружения инцидентов безопасности, таких как проникновение посторонних в закрытые помещения. В работе [46] изучается взаимодействие с 3D-графом, однако в контексте безопасности графы могут использоваться для мониторинга состояния сети, а также для изучения взаимосвязей в социальных сетях. Таким образом, возможно исследование восприятия оператором громоздких графов социальной сети в виртуальной реальности, а также разработка удобных для виртуальной реальности моделей взаимодействия с этими графами.

Исследования виртуальной реальности, которые всё же были связаны с безопасностью, либо использовали технологии виртуальной реальности (шлем виртуальной реальности и различные контроллеры) для мониторинга реальных физических объектов с помощью камер [47], что по своей сути виртуальной реальностью не является, либо решали вопросы безопасности каналов передачи данных пользователя виртуальной реальности [48], что связано с безопасностью, но не связано с виртуальной реальностью самой по себе.

Также в научном сообществе изучалась возможность адаптации интерфейса систем безопасности к физическому состоянию оператора. Например, в [49], по аналогии с отслеживанием уровня усталости водителя [50, 51], на основе данных о скорости совершаемых действий, скорости ответной реакции пользователя и точности его команд вычислялся уровень усталости оператора. Чем выше уровень усталости, тем больше размер элементов интерфейса, меньше контрастность цветовой гаммы, меньше элементов анимации. Однако возможность адаптации жестов в новых интерфейсах именно к задачам безопасности тоже практически не исследовалась.

Множество работ посвящено решению задачи контроля доступа ([52], [53], [54], [55], [56], [57]). При этом чаще всего исследуются возможности аутентификации в виртуальной среде. Так, в [52] рассматривается возможность использования привычных способов аутентификации, таких как пароли и PIN-коды. Также привычные способы аутентификации рассматриваются в [53], где авторы разрабатывают методы оценки эффективности их использования в виртуальной среде. В [54] и [55] изучается аутентификация, основанная на поведении пользователя, и включает в себя такие биометрические данные как движение руками, головой, движение глаз и т.п. Также биометрические способы аутентификации изучаются в [56]. Авторы рассматривают мозговые импульсы как шаблон для биометрической аутентификации. Работа [57] посвящена трёхмерной разновидности теста CAPTCHA, который может использоваться в виртуальной среде.

Работы [58], [59] могут служить примерами решения проблем сетевой безопасности. Обнаружение и предотвращение вторжений – важная задача в данной области, и ей посвящена работа [58], в которых авторы разрабатывают прототип фаервола для тренировок в виртуальной среде. Исследователи [59] разработали виртуальную среду для обучения

взаимодействию с сетевой аппаратурой, которая закладывает основные принципы сетевой безопасности.

Существуют работы, исследующие проблемы криптографии и шифрования, связанные с виртуальной средой. Например, в [60] изучается декодирование текста с помощью визуальной криптографии. В [61] исследуется влияние задержки при шифровании трафика с помощью протоколов TCP и TLS на качество сеансов в виртуальной среде.

Также VR используется для работы в области цифровой криминалистики. В [62] воссоздаются улики в виртуальной среде, а также из трафика и самой системы VR извлекаются необходимые данные.

Работа [63] посвящена решению проблем киберфизической безопасности. Здесь система VR является интерфейсом управления патрульным роботом. Также VR может служить интерфейсом управления для сети Интернета вещей [64] и для критически важных инфраструктур, таких как ядерная энергетика [65].

Некоторые работы рассматривают физические затруднения оператора при использовании средств VR/AR, что можно отнести к проблеме защищённости оператора во время работы с виртуальной реальностью. Например, утомление глаз, а также дезориентация пользователя в реальном пространстве после работы в виртуальной среде [38]. Данный тип проблем может решаться с помощью разработки специализированных интерфейсов, изучения проблем восприятия данных и их визуализации пользователями.

В некоторых работах поднимаются проблемы конфиденциальности пользователя [39]. Например, посторонний человек может увидеть реальные или виртуальные объекты оператора. Данная проблема актуальна на настоящий момент, так как современные приложения располагают огромным количеством личных данных пользователя. Кроме того, технологии VR/AR распространяются, и пользователи имеют возможность взаимодействовать друг с другом.

Однако, проблемы безопасности, связанные с защищённостью интерфейса виртуальной реальности, также требуют решения. Безопасность интерфейсов будет отличаться от безопасности сети, поскольку интерфейс предполагает взаимодействие с человеком, следовательно, многое будет зависеть от физических возможностей человека. Соответственно, угрозы будут делиться на три большие группы:

(1) угрозы системе;

(2) угрозы оператору;

(3) угрозы процессу взаимодействия системы и оператора.

Однако, несмотря на важность учёта влияния угроз интерфейсам на физическое состояние оператора, данная работа сосредотачивается на технической реализации данных угроз.

Таким образом, областью данного исследования являются проблемы, связанные с оцениванием уровня защищённости человеко-компьютерного интерфейса. Необходимо отметить, что процесс формирования интерфейсов не входит задачу данной работы, и задачи подбора моделей визуализации и моделей управления не входят в область настоящего исследования. Цель исследования – в создании методики оценивания защищённости интерфейса, которую можно использовать как на этапе эксплуатации, так и на этапе проектирования. С помощью этой методики можно будет оценить, какие слабые места есть в архитектуре человеко-компьютерного интерфейса, а также общий уровень защищённости интерфейса.

Таким образом, входные данные для решаемых задач являются описания архитектуры интерфейса и возможных угроз, результатом работы методики будут являться слабые места в архитектуре интерфейса и общий уровень его защищённости.

Предлагаемый подход учитывает требования российских стандартов, таких как ГОСТ Р ИСО/МЭК 15408-1-2008 [66] и ГОСТ Р ИСО/МЭК 15408-2-2008 [67], и предполагает, что обеспечение безопасности включает в себя защиту активов от угроз.

Предлагаемая в данной работе методика позволяет, основываясь на модели уязвимостей для интерфейса, оценить возможный ущерб данным пользователя со стороны интерфейса. Таким образом, методика оценки защищённости интерфейсов позволяет повысить осведомленность оператора о возможных рисках для безопасности системы со стороны интерфейса.

1.3 Современное состояние

Когда речь заходит об оценке качества интерфейсов, то оцениваются в первую очередь показатели, связанные с его эргономичностью. Однако практически не уделяется внимание вопросам, связанным с его безопасностью. Существует ряд работ, посвящённых отдельным уязвимостям интерфейсов, однако не предпринималось попыток количественно оценить ущерб, наносимый этими уязвимостями.

Человеко-компьютерный интерфейс – сложный комплекс взаимодействия, включающий в себя человека-оператора и компьютерную систему, с которой он взаимодействует. Уязвимости человеко-компьютерных интерфейсов можно разделить на три большие группы: (1) уязвимости со стороны оператора, (2) уязвимости со стороны системы и (3) уязвимости при пересечении двух сред, виртуальной и реальной.

Кроме того, как человек, так и компьютер имеют способность принимать и передавать информацию, а также хранить и обрабатывать полученные данные. По этой причине, первые две группы можно разделить ещё на три категории. Для уязвимостей со стороны оператора категории будут таковы: (1) уязвимости для органов чувств, (2) уязвимости для психики и восприятия и (3) уязвимости при передаче информации машине. Уязвимости со стороны системы могут быть разделены следующим образом: (1) уязвимости ввода, (2) уязвимости при хранении и обработке данных и (3) уязвимости вывода.

Также оператор может взаимодействовать с другими людьми и объектами, находящимися как в реальной, так и в виртуальной среде. По этой

причине могут возникнуть уязвимости, связанные с пересечением реальной и виртуальной сред.

Данная модель возможных уязвимостей представлена на схеме ниже (рисунок 4):

Рисунок 4 – Схема взаимосвязей уязвимостей разных компонентов системы взаимодействия «Человек – Виртуальная среда».

Рассмотрим более подробно примеры уязвимостей для каждой из категорий полученной модели.

1.3.1 Защищённость интерфейса на основе сенсорных экранов

Уязвимости со стороны оператора. *Угрозы органам чувств.* В данную область угроз входят все те же самые угрозы, что и для «традиционных» визуальных интерфейсов, так как визуализация на сенсорных экранах не отличается от обычной. Также существует ряд работ, посвящённый атакам, угрожающим самочувствию пользователя, например, мигающие изображения, вызывающие усталость, раздражение и дискомфорт у здоровых людей и провоцирующие приступ у больных эпилепсией [68, 69].

Угрозы восприятию. В данную категорию можно включить уязвимости при восприятии оператором данных визуализации, а также уязвимости, связанные с управлением визуализацией. Например, с помощью подбора соответствующей визуализации злоумышленник может спровоцировать ошибку восприятия, при которой данные могут быть неверно

интерпретированы [70]. Данная ошибка восприятия может быть вызвана тем, что графические примитивы, использующиеся при построении визуализации, неправильно нормализованы [2].

Угрозы при процессе передачи информации системе. Существует ряд причин, по которым процесс ввода информации пользователем может быть нарушен. Например, причиной может послужить непривычность интерфейса для пользователя. Так, в работе [71] описывается сложность в освоении сенсорного интерфейса пожилыми людьми. Элементы интерфейса могут быть маленького размера, и с ними неудобно взаимодействовать, повышается процент ошибок при взаимодействии.

Уязвимости со стороны системы на основе сенсорных экранов. *Угрозы при вводе информации.* Может возникать угроза обмана пользователя: примером могут служить активные элементы интерфейса, ведущие на мошеннические сайты и неожиданно появляющиеся в тех местах, куда пользователь собирался нажать [72]. Также возможны сценарии, при которых такой мошеннический элемент интерфейса маскируется под легитимный. Также злоумышленник может сделать его прозрачным. Могут собираться различные конфиденциальные данные на основе паттернов жестов пользователя [73], такие как пол [74, 75], возраст [76, 77], состояние здоровья [78] и т.д. Может осуществляться перехват данных с экранной клавиатуры.

Угрозы при выводе информации. Данная группа угроз состоит в том, что данные могут отображаться некорректно. Элементы интерфейса могут перекрывать контент. Пользователь может самостоятельно перекрывать контент пальцем, пытаясь взаимодействовать с элементами интерфейса, такими как кнопки, выпадающие списки и т.д.

Угрозы при обработке/хранении информации. На этапе обработки и хранения данных могут возникать угрозы типичных сетевых атак, повреждающих данные, делающих эти данные недоступными или наносящих ущерб конфиденциальности пользователя. Примерами таких угроз могут

быть использование вредоносных приложений, вирусов и троянов, атаки типа «человек посередине» (MITM) [79], и т.д.

Пересечение реальной и виртуальной сред. При взаимодействии пользователя с устройством может возникнуть ряд угроз со стороны окружения. Сенсорные устройства уязвимы к атакам через подглядывание (shoulder-surfing attacks), так как процесс работы пользователя с приложением может быть зафиксирован камерой или сторонним наблюдателем [80]. Также по следам от пальцев пользователя можно определить комбинацию чисел или графический ключ для разблокировки [81]. В исследовании [80] рассмотрен ряд мер, которые могут быть приняты для защиты процесса аутентификации от атак данных типов.

Методы биометрической аутентификации для сенсорных устройств также могут быть уязвимы, например, отпечаток пальца можно подделать [82]. Также возможно обмануть систему распознавания лиц с помощью фото, трёхмерного изображения лица, видео, специального макияжа [83] и т.д. Методы борьбы против уязвимости биометрической аутентификации: использовать для аутентификации паттерны движений пользователя во время взаимодействия с кнопочными интерфейсами и смартфонами [84].

Уязвимости сенсорного экрана, в основном, касаются только эргономики и не связываются с компьютерной или информационной безопасностью.

1.3.2 Защищённость интерфейса на основе виртуальной/дополненной реальности

В случае VR/AR, человеко-компьютерный интерфейс включает в себя не только компьютерную систему и взаимодействующего с ней оператора, но также среду, которая их окружает. Эта среда обладает разными уровнями по шкале «реальность – виртуальность» [36], поскольку возможны различные степени пересечения виртуальной и реальной сред и различные степени дополненности реальной среды.

Уязвимости со стороны оператора. *Угрозы органам чувств.* Множество работ посвящено изучению влияния VR на органы чувств оператора и на его физическое самочувствие. Так, работы [85], [86], [87] упоминают сенсорную перегрузку, которая может быть вызвана с помощью мигания экрана, изменения яркости, громких звуков. В некоторых работах рассматривается влияние на оператора с помощью аудиоинтерфейса [88], что верно также и для VR, так как головные дисплеи часто имеют возможность подключения аудио-гарнитуры. В работах часто упоминается так называемая кибернетическая болезнь [89], которая выражается в ощущениях физического недомогания оператора от присутствия в VR [90], [91], [92] (тошнота, головокружение, временная дезориентация в пространстве), однако в исследовании [91] связь между присутствием в VR и дискомфортом отрицается. В статье [93] также предлагается правовое регулирование технологии VR на основании недостаточного количества исследований о влиянии VR на организм человека.

Угрозы психике и восприятию. Ряд работ рассматривает угрозы психике и восприятию, которые, предположительно, могут повлиять на психическое здоровье пользователя и заставить его утратить связь с физической реальностью [93]. Также с помощью данного типа уязвимостей можно обмануть восприятие оператора и заставить его некорректно воспринимать VR [94], например, изменить внешний вид виртуальной среды так, что пользователь будет натыкаться на физические объекты помещения, в котором находится.

Угрозы при процессе передачи информации системе. Найдены работы, посвящённые изучению связи между виртуальными аватарами пользователей и их поведением [95], [96]. Такая связь не рассматривалась как уязвимость, однако это может быть источником угрозы при процессе передачи информации системе. Например, взаимосвязь между внешними атрибутами аватара другого пользователя и социальным поведением испытуемого [95] может дать возможность для социального инжиниринга. Также пользователь

может бессознательно повторять действия своего аватара [96], что предоставляет возможность заставить оператора повторить какой-либо простой жест и таким образом воздействовать на жестовый интерфейс находящегося рядом устройства.

Уязвимости со стороны системы VR/AR. *Угрозы ввода информации.* Работы, посвящённые угрозам ввода информации, рассматривают возможные нарушения конфиденциальности ввиду самопроизвольного включения камеры на устройствах VR ([94], [97]), наблюдения за пользователем во время работы ([98], [99]), а также перехвата визуальных данных [85] или голосовых данных [88] о виртуальной среде. Кроме того, в работе [99] показано, что базовые станции системы виртуальной реальности уязвимы к атаке с помощью источника инфракрасного излучения, которая может причинить ущерб целостности и доступности пользовательских данных.

Угрозы вывода информации. Угрозы вывода информации могут заключаться в следующем. Изменение внешнего вида виртуальной среды [94], при котором будет отображаться ложная информация (в случае AR и VR), либо пользователь будет сталкиваться с физическими объектами реальной среды, находясь в VR. Может происходить посредством вредоносных приложений [100], например, приложение может скрыть дорожные знаки или другие реальные объекты на дороге [101]. Также рассматривались угрозы конфиденциальности: в [102] авторы предлагали защитить информацию с помощью QR-кодов с закодированным в них контентом при выводе данных AR, в [103] рассматривались стратегии по использованию носимых устройств, в [104] предлагают адаптивный подход к разработке политик безопасности визуального вывода устройств AR. В [97] озвучена проблема перехвата и подмены данных для вывода недоверенными приложениями.

Угрозы при обработке/хранении информации. Исследование [85] описывает вмешательство в обрабатываемые системой данные, например, с

целью дальнейшей подмены информации, которая выводится на экраны. Также виртуальные системы подвержены общеизвестным сетевым атакам (например, прослушивание пакетов [87], распространение вредоносного кода [87] или DDoS [105]), нарушающим конфиденциальность и целостность данных, либо искажающим эти данные [87]. Исследование [97] также упоминает возможность использования сохранённых данных третьей стороной.

Пересечение реальной и виртуальной сред. При взаимодействии с другими людьми во время сеанса VR/AR могут возникнуть угрозы нарушения конфиденциальности и несанкционированного доступа. В [97] рассматривается возможность непреднамеренно показать виртуальный или реальный объект, который не должен быть виден другим пользователям. Целый ряд работ ([98], [106], [107], [108], [109], [110], [111]) изучает возможности проведения атаки через наблюдение за процессом аутентификации, а также предлагает методы борьбы с этой атакой через повышение надёжности аутентификации. Авторы [98] решают проблему ущерба конфиденциальности на пересечении виртуальной и реальной сред с помощью управления политиками безопасности. В исследовании [106] проблема утечки паролей при аутентификации решалась с помощью выбора трёхмерных объектов в виртуальной среде. Исследование, проведённое в работах [107, 108], рассматривает возможность использования трёхмерных кубов с нанесёнными на них числами с целью нераскрытия кодовой комбинации возможному наблюдателю. В работе [109] значения на цифровой клавиатуре перемешивались с целью снижения риска перехвата числового кода. Авторы [110] решают проблему возможного ущерба конфиденциальности с помощью визуальной криптографии. В исследовании [111] показано, что для повышения надёжности аутентификации могут быть использованы биометрические методы аутентификации, такие как паттерны движений рук при наборе PIN-кода.

На основе рассмотренных примеров уязвимостей интерфейса и в соответствии с описанной моделью уязвимостей была составлена таблица 2.

Таблица 2 – Классификация выявленных проблем интерфейса VR.

Тип уязвимости	Уязвимость интерфейса	Примеры угроз	Примеры статей
Оператор	Органы чувств	Угрозы самочувствию человека	[85][86][87][89][90] [91][92][93]
		Акустические атаки	[88]
		Атаки на визуальный канал	[86]
	Психика и восприятие	Искажение восприятия человека	[94]
		Угрозы психическому состоянию человека	[93]
	Передача данных системе VR	Атаки через психологию поведения	[95][96]
Система VR/AR	Ввод	Угрозы конфиденциальности	[85][88][94][97][98] [99]
	Обработка и хранение данных	Искажение/подмена информации	[85][87]
		Прослушивание данных	[87][97]
		Атаки типа «отказ в обслуживании»	[105]
	Вывод	Изменение визуализации через вредоносный код или приложения	[94][97][100][101]
		Ошибки визуализации	[101]
		Угрозы конфиденциальности	[102] [104]
		Несанкционированный доступ к устройствам вывода	[103]
Пересечение виртуальной и реальной сред	Взаимодействие с другими людьми	Атаки через наблюдение	[98][106][107][108] [109][110][111]
		Угроза конфиденциальности	[97]

Из проведённого исследования видно, что самые хорошо изученные угрозы – это атаки на систему, где происходит обработка и хранение данных. Эти атаки являются наиболее привычными для компьютерной безопасности. Угрозы при взаимодействии с другими людьми на пересечении виртуальной и реальной сред можно рассматривать как проблему, типичную для VR/AR. Однако этой проблеме посвящено довольно много исследований, по сравнению с остальными. Наименее изученной областью оказалось влияние технологии виртуальной реальности на психику и восприятие человека. При этом влияние этой технологии на психологию поведения, которая может

оказаться уязвимостью на этапе передачи данных системе виртуальной реальности, ранее не рассматривалось как угроза. Таким образом, угрозы при передаче данных системе виртуальной реальности можно расценивать как новую проблему.

Существует небольшое количество обзорных работ, рассматривающих уязвимости взаимодействия пользователя с интерфейсом виртуальной или дополненной реальности с точки зрения информационной безопасности. Однако количество работ, посвящённых систематизации угроз в этой области, с учётом постепенного устаревания исследований, всё ещё не является достаточным.

Таким образом, руководствуясь исследованиями современных интерфейсов (сенсорные экраны и VR/AR), можно выделить три основные группы угроз данным типам интерфейсов:

(1) угрозы самому интерфейсу;

(2) угрозы оператору интерфейса;

(3) угрозы, возникающие при взаимодействии на пересечении виртуальной и реальной сред.

Первые две группы можно также разделить на три категории. (1) делится на ввод данных, обработку/хранение данных и вывод данных; (2) можно разделить на угрозы самочувствию, угрозы психике и восприятию, угрозы при передаче данных системе.

Однако, сравнительный анализ имеющихся источников позволяет сделать следующий вывод: в настоящий момент времени не выявлено общепринятой системы классификации уязвимостей интерфейсов. По этой же причине нет и единой системы оценивания уязвимостей интерфейсов. Существует общая система оценки уязвимости (CVSS [123]), которая применяется преимущественно для оценки безопасности хостов компьютерных сетей и оценивает ущерб от уязвимостей программного обеспечения и сетевых протоколов. Также известны открытые базы данных уязвимостей хоста, такие как CVE, OSVDB, NVD [144, 145], которые

содержат информацию об уязвимостях и ссылки на эксплойты, однако эти уязвимости не связаны с интерфейсами. По этой причине данные типы уязвимостей не могут быть применены к процессу человеко-компьютерного взаимодействия. Кроме того, система оценки CVSS не учитывает ряд важных особенностей интерфейсов, поэтому в нынешнем виде не может быть применена к оценке защищённости интерфейсов. Таким образом, система CVSS оценивает уязвимости, однако не отображает уровень защищённости человеко-компьютерного интерфейса. При этом CVSS не учитывает параметры, характерные для человеко-компьютерных интерфейсов, такие как урон самому оператору от атаки на интерфейс.

Без подобной системы для оценивания человеко-компьютерных интерфейсов невозможно оценить ущерб, который могут нанести атаки с применением этих уязвимостей. Поэтому задача данного исследования – разработать методику оценивания уровня критичности уязвимостей у современных видов интерфейсов, таких как сенсорные экраны и виртуальная реальность.

1.4 Требования к системам оценивания защищённости интерфейса

Анализ современных источников в области человеко-компьютерного взаимодействия с системами информационной и компьютерной безопасности выявил следующую проблему. Современные интерфейсы, основанные на сенсорных экранах и виртуальной реальности – молодая развивающаяся область, поэтому очень мало работ посвящается изучению взаимодействия пользователей с этими интерфейсами. Данное обстоятельство порождает недостаток исследований уязвимостей этих интерфейсов. По этой причине на настоящий момент всё ещё отсутствует общая система оценки уровня критичности этих уязвимостей, следовательно, невозможно оценить защищённость интерфейса взаимодействия оператора с приложением информационной безопасности.

В целях разрешения этого противоречия проводилась разработка методики оценивания современных человеко-компьютерных интерфейсов,

использующих технологии сенсорных экранов и виртуальной реальности, которые могут использоваться, в том числе, для управления моделями визуализации в системах компьютерной безопасности. Основанием данной методики является интегрированная модель человеко-компьютерного взаимодействия с различными системами, в том числе, системами безопасности. Данная модель включает в себя модели компонента собственно взаимодействия оператора с системой безопасности, модели визуализации данных, обрабатываемых системами безопасности, и систему оценивания уязвимостей интерфейса. При этом интегрированная модель будет обеспечивать как взаимодействие со стороны системы (обработанные данные будут визуализироваться системой и показываться пользователю), так и взаимодействие со стороны пользователя (пользователь взаимодействует с визуализацией, таким образом, система получает новую информацию, вновь обрабатывает её и снова визуализирует). Ключевой особенностью работы также является то, что оценка интерфейса будет производиться с точки зрения того, как интерфейс оператора влияет на обеспечение безопасности.

Сравнительный анализ исследовательских работ по направлению оценивания защищённости человеко-компьютерных интерфейсов позволил определить требования к компонентам системы оценивания защищённости человеко-компьютерного интерфейса, за основу которой должны быть взяты разрабатываемые модели, алгоритмы и методика. Требования к системе оценивания защищённости человеко-компьютерного интерфейса могут быть как функциональные, так и нефункциональные. С помощью функциональных требований задаётся набор функций системы оценивания защищённости интерфейса. В качестве нефункциональных требований выступают ограничения по отношению к ресурсам, которые должна потреблять система оценивания защищённости интерфейса. При этом ресурсы могут быть временными, энергетическими и т.д.

Ниже определены функциональные требования к разрабатываемой системе оценивания защищённости интерфейса. Предложенная методика должна позволять создавать модели для:

1) общей архитектуры интерфейса;

2) характеристик отдельных элементов интерфейса;

3) политик безопасности, применяемых для оцениваемого интерфейса, а также возможных контрмер (например, аутентификация).

Также предлагаемая система оценивания должна оценивать общий уровень защищённости интерфейса. То есть, с помощью такой системы оператор должен иметь возможность рассчитывать показатели уязвимости интерфейса на основе разработанной модели уязвимостей:

1) на этапе проектирования интерфейса и подбора его элементов;

2) в процессе эксплуатации интерфейса, в случае если были внесены какие-либо изменения в его архитектуру (подбор элементов с другими характеристиками, замена отдельных элементов, появление новых уязвимостей и т.д.);

3) во время выбора мер по противодействию угрозам интерфейсу, чтобы оценить их влияние на показатели защищённости интерфейса (например, при повышении защищённости одного из элементов интерфейса, можно ухудшить общую оценку защищённости интерфейса);

4) во время выбора мер по противодействию угрозам интерфейсу, чтобы оценить их влияние на показатели удобства использования интерфейса (например, при повышении защищённости одного из элементов интерфейса, можно ухудшить удобство использования).

Также стоит отметить, что, кроме повышения защищённости интерфейса, требуется при этом не ухудшить или лишь незначительно ухудшить его удобство использования. Так, предложенная методика оценки защищённости интерфейсов позволит через повышение осведомленности оператора об уязвимостях повысить защищённость системы, а также

оценить, понизились ли и насколько понизились показатели удобства использования интерфейса.

Нефункциональные требования можно разбить на группы, соответствующие двум показателям затрат ресурсов во время принятия решений: (1) оперативность и (2) ресурсопотребление.

Оперативность – свойство системы оценивания защищённости интерфейса, позволяющее формировать результат за минимальный отрезок времени. Требование к оперативности можно задать в следующем виде:

$$TIME_I \leq \min_{s \in S} TIME_I^S, \quad (1)$$

где $TIME_I$ – время, которое необходимо для получения оценки защищённости интерфейса I, применяя предлагаемую систему оценивания, S – множество систем оценивания защищённости, $TIME_I^S$ – время, затрачиваемое на анализ защищённости интерфейса для системы оценивания защищённости $s \in S$. Также следует учитывать, что время $TIME_I^S$ может зависеть от этапа жизненного цикла интерфейса. Оценивание защищённости интерфейса на этапе его проектирования может занимать больше времени, чем на этапе эксплуатации, так как возможно наличие множества вариантов архитектуры одного и того же интерфейса, которые нужно проанализировать.

Поскольку речь идёт о системах информационной безопасности, анализ состояния защищённости системы должен проводиться за время, близкое к реальному времени. По этой причине необходимо задать некоторую константу, которую недопустимо превышать. Данное требование можно выразить в виде следующей формулы:

$$P_{\text{ОП}}(TIME_I \leq TIME^{\text{ДОП}}) \geq P_{\text{ОП}}^{\text{ДОП}}, \quad (2)$$

где $P_{ОП}$ – вероятность получения результата анализа защищённости в течение заданного времени, $TIME^{\text{ДОП}}$ – допустимая величина временных

затрат на оценивание защищенности, $P_{ОП}^{\text{ДОП}}$ – допустимое значение вероятности получения результата анализа защищённости.

На основании результатов экспертной оценки, а также проведённых экспериментов, значение $TIME^{\text{ДОП}}$ было выбрано равным 3 минутам для оценивания одной уязвимости (поскольку данные вводит оператор). Основываясь на исследованиях систем оценивания защищённости сети [112] было определено максимальное значение времени реакции системы – 30 секунд.

Ресурсопотребление отражает, какие ресурсы должны быть затрачены на реализацию анализа защищённости интерфейса, в том числе программные, аппаратные и кадровые ресурсы, необходимые данные, и т.д., а также их количество. Требования к данному свойству могут быть заданы следующей формулой:

$$P_{\text{РЕС}}(r \le R^{\text{ДОП}}) \ge P_{\text{РЕС}}^{\text{ДОП}}, \quad (3)$$

где r – ресурсы, необходимые для оценивания защищённости интерфейса, $R^{\text{ДОП}}$ – допустимое значение затрачиваемых ресурсов, $P_{\text{РЕС}}$ – вероятность того, что r не превышает $R^{\text{ДОП}}$, $P_{\text{РЕС}}^{\text{ДОП}}$ – допустимый минимальный порог такой вероятности. При этом на работу операционной системы компьютера также затрачиваются ресурсы, однако работа операционной системы в данном случае не учитывается. По этой причине допустимая затрата ресурсов $R^{\text{ДОП}}$ задаётся равной 30% ($R^{\text{ДОП}} = 0{,}3$) от суммарного объёма затрачиваемых ресурсов.

Рассмотренные выше свойства и соответствующие им требования представлены в таблице 3.

Таблица 3 – Требования к системе оценивания защищённости интерфейса

Свойство	**Показатели**	**Требования**
Оперативность	Вероятность того, что время, необходимое для получения результата оценки защищенности, не	$P_{\text{ОП}}(TIME_I \le TIME^{\text{ДОП}}) \ge P_{\text{ОП}}^{\text{ДОП}}$

Свойство	Показатели	Требования
	будет превышать допустимое значение.	
Ресурсо-потребление	Вероятность того, что количество использованных ресурсов не будет превышать допустимое значение.	$P_{\text{РЕС}}(r \le R^{\text{ДОП}}) \ge R_{\text{РЕС}}^{\text{ДОП}}$

1.5 Постановка задачи исследования

Поставленная задача диссертационного исследования заключается в следующем:

(1) разработать модели уязвимостей интерфейсов;

(2) разработать алгоритмы расчёта показателей уязвимостей человеко-компьютерных интерфейсов;

(3) разработать методику оценивания защищённости человеко-компьютерных интерфейсов;

(4) разработать архитектуру системы оценивания защищённости интерфейсов.

Целевой функцией методики оценки защищённости интерфейсов является минимизация уязвимости человеко-компьютерного интерфейса при соблюдении требований к другим свойствам (оперативность и ресурсопотребление). Задача исследования заключается в поиске целесообразного интерфейса с минимальной уязвимостью при определенных условиях. Суть задачи состоит в том, что имеется N различных интерфейсов, каждый из которых характеризуется собственным множеством уязвимостей $\mathrm{VULN_n}$ со своими рисками $\mathrm{BS_{in}(I_n)}$. Требуется найти целесообразный вариант интерфейса $\mathrm{I_0}$, обеспечивающий минимизацию возможных рисков для безопасности $\mathrm{BS_{\sum}(I_0)}$ (4):

$$BS_{\sum}(I_0) = \min_{n \in N} \sum_{i=1}^{VULN_n} BS_{in}(I_n) \qquad (4)$$

Требования к *оперативности* $P_{\text{ОП}}(TIME_I \le TIME^{\text{ДОП}}) \ge P_{\text{ОП}}^{\text{ДОП}}$, где $P_{\text{ОП}}^{\text{ДОП}} = 0{,}99$, допустимое время оценки защищенности $TIME^{ДОП}$ = 1 минута для этапа проектирования системы взаимодействия.

Требования к *ресурсопотреблению*: $P_{\text{РЕС}}(r \le R^{\text{ДОП}}) \ge P_{\text{РЕС}}^{\text{ДОП}}$, где $P_{\text{РЕС}}^{\text{ДОП}} = 0{,}99$, $R^{\text{ДОП}} = 0{,}3$ (30% от суммарного количества ресурсов, которые могут быть использованы приложениями). Предъявляются к следующим видам ресурсов: оперативная память компьютера, объем жесткого диска, а также процессорное время, используемое системой оценивания. Таким образом, требования к системе принимают следующий вид (таблица 4):

Таблица 4 – Требования к системе оценивания защищённости интерфейса

Свойство	**Показатели**	**Требования**
Оперативность	Вероятность того, что время, необходимое для получения результата оценки защищенности.	$P_{\text{ОП}}(TIME_I \le 1 \text{ мин}) \ge 0{,}99$
Ресурсо-потребление	Вероятность того, что количество использованных ресурсов не будет превышать допустимое значение.	$P_{\text{РЕС}}(r \le 0{,}3) \ge 0{,}99$

Таким образом, задачей диссертационной работы является разработка:

1) модели человеко-компьютерного интерфейса *I*, которая должна содержать все данные об элементах анализируемого человеко-компьютерного интерфейса, необходимые для оценивания интерфейса, в том числе: взаимосвязи между элементами, сведения о программной части, сведения об аппаратной части, настройки системы защиты информации в системе (например, аутентификация по паролю или биометрии), и т.д.;

2) модели уязвимостей интерфейса *M*, которая определяет уязвимые элементы системы и возможности использования данных уязвимостей для проведения различных типов атак;

3) алгоритма оценивания защищённости интерфейса, позволяющего оценивать различные типы уязвимостей интерфейса и общего уровня защищённости человеко-компьютерных интерфейсов;

4) методики оценивания защищённости интерфейсов, использующей все разработанные модели и алгоритмы на соответствующих этапах жизненного цикла системы человеко-компьютерного взаимодействия;

5) архитектуры системы оценивания защищенности человеко-компьютерных интерфейсов на основе моделирования уязвимостей и расчёта показателей уязвимостей, реализующей предложенную методику оценивания уязвимостей.

Далее следует более детально рассмотреть алгоритмы и методику.

В качестве входных данных для алгоритма оценивания защищённости интерфейса используется (I, M), где I – модель анализируемого интерфейса, а $M = <m_1, m_2, \ldots, m_i>$ – модели уязвимостей, где $m_1, m_2, \ldots, m_i$ – отдельные уязвимости. Результатом работы данного алгоритма является (S), где S – суммарный показатель защищённости интерфейса, учитывающий критичности всех уязвимостей, оцениваемых в системе, т.е. количественный показатель того, насколько защищена система. Таким образом, результат работы алгоритма отражает состояние защищённости анализируемого интерфейса относительно существующих уязвимостей интерфейсов.

1.6 Выводы по главе 1

1. Защищённость человеко-компьютерных интерфейсов – важный параметр, особенно применительно к системам информационной и компьютерной безопасности, однако данному параметру не уделяется достаточно внимания. По этой причине при проектировании интерфейсов могут не учитываться возможные уязвимости, которые могут принести ущерб конфиденциальности, целостности и доступности данных, с которыми взаимодействует оператор. В настоящее время не выявлено систем оценивания уровня защищённости человеко-компьютерных интерфейсов. Поэтому задача разработки методики оценивания уровня защищённости человеко-компьютерных интерфейсов, учитывающей все существующие на данный момент типы уязвимостей современных видов интерфейсов

(виртуальная реальность и сенсорные экраны) является актуальной и не решённой в настоящее время.

2. В данной главе обосновывается разработка методики оценивания защищённости интерфейса. Показано, что данная методика должна основываться на оценивании уязвимостей элементов современных типов интерфейсов (виртуальная реальность и сенсорные экраны) на основе данных об анализируемом человеко-компьютерном интерфейсе (элементы интерфейса, связи между элементами, настройки безопасности при наличии), определённых моделей уязвимостей и будет применяться на различных этапах существования интерфейса (проектирование и эксплуатация). Для возможности разработки данной методики должны быть определены модели интерфейсов, а также модели их уязвимостей. Использование данной методики в существующих средствах информационной безопасности позволит оценивать уровень защищённости интерфейса против обширного множества уязвимостей, повысить качество проектирования человеко-компьютерных интерфейсов для приложений информационной безопасности; (3) повысить защищённость интерфейса на этапе эксплуатации с помощью оценивания контрмер (закрыть какую-то уязвимость и заново оценить изменённый интерфейс).

3. Основные задачи при разработке данной методики заключаются в следующем: (1) модели интерфейсов I; (2) модели уязвимостей M; (3) алгоритмы оценивания уязвимостей; (4) методика применения алгоритмов оценивания уязвимостей для оценивания защищённости интерфейса; (5) архитектура системы оценивания защищённости человеко-компьютерных интерфейсов на основе оценивания их уязвимостей.

4. Сформулированная задача определяет цель проектирования методики как поиск интерфейса с минимальной уязвимостью среди множества оцениваемых интерфейсов. Выполнение этой задачи необходимо для правильного оценивания уровня защищённости системы взаимодействия оператора с интерфейсом в различных приложениях, в том числе в

приложениях информационной безопасности на разных этапах существования интерфейса. При этом должны соблюдаться требования к оперативности и ресурсопотреблению процессов оценивания защищенности.

ГЛАВА 2 МОДЕЛИ И АЛГОРИТМЫ ОЦЕНИВАНИЯ ЗАЩИЩЕННОСТИ ЧЕЛОВЕКО-КОМПЬЮТЕРНОГО ИНТЕРФЕЙСА

Атаки, затрагивающие человеко-компьютерный интерфейс, подразделяются на две категории: (1) атаки на интерфейс и (2) атаки на пользователя через интерфейс. Несмотря на то, что интерфейсы обладают большим разнообразием, они также обладают набором общих характеристик.

Таким образом, необходимо разработать модели объекта угрозы (человеко-компьютерного интерфейса) и причины угрозы (уязвимостей).

2.1 Модели интерфейсов

В процессе исследования было разработано две модели человеко-компьютерного интерфейса: концептуальная модель, отображающая потоки взаимодействия между интерфейсом и когнитивным аппаратом человека, и модель данных интерфейса, использующая теоретико-множественный подход. Более подробно модели интерфейсов рассмотрены ниже.

2.1.1 Концептуальная модель

Концептуальная модель взаимодействия внутри интерфейса «система-оператор» предполагает взаимодействие оператора с интерфейсом и интерфейса с оператором. Интерфейс состоит из компонента визуализации (который отображает обработанные данные) и компонента управления (который позволяет взаимодействовать с этими данными через визуализацию).

Данная концептуальная модель (рисунок 5) учитывает когнитивный аппарат оператора. Под когнитивным аппаратом понимается (1) восприятие оператора и (2) обработка оператором полученной информации. Процесс взаимодействия выглядит следующим образом. Оператор видит данные, отображаемые на визуализации, то есть воспринимает данные органами чувств (зрительный канал восприятия в случае визуализации). Далее оператор трактует полученные данные, иными словами, происходит

обработка данных визуализации. Затем оператор взаимодействует с моделью визуализации в соответствии с тем решением, которое он принимает на основе трактовки данных, которые он получил.

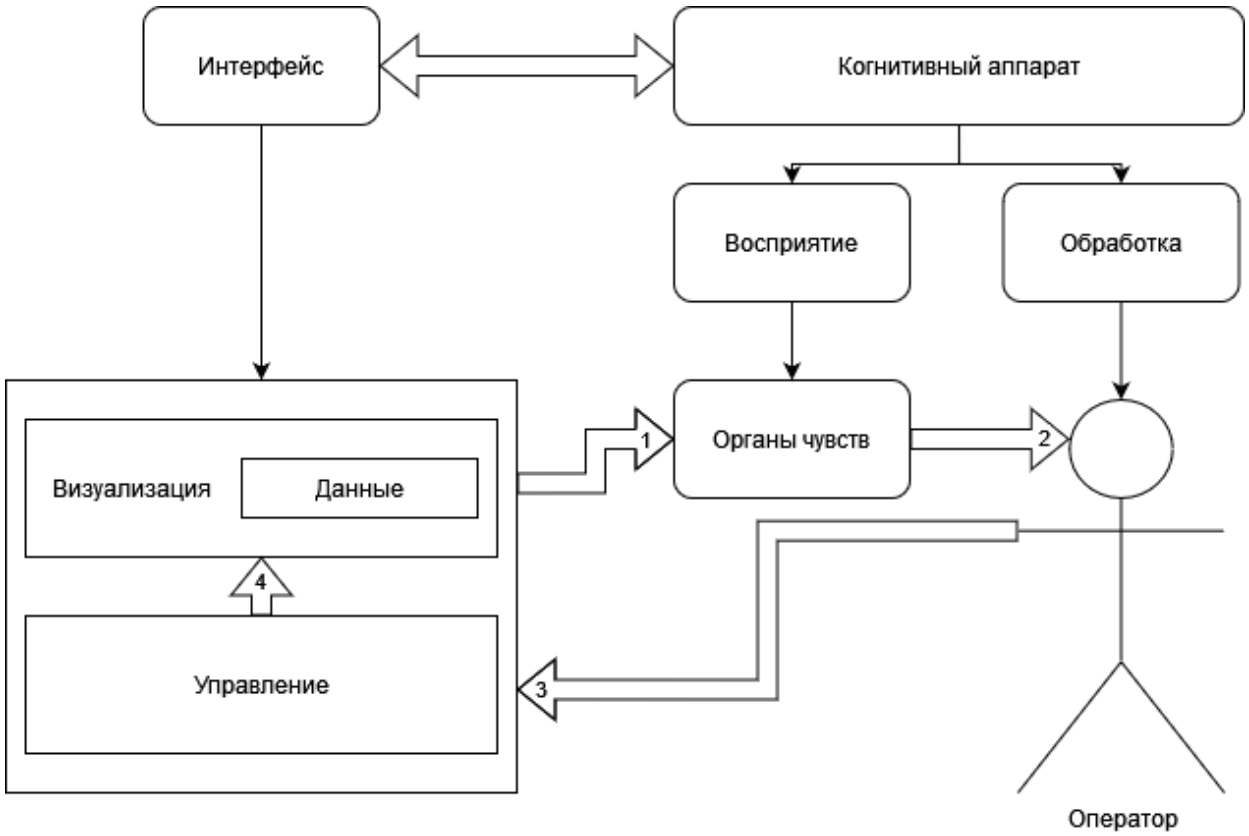


Рисунок 5 – Концептуальная модель человеко-компьютерного интерфейса

Можно выделить четыре типа взаимодействия между системой и оператором:

1) Компонент визуализации → Органы чувств оператора. Данные, собранные системой, отображаются в виде модели визуализации, которую оператор воспринимает своими органами чувств.

2) Органы чувств оператора → Мозг оператора. От органов чувств информация передаётся мозгу оператора, после чего обрабатывается, и оператор принимает решение.

3) Оператор → Компонент управления. На основе принятого решения оператор вносит изменения в систему с помощью компонента управления интерфейса.

4) Компонент управления → Компонент визуализации. Внесённые изменения отображаются с помощью модели визуализации.

Таким образом, получается кольцо взаимодействия между оператором и системой. Процессы обработки информации мозгом человека не рассматриваются в данной работе, так как находятся в области медицинских, а не технических наук. По этой причине данная работа концентрируется на 3 потоках из данной модели (рисунок 6): (1) компонент визуализации → оператор, (2) оператор → компонент управления, (3) компонент управления → компонент визуализации.

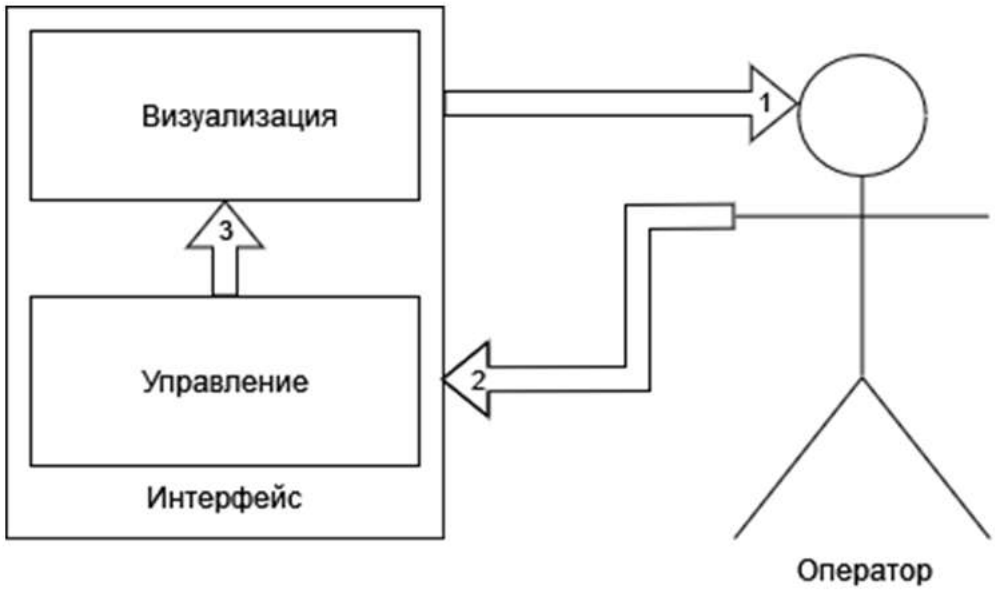


Рисунок 6 – Кольцо взаимодействия оператора с интерфейсом

2.1.2 Теоретико-множественная модель

Для того чтобы описать человеко-компьютерные интерфейсы, содержащие различные элементы, была разработана модель интерфейса. Данная модель содержит модели элементов человеко-компьютерных интерфейсов на основе сенсорных экранов и виртуальной реальности, и связи между этими элементами. Предлагаемую модель можно описать с помощью теории множеств. На рисунке 7 показаны основные элементы человеко-компьютерного интерфейса, которые должна описывать модель.

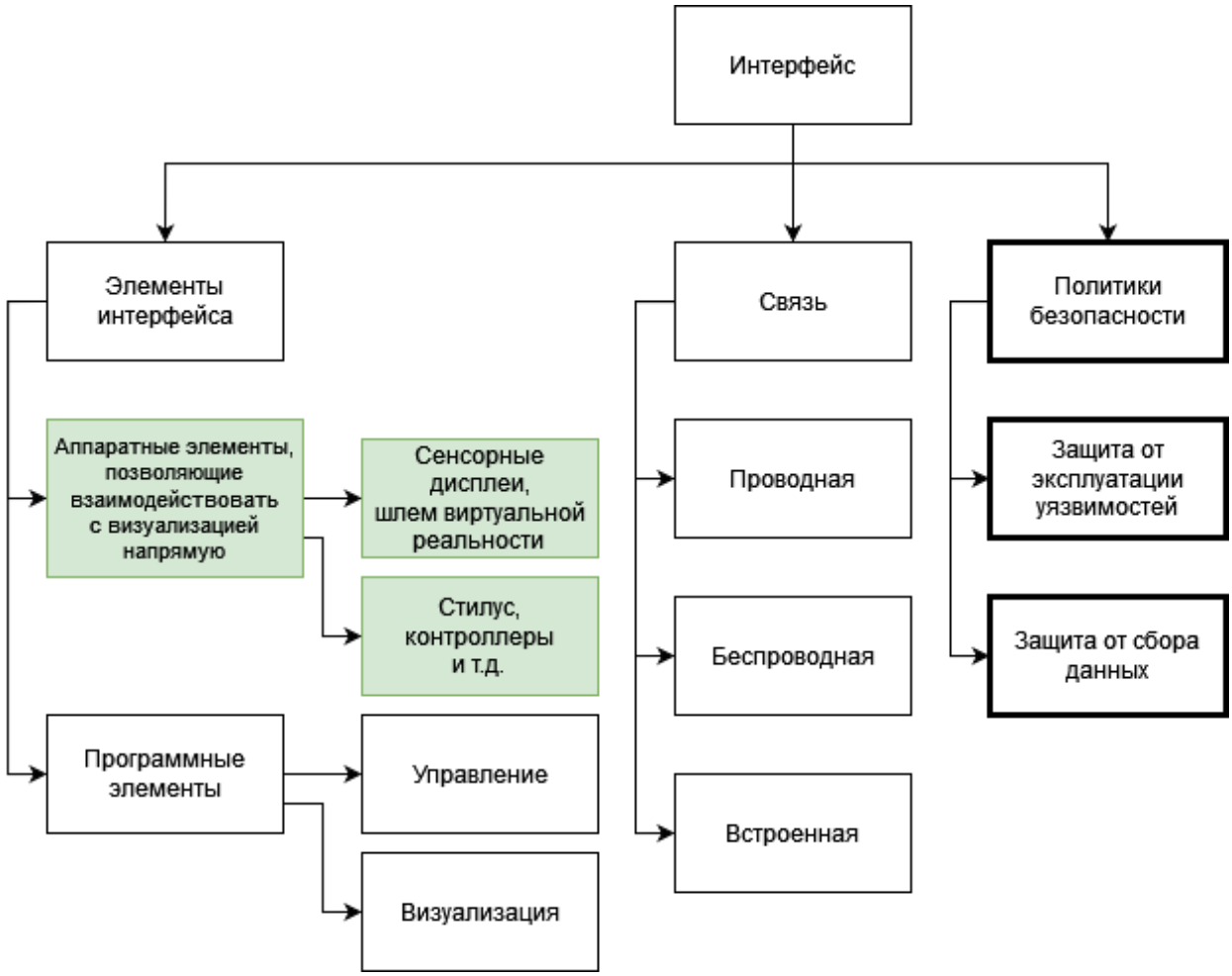


Рисунок 7 – Схема основных элементов человеко-компьютерного интерфейса

Для того чтобы представить элементы человеко-компьютерного интерфейса формально, разрабатывалась модель, которая включает в себя физические и программные элементы интерфейса, а также связи между ними (при наличии).

Структуру полученных моделей можно представить с помощью теоретико-множественного подхода. Модель интерфейса задаётся в виде кортежа $I = <E, L, S>$, где:

- E – множество элементов интерфейса;

- L – множество связей между элементами интерфейса, которые описывают возможные способы взаимодействия (проводная связь между элементами, беспроводная и встроенная, когда один элемент интерфейса встроен в другой);

- S – возможные настройки безопасности интерфейса при их наличии, описывают каждый элемент интерфейса с точки зрения его защищенности от реализации атак, использующих различные уязвимости.

При более детальном рассмотрении, элемент модели $e_i \in E$ описывает один i-ый элемент множества элементов интерфейса E и имеет следующий вид: $e_i = <elementType>$, где $elementType \in ElementTypes$ – это тип элемента.

Модель взаимосвязей между элементами интерфейса задается в виде $L = <e_i, e_{i+1}, CT>$, где CT = {*wire, wireless, embedded*} – тип соединения между элементами интерфейса, а именно: проводной (элементы интерфейса соединяются между собой с помощью провода), беспроводной (беспроводная связь: Bluetooth, инфракрасное излучение и т.п.), встроенный (один элемент интерфейса встроен в другой, например, элемент визуализации и память соединены в одном устройстве), соответственно. Используя знания о типе связей между элементами, нарушитель также может произвести атаку на интерфейс.

Модель параметров безопасности элементов интерфейса P задается следующим образом: $P = <E, V>$, где E – множество элементов интерфейса, для которых могут быть определены настройки безопасности, V – множество закрытых уязвимостей (модель уязвимостей описывается в разделе 2.2). Отсюда следует, что элемент множества настроек безопасности $p_i \in P$ имеет следующий вид: $p_i = <d, v>$, где $d \in D$ – это устройство, на котором задана настройка безопасности, $v \in V$ – уязвимость, эксплуатация которой для данного элемента интерфейса невозможна. Моделирование параметров безопасности позволяет описать такие методы защиты интерфейса, как пароли, двухфакторная аутентификация, биометрия и т.п.

2.2 Модель уязвимостей

Иностранные стандарты [113, 114, 115] под уязвимостью подразумевают слабые места в организации безопасности системы, архитектуре системы, управлении, персонале, программном и аппаратном

обеспечении, которые могут быть использованы злоумышленником для получения несанкционированного доступа к информации, для нарушения обработки данных, для причинения вреда компьютерной системе или её деятельности. Российские стандарты [116] определяют уязвимость как некое свойство компьютерной системы, которое даёт возможность реализовать угрозу безопасности информации, обрабатываемой в системе. При этом угроза определяется как совокупность явлений, действий или процессов, создающих реальную или потенциальную опасность для данных, обрабатываемых системой.

В данной работе под «уязвимостью» следует понимать слабое место в системе, а также при взаимодействии пользователя с этой системой, при использовании которого злоумышленник может нанести вред пользователю или системе. При этом уязвимость сама по себе не наносит ущерб, однако делает нанесение ущерба возможным [115]. Под угрозой в данном случае понимается совокупность действий, которые могут привести к опасности для обрабатываемых данных или для оператора. В соответствии с приведёнными выше определениями здесь и далее рассматриваются уязвимости, позволяющие осуществлять угрозы для современных типов человеко-компьютерных интерфейсов.

В настоящее время для уязвимостей интерфейсов ещё не составлено таких же баз данных, как общепризнанные базы «Распространённые уязвимости и риски» (CVE) или «Перечисление и классификация распространённых шаблонов атак» (CAPEC) [117, 118]. Однако для описания модели уязвимостей было принято решение взять за основу общепринятые стандарты, поэтому предлагаемая модель содержит ряд общепринятых параметров, а также ряд новых параметров, типичных для человеко-компьютерных интерфейсов. Таким образом, параметры уязвимостей подбирались с учётом особенностей безопасности, которыми обладают человеко-компьютерные интерфейсы.

Для того, чтобы оценить серьёзность уязвимости, необходимо понимать, что для этого нужно злоумышленнику, каким образом он должен взаимодействовать с системой, насколько подготовленным должен быть злоумышленник, и какой ущерб получают данные или пользователь. Эти параметры определяют, насколько сложно провести атаку и какой она наносит урон, т.е. позволяют судить о критичности уязвимости.

Параметры, позволяющие судить о сложности атаки, следующие.

(1) Права, которыми должен располагать злоумышленник для совершения атаки.

(2) Должен ли злоумышленник самостоятельно взаимодействовать с системой.

(3) Какой канал используется для взаимодействия, визуальный или звуковой.

(4) Уровень подготовки, которым должен обладать злоумышленник для совершения атаки.

Параметры, описывающие ущерб, представлены ниже, причём три показателя относятся к ущербу данным, и один – ущерб для физического состояния и самочувствия непосредственно оператора.

(1) Ущерб конфиденциальности.

(2) Ущерб целостности.

(3) Ущерб доступности.

(4) Ущерб физическому состоянию оператора.

Таким образом, получены 8 показателей: (1) требуемые привилегии, (2) сложность эксплуатации уязвимости, (3) взаимодействие, (4) канал восприятия; (5) конфиденциальность, (6) целостность, (7) доступность, (8) физическое состояние оператора.

Среди полученных показателей остаются неизменными по отношению к системе CVSS *«требуемые привилегии»* (какие права нужны злоумышленнику для проведения атаки), *«сложность эксплуатации уязвимости»* (какой уровень подготовленности нужен злоумышленнику для

совершения атаки), а также показатели ущерба *конфиденциальность*, *целостность* и *доступность*. Параметр *«взаимодействие»* претерпел изменения по сравнению со своим общепринятым значением и показывает, кто именно должен взаимодействовать с интерфейсом: оператор или злоумышленник. Также было принято решение исключить параметр *«вектор атаки»*, так как в случае человеко-компьютерного интерфейса этот параметр всегда будет иметь значение «физический» (physical). Однако был добавлен один параметр ущерба *«урон оператору»*, который характеризует степень тяжести ущерба, наносимого физическому состоянию человека, работающего с интерфейсом. Также был добавлен параметр *«канал восприятия»*, который характеризует пути проведения атаки для интерфейса (визуальный или звуковой).

Таким образом, был добавлен новый параметр ущерба, добавлен параметр используемого для атаки канала восприятия, исключён параметр «вектор атаки», изменён параметр «взаимодействие», остальные параметры были адаптированы с учётом особенностей безопасности человеко-компьютерных интерфейсов. Рассмотрим базовые показатели уязвимостей более подробно.

1. Требуемые привилегии – Privileges Required (PR). Этот показатель действует в случае, если интерфейс защищён. Например, защищён паролем или другими алгоритмами аутентификации.

$$PrivilegesRequired(v) = \begin{cases} None, \\ Low, \\ High. \end{cases}$$

• *Нет (None).* Данное значение параметра выставляется, если интерфейс не защищён.

• *Низкий (Low).* Данное значение параметра выставляется, если злоумышленник располагает привилегиями такими же, как и у других пользователей.

• *Высокий (High).* Для того, чтобы воспользоваться уязвимостью, требуются административные привилегии для доступа к уязвимому компоненту.

2. Сложность эксплуатации уязвимости – Complexity (C). Параметр описывает, насколько сложно воспользоваться уязвимостью. Данный параметр не зависит от злоумышленника.

$$Complexity(v) = \begin{cases} Low, \\ High. \end{cases}$$

• *Низкая (Low).* Не требуется каких-либо особенных условий для того, чтобы воспользоваться уязвимостью и произвести атаку на интерфейс.

• *Высокая (High).* Для того, чтобы воспользоваться уязвимостью, требуется некоторая подготовка, а эксплуатация уязвимости может требовать ресурсных и временных затрат.

3. Вовлечённость пользователя – User Interaction (UI). Параметр описывает, требуется ли участие самого пользователя, чтобы злоумышленник смог воспользоваться уязвимостью.

$$Criticality(v) = \begin{cases} Operator, \\ Attacker. \end{cases}$$

• *Злоумышленник (Attacker).* Злоумышленник может воспользоваться уязвимостью без участия пользователя.

• *Оператор (Operator).* Чтобы злоумышленник мог воспользоваться уязвимостью, пользователь должен совершать какие-либо действия с интерфейсом.

4. Канал восприятия – Perception Channel (CH). Параметр, отображающий канал, по которому проводится атака – визуальный или звуковой.

$$Channel(v) = \begin{cases} Video, \\ Audio. \end{cases}$$

• *Визуальный (Visual)* [119]. Злоумышленнику требуется увидеть какие-то данные, с которыми взаимодействует пользователь (если *UI = Operator*). Злоумышленник должен производить какие-либо действия с камерой устройства, например, показать изображение лица пользователя, чтобы обойти биометрическую аутентификацию (используется для таких атак, как спуфинг камеры) (если *UI = Attacker*).

• *Звуковой (Audial)* [120]. Злоумышленнику требуется услышать звуковой сигнал от интерфейса (если *UI = Operator*). Злоумышленник должен передавать какой-то сигнал аудиоприёмнику интерфейса. Например, передача команд в ультразвуковом диапазоне, когда приёмник слышит команду, а сторонний наблюдатель – нет (если *UI = Attacker*).

5. Урон оператору – Physical Damage (PD). Показатель, специфичный для человеко-компьютерных интерфейсов, отображающий, насколько серьёзный урон получил оператор в результате атаки на интерфейс.

$$PhysicalDamage(v) = \begin{cases} None, \\ Low, \\ High. \end{cases}$$

• *Нет (None).* Атака с применением данной уязвимости не вызывает физического урона оператору, то есть, не воздействует на его органы чувств, психику и/или восприятие.

• *Низкий (Low).* Атака вызывает незначительный физический урон вроде повышения усталости у оператора (например, смена частоты мерцания

монитора, незаметная невооружённым глазом, но утомляющая при работе с этим монитором).

• *Высокий (High).* Атака вызывает сильный физический урон вроде возникновения ощущения тошноты, головокружения, потери ориентации в пространстве, приступов и т.п.

6. Конфиденциальность – Confidentiality (C). Показатель позволяет оценить ущерб конфиденциальности данных пользователя, который наносит используемая уязвимость.

$$Confidentiality(v) = \begin{cases} Low, \\ Medium, \\ High. \end{cases}$$

• *Нет (None).* Ущерба конфиденциальности не происходит.

• *Низкий (Low).* Низкий уровень ущерба конфиденциальности, злоумышленник получает доступ к ограниченному объёму информации.

• *Высокий (High).* Высокий уровень ущерба конфиденциальности, злоумышленник получает полный доступ к ресурсам затронутого компонента интерфейса или полученная злоумышленником информация обладает большой важностью.

7. Целостность – Integrity (I). Показатель позволяет оценить ущерб целостности данных пользователя, который наносит используемая уязвимость.

$$Integrity(v) = \begin{cases} Low, \\ Medium, \\ High. \end{cases}$$

• *Нет (None).* Ущерба целостности не происходит.

• *Низкий (Low).* Низкий уровень ущерба целостности, данные изменены незначительно, и эти изменения не оказывают серьёзного влияния на компонент интерфейса.

• *Высокий (High).* Высокий уровень ущерба целостности. Злоумышленник может изменить любые данные, касающиеся данного компонента, либо модификация этих данных злоумышленником будет иметь серьёзные последствия.

8. Доступность – Availability (A). Показатель позволяет оценить ущерб доступности данных пользователя, который наносит используемая уязвимость.

$$Availability(v) = \begin{cases} Low, \\ Medium, \\ High. \end{cases}$$

• *Нет (None).* Ущерба доступности не происходит.

• *Низкий (Low).* Низкий уровень ущерба доступности, происходит снижение производительности или частичный/временный отказ в обслуживании, однако злоумышленник не имеет возможности полностью лишить доступа к данным.

• *Высокий (High).* Высокий уровень ущерба доступности, который означает полную потерю доступа к ресурсам затронутого компонента, либо серьёзные последствия при потере доступности к данным затронутого компонента.

9. Сфера действия уязвимости – Scope (S). В качестве дополнительного параметра, влияющего на критичность уязвимости, принято решение оставить параметр «сфера действия уязвимости». Параметр характеризует степень распространения уязвимости на другие компоненты интерфейса, оказано ли влияние на другие компоненты, кроме уязвимого. Значение *Scope* зависит от количества компонентов, затронутых одной уязвимостью.

$$Scope = \begin{cases} Unchanged, \\ Changed. \end{cases}$$

• *Без изменений (Unchanged).* Использование уязвимости влияет только на уязвимый компонент.

• *Изменённая (Changed).* Использование уязвимости затрагивает не только уязвимый компонент, а влияет также и на другие компоненты интерфейса. Тогда ущерб конфиденциальности, целостности и доступности оценивается для затронутых компонентов.

Данные параметры сгруппированы в таблице 5.

Таблица 5 – Значения параметров уязвимостей

Показатель	**Значение**
Сложность эксплуатации уязвимости	Low
	High
Требуемые привилегии	None
	Low
	High
Взаимодействие	None
	Required
Канал восприятия	Audial
	Visual
Конфиденциальность/ Целостность/ Доступность/ Урон оператору	None
	Low
	High

Модель угроз, в соответствии с полученными параметрами, выглядит следующим образом: v_i =<*pd, ec, pr, ui, ch, c, i, a*>, где *pd* – урон оператору, *ec* – сложность использования уязвимости, *pr* – требуемые привилегии, *ui* – взаимодействие, *ch* – канал восприятия, *c* – риск для конфиденциальности, *i* – риск для целостности, *a* – риск для доступности.

С использованием этой модели оценивание уязвимости осуществляется экспертами при заполнении таблицы, пример которой дан ниже.

Таблица 6 – Пример заполнения таблицы экспертами

Показатель	**Значение, балл**	**Экс.1**	**Экс.2**	**…**	**Экс.N**
Сложность эксплуатации уязвимости	Низкая, 1	Низкая	Низкая	…	Высокая
	Высокая, 2				

Показатель	Значение, балл	Экс.1	Экс.2	…	Экс.N
Требуемые привилегии	Нет, 0	Нет	Низкие	…	Нет
	Низкие, 1				
	Высокие, 2				
Взаимодействие	Оператор, 1	Оператор	Оператор	…	Оператор
	Злоумышленник, 2				
Канал восприятия	Аудио, 1	Видео	Видео	…	Видео
	Видео, 2				
Конфиденциальность/ Целостность/ Доступность/ Урон оператору	Нет ущерба, 0	Низкий	Нет ущерба	…	Низкий
	Низкий, 1				
	Высокий, 2				

Каждому категориальному значению показателя присваивается целочисленный балл. Баллы значений, выбранных экспертами, суммируются, а затем сумма делится на количество экспертов [147]. В случае если получено дробное число, результат округляется до ближайшего целого. Пусть уязвимость оценивают 3 эксперта. Сложность эксплуатации уязвимости они оценили как «Низкая» (1 балл), «Низкая» (1 балл) и «Высокая» (2 балла). Полученная оценка показателя равна (1+1+2)/3=1,33. В результате округления до ближайшего целого даёт 1. Таким образом, сложность эксплуатации уязвимости следует оценить как низкую.

2.3 Алгоритм оценивания защищённости интерфейса

Для того чтобы оценить защищённость системы человеко-компьютерного взаимодействия, используются алгоритмы оценивания каждой выявленной уязвимости в отдельности и алгоритмы оценивания защищённости интерфейса, учитывающие все выявленные уязвимости.

Основными параметрами объекта, по которым можно судить о том, насколько успешна разработка объекта, являются качество, эффективность и результативность. Данный раздел посвящён оцениванию интерфейса. Чтобы оценить какой-либо объект, требуется измерить качество и эффективность работы этого объекта.

Чтобы разработать какой-либо объект или модель, перед разработкой необходимо определить ряд параметров:

(1) с какой целью создаётся объект или модель,

(2) какие у него основные характеристики,

(3) какие из этих характеристик являются важными для изучения,

Чтобы оценить полученный объект или модель, требуется:

(1) описать его в соответствии с его имеющимися параметрами,

(2) оценить его соответствие основным свойствам, которыми он должен обладать.

В соответствии с ISO 8402-2000, *качество* – это набор характеристик объекта, по которым можно определить, способен ли объект выполнять заданные функции и удовлетворять потребностям пользователей, при этом потребности могут быть как реальные, так и предполагаемые [124, 125, 126, 127, 128]. Под объектом может пониматься как материальная продукция, так и какая-либо услуга [128].

Поскольку в данной работе рассматриваются модели интерфейсов, следует также привести определение оценивания качества модели. Согласно отечественным и международным стандартам [127, 129, 130], под качеством модели понимается свойство или набор свойств какой-либо модели, которые позволяют назвать её пригодной для использования по назначению. Иными словами, качество модели, также как и качество объекта, определяется как способность выполнять требования, поставленные пользователем.

В качестве основных свойств моделей могут быть выделены следующие понятия [131]:

- адекватность, т.е. показатель того, насколько модель соответствует оригиналу;

- простота, может оцениваться через другой показатель, сложность модели, т.е., например, насколько сложную структуру имеет модель;

- оптимальность;

- гибкость, т.е. насколько легко можно доработать модель в случае необходимости;

- адаптивность, т.е. насколько легко можно приспособить модель к меняющимся условиям;

- развиваемость – более общее понятие, включающее в себя «гибкость» и «адаптивность»;

- интеллектуальность – может по-разному трактоваться, однако в информационных технологиях интеллектуальными часто называют такие модели, которые используют какие-либо математические алгоритмы, повышающие адаптивность модели, например, статистические алгоритмы или алгоритмы машинного обучения;

- универсальность, т.е. насколько данная модель подходит, например, разным областям человеческой деятельности;

- проблемная ориентация, т.е. ориентированность модели на решение какой-то определённой проблемы или задачи;

- надёжность, т.е. способность модели выполнять её функции без отказа в определённый временной срок;

- эффективность машинной реализации – т.е., насколько хорошо выполняет свои функции машинная реализация разработанной модели;

- устойчивость, т.е. возможность с помощью данной модели в тех же условиях повторить полученный ранее результат;

- инвариантность, т.е. неизменность модели при изменении условий;

- наблюдаемость, т.е. возможность вычислить начальное состояние системы по значению на её выходе [132];

- управляемость – возможность управлять состоянием системы, приводя её из начального состояния в требуемое состояние [132];

- чувствительность – т.е. влияние изменения входных параметров на выходные характеристики;

- и др.

По признаку наличия/отсутствия этих свойств можно судить о качестве модели [133]. Однако, в зависимости от задачи, которую должна выполнять разрабатываемая модель, ключевыми могут оказаться разные свойства. Например, модель может быть простой и ориентированной на решение какой-либо конкретной проблемы, однако при этом не быть адаптивной, так как рассчитана на короткий срок. Поскольку человеко-компьютерные интерфейсы, рассматриваемые в данной работе, предназначены для приложений информационной и компьютерной безопасности, они должны обладать тремя основными свойствами: (1) защищённость, (2) скорость и (3) точность. Эти три свойства можно определить следующим образом.

Защищённость – измеряется через уровень критичности уязвимостей, содержащихся в интерфейсе, при этом уровень критичности должен быть низким.

Скорость – время, за которое оператор справляется с поставленной задачей.

Точность – допустимое количество ошибок оператора.

Таким образом, в данном разделе представлен алгоритм получения общей оценки интерфейса с применением алгоритма оценивания уязвимостей каждого из элементов этого интерфейса, выработанного в разделе 2.3. При общей оценке рассматривается не только параметр *защищённости*, но также параметры *точности* и *скорости* принятия решений оператором как параметры удобства использования. Таким образом, общая оценка эффективности интерфейса складывается из оценки защищённости принятия решений и оценки удобства использования в смысле оперативности работы с интерфейсом.

Оценивание защищённости интерфейса проводится на основании тех показателей, которые были использованы для оценивания уязвимостей. В данном случае предполагается оценивание уязвимости всей системы человеко-компьютерного взаимодействия на основании знания обо всех присутствующих в этой системе уязвимостях. Таким образом, происходит

суммирование значений всех уязвимостей системы для оценки общего состояния безопасности системы.

В наиболее общем виде алгоритм оценки защищённости интерфейса будет состоять из следующих шагов (блок-схема на рисунке 8).

(1) На вход алгоритма поступает набор элементов интерфейса и их уязвимостей.

(2) Производится расчёт уровня уязвимости для каждого уязвимого элемента.

(3) Вводятся значения уровней уязвимости для каждого элемента.

(4) Рассчитывается суммарный показатель защищённости интерфейса.

(5) На выход алгоритма поступает значение суммарного показателя защищённости интерфейса.

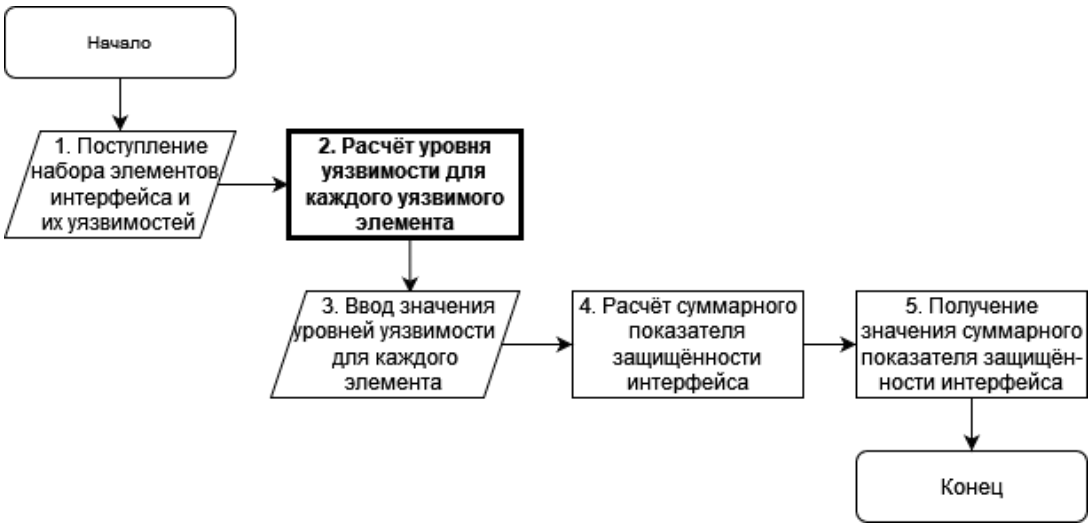


Рисунок 8 – Блок-схема алгоритма оценки интерфейса

Таким образом, вначале на основе значений параметров каждого элемента (см. раздел 2.1), оператор определяет, какие элементы интерфейса могут быть уязвимы.

Далее оператор, располагая знаниями об уязвимостях элементов интерфейсов, на основе показателей уязвимостей, определённых в разделе 2.2, рассчитывает уровень уязвимости для каждого уязвимого элемента. Расчёт уровня каждой уязвимости происходит по алгоритму, описанному в

разделе 2.3.1. Таким образом, выявляется уровень критичности каждой уязвимости.

Затем оператор, используя полученные значения уровней уязвимости каждого элемента, рассчитывает суммарный показатель защищённости интерфейса (раздел 2.3.2.).

2.3.1 Расчёт уровня уязвимости элемента интерфейса

В разделе 2.2 было произведено ранжирование уязвимостей по группам [122]. Получилось 8 показателей уязвимости:

(1) урон оператору, который имеет 3 возможных значения;

(2) сложность эксплуатации уязвимости, которая имеет 2 возможных значения;

(3) требуемые привилегии с 3 возможными значениями;

(4) взаимодействие с 2 возможными значениями;

(5) канал восприятия с 2 возможными значениями;

(6) ущерб конфиденциальности с 3 возможными значениями;

(7) ущерб целостности с 3 возможными значениями;

(8) ущерб доступности с 3 возможными значениями.

Показатели были разделены на две категории:

(1) Влияние на ущерб данным (сюда вошли показатели конфиденциальности, целостности и доступности) и урон оператору;

(2) Возможность эксплуатации уязвимости (в эту категорию вошли сложность эксплуатации уязвимости, требуемые привилегии, взаимодействие и канал восприятия);

Для расчёта уровня уязвимости (шаг 2) были определены числовые значения показателей уязвимостей, которые представлены в таблице 7. Значения параметров определены на основании анализа соответствующих работ [122, 123] и опроса экспертов по информационной безопасности. Были собраны 7 экспертных мнений о том, какое значение должно быть у каждого показателя уязвимостей интерфейсов. Количество опрашиваемых экспертов основывалось на опыте найденных исследований в области разработки

человеко-компьютерных интерфейсов и в области компьютерной безопасности. Так, исследования [136, 137] говорят, что для того, чтобы оценить интерфейс, достаточно 5 экспертов, так как они видят 85% проблем. При этом исследования оценивания систем принятия решений говорят, что для данной задачи достаточно 10 экспертов [138]. Однако, в работе [139] показано, что слишком большое количество экспертов для компьютерной безопасности ухудшает показатели оцениваемой системы безопасности. Вместе с тем для оценивания какой-либо системы может учитываться уровень компетентности эксперта, например, по шкале от 1 (очень низкий уровень) до 5 (очень высокий уровень) [138]. В случае предлагаемой системы, оценивавшие её эксперты имели уровень компетентности 4 (высокий уровень) и 5 (очень высокий уровень), так как были кандидатами и докторами наук соответственно. Учитывая анализ вышеприведённых исследований, количество экспертов от 5 до 10 является достаточным.

Таблица 7 – Значения параметров уязвимостей

Показатель	**Значение**	**Числовое значение**
Сложность эксплуатации уязвимости	Низкая	0.65
	Высокая	0.42
Требуемые привилегии	Нет	0.86
	Низкие	0.72 (0.78, если сфера действия уязвимости поменялась)
	Высокие	0.67 (0.6, если сфера действия уязвимости поменялась)
Аудио-канал восприятия	Взаимодействие – оператор	0.72
	Взаимодействие – атакующий	0.68
Видео-канал восприятия	Взаимодействие – оператор	0.73
	Взаимодействие – атакующий	0.69
Конфиденциальность/ Целостность/ Доступность/ Урон оператору	Нет ущерба	0
	Низкий	0.12
	Высокий	0.52

Подбор параметров осуществлялся в соответствии с набором следующих принципов.

1. Значение оценки должно находиться в пределах от 0 до 10 включительно. Диапазон значений – [0, 10].

2. Оценки различных типов уязвимостей должны количественно отличаться друг от друга.

3. Группы уязвимостей с разными уровнями критичности должны подчиняться закону нормального распределения.

Используя эти показатели, был проведён эксперимент оценки 336 уязвимостей, являющихся различными комбинациями данных показателей. Возможные 14 комбинаций ущерба, дающие разную оценку, были следующие:

1) 1 показатель ущерба со значением «низкий»;

2) 2 показателя ущерба со значением «низкий»;

3) 3 показателя ущерба со значением «низкий»;

4) 4 показателя ущерба со значением «низкий»;

5) 1 показатель ущерба со значением «высокий»;

6) 2 показателя ущерба со значением «высокий»;

7) 3 показателя ущерба со значением «высокий»;

8) 4 показателя ущерба со значением «высокий»;

9) 1 показатель ущерба со значением «высокий» и 1 показатель ущерба со значением «низкий»;

10) 1 показатель ущерба со значением «высокий» и 2 показателя ущерба со значением «низкий»;

11) 1 показатель ущерба со значением «высокий» и 3 показателя ущерба со значением «низкий»;

12) 2 показателя ущерба со значением «высокий» и 1 показателя ущерба со значением «низкий»;

13) 2 показателя ущерба со значением «высокий» и 2 показателя ущерба со значением «низкий»;

14) 3 показателя ущерба со значением «высокий» и 1 показатель ущерба со значением «низкий».

Поскольку есть 2 возможных значения сложности эксплуатации уязвимости, 3 возможных значения показателя требуемых привилегий, 2 учитываемых канала передачи и 2 возможных типа взаимодействия (взаимодействует атакующий или оператор), комбинаций получилось всего 336. При оценке данных комбинаций с помощью приведённого выше алгоритма распределение уровня критичности оказалось следующим (рисунок 11).

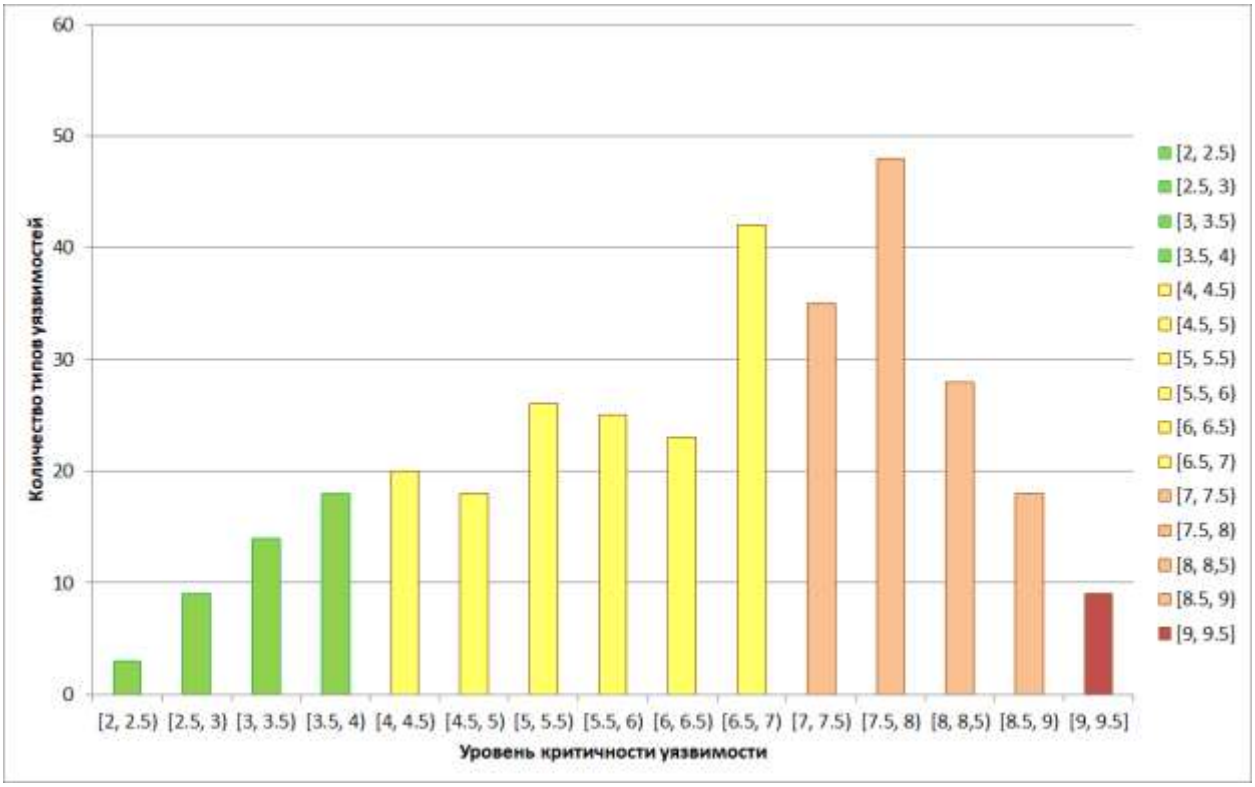


Рисунок 11 – Распределение уровня критичности оцениваемых уязвимостей

На рисунке видно, что среди оцениваемых уязвимостей присутствуют уязвимости различного уровня критичности. При этом наибольшим оказалось количество уязвимостей среднего уровня (столбцы выделены жёлтым цветом), наименьшим – количество уязвимостей низкого (зелёные столбцы) и критического (красный столбец) уровня. Уязвимости высокого

уровня критичности представлены оранжевыми столбцами. Таким образом, распределение уровней критичности уязвимостей близко к нормальному.

На рисунке 12 видно распределение уязвимостей базы NVD. Отсюда видно, что полученное распределение не хуже, чем у существующей базы данных уязвимостей.

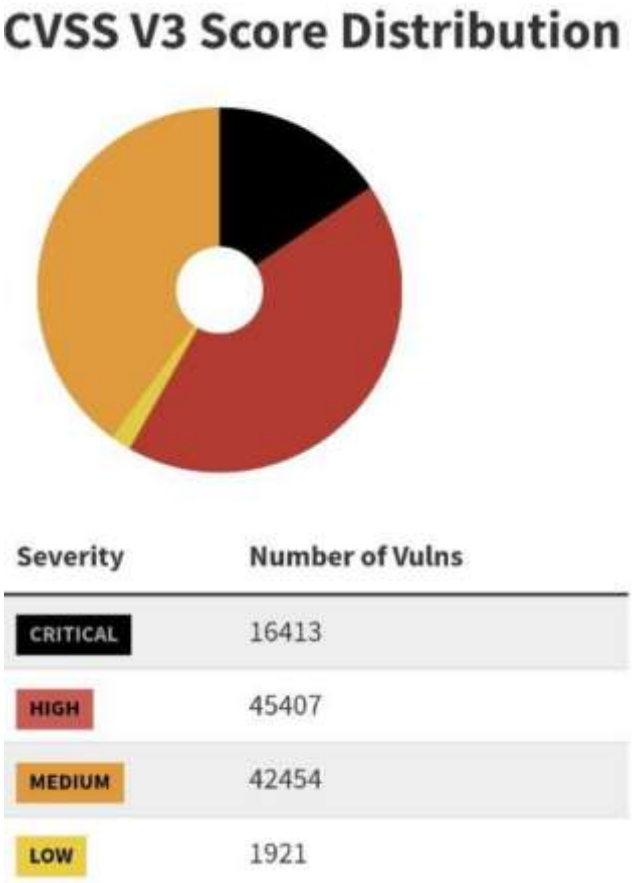


Рисунок 12 – Распределение уровня критичности уязвимостей (NVD) [146]

Ниже более подробно рассматривается второй шаг алгоритма оценивания защищённости интерфейса, расчёт уровня уязвимости элемента интерфейса, использующий показатели уязвимости, определённые и описанные в разделе 2.2. Набор показателей уязвимости включает в себя показатели возможного ущерба для данных от применения уязвимости (ущерб конфиденциальности, целостности или доступности), показатель урона оператору (описывающий ущерб физическому состоянию оператора) и показатели, характеризующие сложность действий атакующего (сложность

эксплуатации и уровень подготовленности, требуемые привилегии, взаимодействие и канал восприятия).

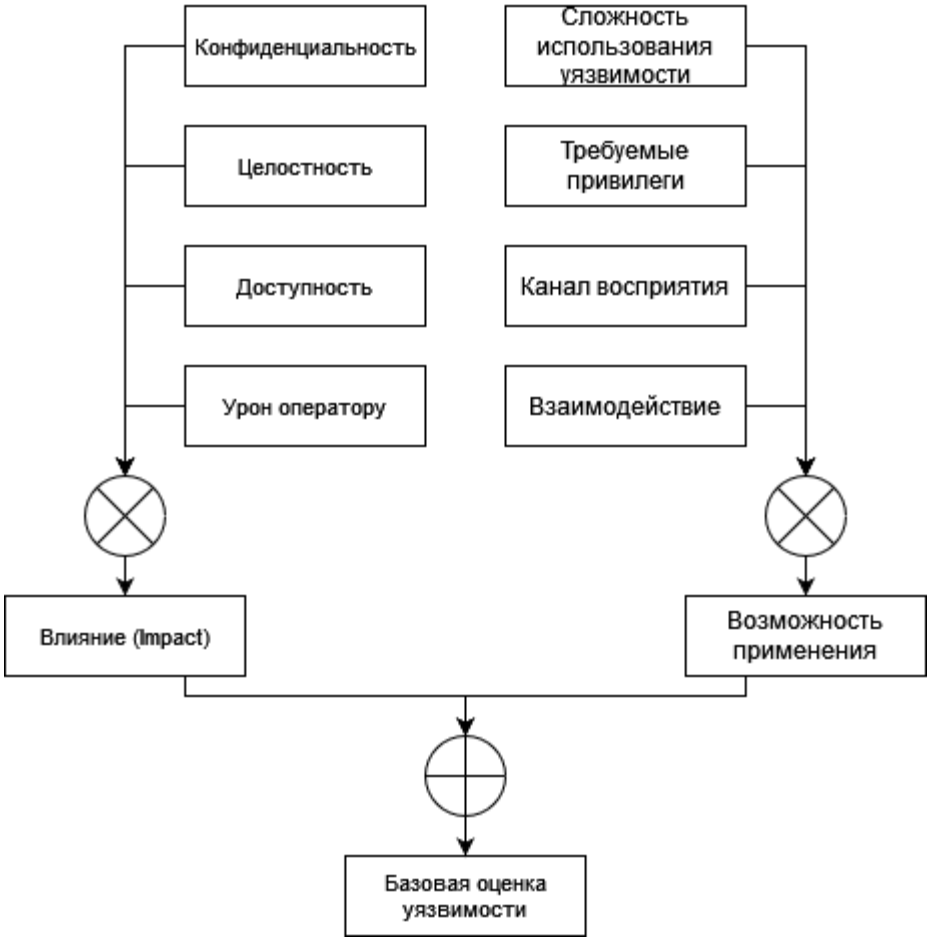


Рисунок 9 – Блок-схема алгоритма расчёта влияния применения уязвимости на систему

Показатели, входящие в одну группу, перемножаются между собой, затем полученные промежуточные значения складываются, как показано на рисунке 9. Символами $\otimes$ и $\oplus$ обозначаются математические преобразования: $\otimes$ обозначает преобразования с участием перемножения, $\oplus$ обозначает операции с участием сложения.

Алгоритм оценивания уязвимости формально можно представить в виде математических выражений, за основу которых взяты формулы, используемые в системе CVSS, так как она обладает открытой структурой, содержащей формулы и показатели. В формулы были внесены изменения в

соответствии с предлагаемой моделью уязвимостей (см. раздел 2.2). Для оценки уязвимостей выполняются следующие шаги.

1. Вначале оценивается уровень ущерба для конфиденциальности, целостности и доступности данных, а также, в отличие от исходной системы оценки CVSS, учитывается урон, наносимый оператору. В случае человеко-компьютерного интерфейса эти показатели будут одинаково важны. Также оцениваются сложность использования уязвимости, требуемые привилегии, параметр взаимодействия и параметр «канал восприятия». Полученные значения вводятся в систему оценивания защищённости интерфейса.

2. Далее производится расчёт уровня каждой уязвимости.

2.1. Формула расчёта суммарного ущерба выглядит следующим образом:

$$preImpact = 1 - [(1 - C) \times (1 - I) \times (1 - A) \times (1 - D)]. \quad (5)$$

где $preImpact$ – предварительный ущерб данным и оператору, C – ущерб конфиденциальности, I – ущерб целостности, A – ущерб доступности, D – урон, наносимый оператору.

Влияние ущерба *Imp* в случае, если сфера действия уязвимости – только уязвимый компонент (*Scope* = *Unchanged*):

$$Imp = preImpact \times 6{,}42. \quad (6)$$

Влияние ущерба в случае, если сфера действия уязвимости может распространяться на другие компоненты интерфейса, кроме уязвимого:

$$Imp = 7.52 \times (preImpact - 0.029) - 3.25 \times (preImpact - 0.02)^{15}. \quad (7)$$

2.2. Возможность применения уязвимости, в отличие от исходной системы оценки (CVSS) включает изменённый параметр «взаимодействие» и новый параметр «канал восприятия»:

$$Exp = 8.22 \times EC \times PR \times UI \times CH, \quad (8)$$

где EC – сложность использования уязвимости (с двумя возможными значениями), PR – требуемые привилегии (с тремя

возможными значениями), *UI* – параметр взаимодействия (с двумя возможными значениями), *CH* – параметр «канал восприятия» (с двумя возможными значениями).

2.3. Общая оценка уязвимости рассчитывается по формуле 11. Формула расчёта общей оценки уязвимостей оставлена без изменений, поскольку в целях объединения исходной системы с предлагаемой оценка ущерба должна сохранять те же принципы, что и в CVSS. Таким образом, общая оценка уязвимости представлена ниже:

$$BS_k = \begin{cases} 0, Imp = 0 \\ \min[(Imp + Exp), 10], Scope = Unchanged \\ \min[1{,}08 \times (Imp + Exp), 10], Scope = Changed \end{cases},$$

где BS_k – базовая оценка уровня критичности угрозы для k-ого компонента интерфейса. При этом распределение уровня критичности уязвимости в соответствии с полученными значениями следующее [121]:

$$Criticality(v) = \begin{cases} Low, \ BS_k(v) \in (0; 4), \\ Medium, \ BS_k(v) \in [4; 7), \\ High, \ BS_k(v) \in [7; 9), \\ Critical, \ BS_k(v) \in [9; 10]. \end{cases}$$

Из (10) следует, что, в зависимости от значения базовой оценки, уязвимость может иметь 4 уровня: (1) до 4 баллов – низкий уровень; (2) от 4 баллов до 7 баллов – средний уровень; (3) от 7 баллов до 9 баллов – высокий уровень; (4) от 9 баллов до 10 баллов включительно – критический уровень. Показатель критичности (*Criticality(a,v)*) не является точной количественной мерой и нужен для того, чтобы оценить серьёзность одной уязвимости относительно другой, а не точно измерить.

Шаги второго этапа алгоритма, описанные выше, представлены в виде блок-схемы на рисунке 10.

2.3.2 Расчёт общего уровня защищённости интерфейса

В данном разделе более подробно рассматривается четвёртый шаг алгоритма оценивания защищённости интерфейса.

3. Вводятся полученные значения уязвимостей каждого элемента интерфейса $BS_{in}(I_n)$.

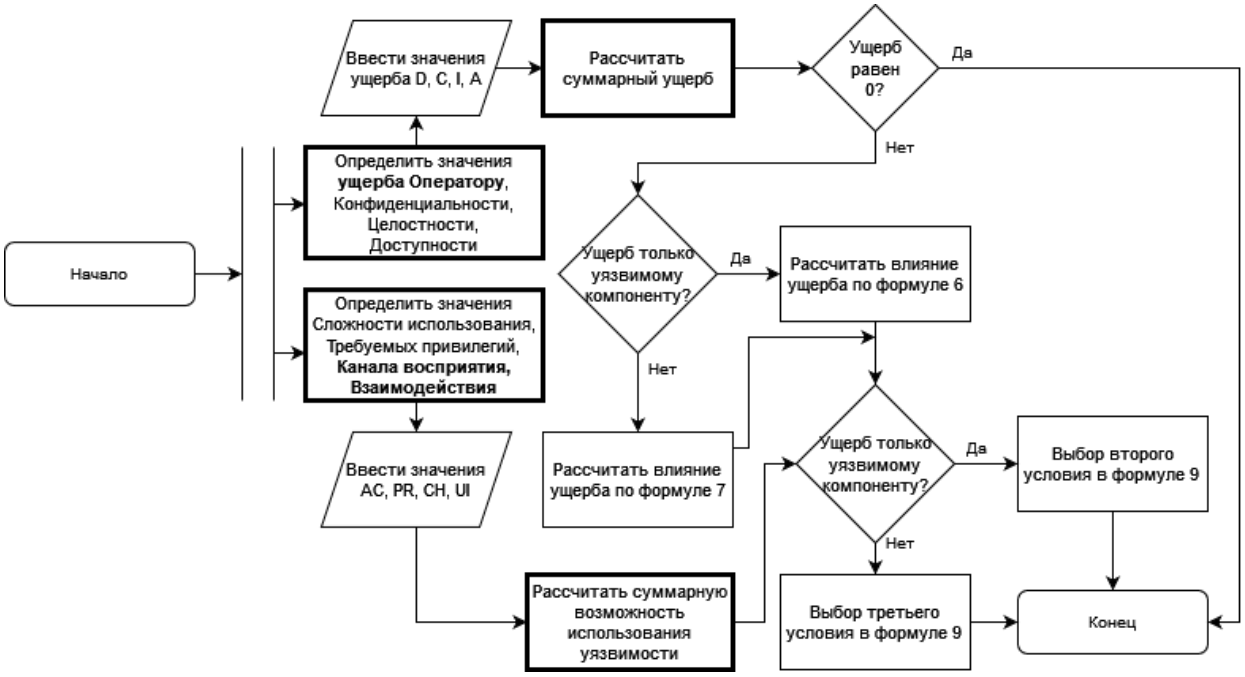


Рисунок 10 – Блок-схема второго этапа алгоритма оценивания защищённости интерфейса

4. Процесс расчёта суммарного показателя уязвимости интерфейса (*Vulnerability(i)*) (с учётом коэффициентов важности групп уязвимостей) можно представить в виде (11):

$$BS_{\sum}(I_n) = 0{,}01 \sum_{i=1}^{VULN_{n1}} BS_{in}(I_n)(Low) + 1 \sum_{i=1}^{VULN_{n2}} BS_{in}(I_n)(Medium) + 124{,}5 \sum_{i=1}^{VULN_{n3}} BS_{in}(I_n)(High) + 13\,969 \sum_{i=1}^{VULN_{n4}} BS_{in}(I_n)(Critical), \quad (11)$$

где $BS_{in}(I_n)$ – полученные на втором шаге значения уязвимостей каждого элемента интерфейса, $VULN_n = VULN_{n1} + VULN_{n2} + VULN_{n3} + VULN_{n4}$ – число уязвимостей низкого, среднего, высокого и критического уровней соответственно. Коэффициенты 0,01, 1, 124,5, 13 969 были подобраны таким образом, чтобы значения различных уровней уязвимостей не пересекались, и можно было однозначно определить, что уязвимость такого уровня существует. Таким образом, значения показателя уязвимости интерфейса следующие:

$$Vulnerability(i) = \begin{cases} None, BS_{in}(I_n) = 0, \\ Low,\ 0 < BS_{in}(I_n) \leq 1{,}5, \\ Medium, 1{,}5 < BS_{in}(I_n) \leq 870{,}3, \\ High, 870{,}3 < BS_{in}(I_n) \leq 125\ 720{,}1, \\ Critical, BS_{in}(I_n) > 125\ 720{,}1. \end{cases}$$

где i – i-ая уязвимость n-ого интерфейса, $BS_{in}(I_n)$ – базовая оценка уровня критичности уязвимости n-ого интерфейса I_n.

5. Тогда расчёт уровня защищённости интерфейса ($Security(I_n)$) производится в соответствии со следующим выражением:

$$Security(I_n) = \begin{cases} Unsafe, Vulnerability(I_n) = Critical, \\ Low, Vulnerability(I_n) = High, \\ Medium, Vulnerability(I_n) = Medium, \\ High, Vulnerability(I_n) = Low, \\ Secure, Vulnerability(I_n) = None. \end{cases}$$

6. Среди оцениваемых интерфейсов выбирается тот, который обладает наибольшим уровнем защищённости.

2.5 Выводы по главе 2

1. Были разработаны две модели человеко-компьютерного интерфейса: концептуальная модель и модель на основе теории множеств. В концептуальной модели описываются потоки данных в кольце взаимодействия между оператором и интерфейсом. Данная модель отличается тем, что она учитывает когнитивный аппарат оператора, а также является общей и подходит для всех типов визуальных человеко-

компьютерных интерфейсов, в том числе для современных типов, основанных на технологиях сенсорных экранов и виртуальной реальности. Теоретико-множественная модель имеет два уровня: уровень архитектуры интерфейса и уровень описания отдельных характеристик компонентов интерфейса. Данная модель отличается тем, что учитывает параметры уязвимостей, характерные для компонентов человеко-компьютерного интерфейса. Кроме того, разработана модель угроз для человеко-компьютерных интерфейсов. Особенностью данной модели является использование существующих стандартов описания уязвимостей и шаблонов атак, однако при этом содержит параметры, характерные для человеко-компьютерных интерфейсов и не характерные для сетевой безопасности. Использование стандартов позволяет применять полученные модели совместно с моделями компьютерной сети для обеспечения комплексной оценки безопасности компьютерной системы, которая учитывала бы как сетевую безопасность, так и безопасность человеко-компьютерного интерфейса.

2. Предложенные алгоритмы оценивания уязвимостей и оценивания защищённости человеко-компьютерных интерфейсов отличаются тем, что результат их применения направлен на повышение осведомленности оператора и разработчика об уровне защищённости человеко-компьютерного интерфейса. Данные алгоритмы отличаются тем, что используют параметры, характерные именно для человеко-компьютерных интерфейсов, однако основаны на существующих стандартах, поэтому могут быть использованы для комплексной оценки безопасности компьютерной системы. Также разработан алгоритм оценивания удобства использования человеко-компьютерного интерфейса, направленный на то, чтобы оценить, насколько изменилось удобство использования интерфейса после применения к нему контрмер по обеспечению безопасности интерфейса.

ГЛАВА 3 МЕТОДИКА ОЦЕНИВАНИЯ ЗАЩИЩЕННОСТИ ЧЕЛОВЕКО-КОМПЬЮТЕРНОГО ИНТЕРФЕЙСА И ЕЁ ЭКСПЕРИМЕНТАЛЬНАЯ ОЦЕНКА

3.1 Методика оценивания человеко-компьютерных интерфейсов

Существует множество этапов жизненного цикла сложной системы, планирование, проектирование, пилотный запуск, эксплуатация, утилизация. Основными полагаем проектирование и эксплуатацию. Соответственно, оценка защищённости может проводиться на этапе проектирования, чтобы избежать появления уязвимостей, и на этапе эксплуатации, чтобы уже существующие уязвимости закрыть. Данная методика может быть применена как на обоих основных этапах существования интерфейса: проектирование и эксплуатация.

Методика оценивания человеко-компьютерного интерфейса включает в себя две составляющие: (1) оценивание защищённости и (2) оценивание удобства использования человеко-компьютерного интерфейса. Предлагаемая методика обладает следующими этапами:

1. Сбор информации. Информация собирается двумя путями: (1) с помощью модулей выбора уязвимостей и угроз, уже известных системе оценки, (2) ручной ввод оператором параметров уязвимостей и угроз, ещё не известных системе оценки, для оценивания уровня критичности конкретной уязвимости.

2. Анализ защищённости интерфейса. На данном этапе происходит ввод информации об уровнях критичности уязвимостей человеко-компьютерного интерфейса, расчёт суммарного показателя защищённости интерфейса, а также предлагаются возможные контрмеры против обнаруженных угроз.

3. Анализ удобства использования модифицированного интерфейса происходит с учётом применяемых к интерфейсу мер против уязвимостей и угроз.

Далее этапы методики будут рассмотрены более подробно (рисунок 14).

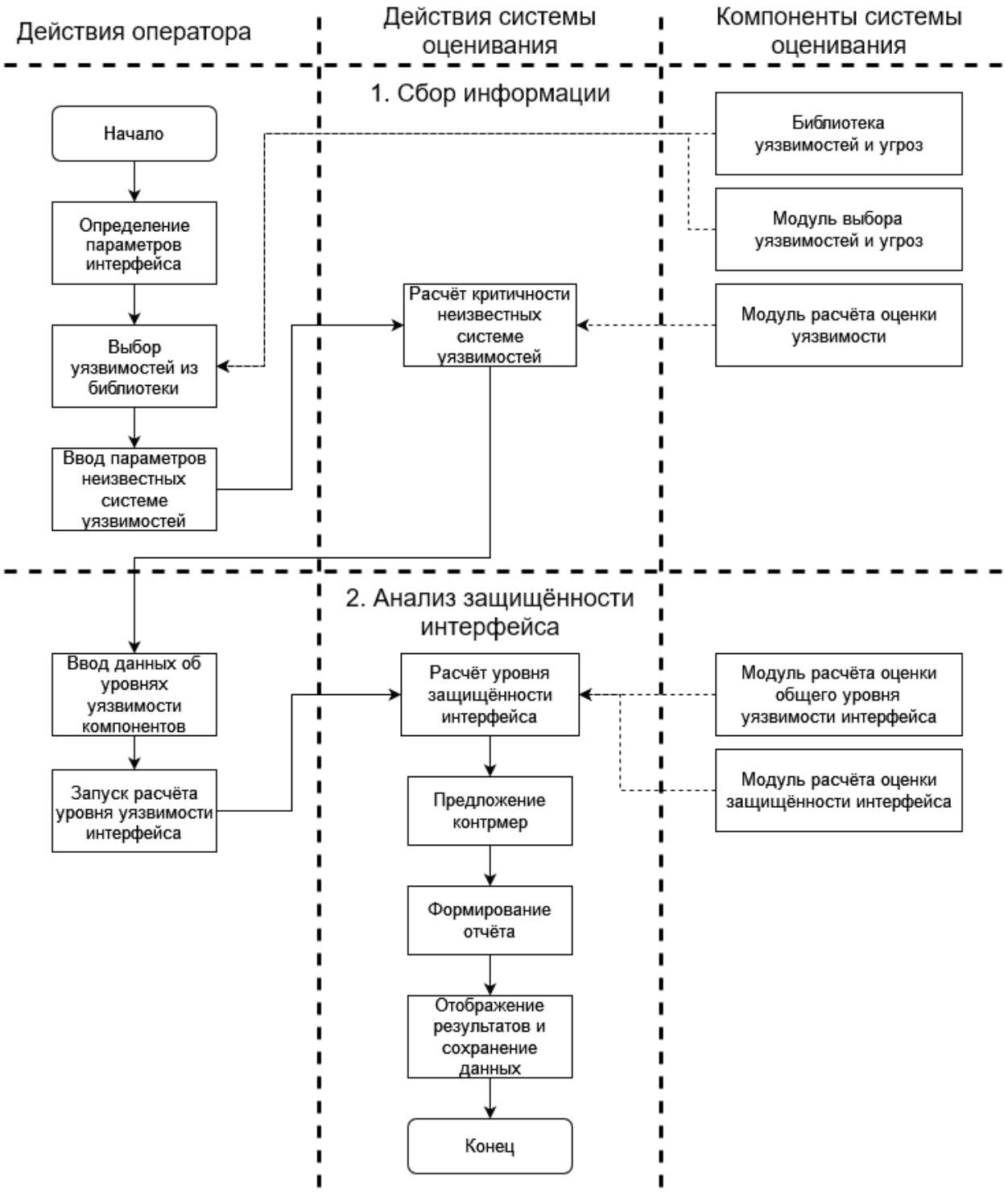


Рисунок 14 – Методика: этапы 1 и 2

Первый этап разделяется на следующие шаги:

1) вначале оператор определяет параметры человеко-компьютерного интерфейса: какой канал восприятия использует интерфейс –

визуальный или звуковой, какой тип технологии – обычный графический, сенсорный, виртуальная реальность и т.д.

2) формирование входных данных при помощи выбора уже известных системе угроз. Выбор осуществляется с помощью программной реализации модуля выбора угроз.

3) оценка неизвестных модулю уязвимостей и угроз, которая осуществляется с помощью программной реализации второго шага алгоритма, рассчитывающего уровень уязвимостей.

4) далее система оценивания производит расчёт уровня критичности уязвимостей, отсутствовавших в модуле выбора, и выдаёт результат.

На втором этапе можно выделить следующие шаги:

1) на основе данных, полученных на предыдущем этапе, оператор вводит данные об уязвимостях в системе.

2) далее оценивается суммарный уровень уязвимостей и оценивается общий уровень защищённости с помощью четвёртого шага алгоритма оценивания защищённости человеко-компьютерного интерфейса.

3) система отображает общий уровень защищённости интерфейса и предлагает возможные контрмеры.

4) оператор или разработчик могут применить контрмеры против возможных угроз.

В конце проводится оценка удобства использования человеко-компьютерного интерфейса с учётом принятых контрмер, чтобы убедиться, что удобство использования не стало ниже допустимого порога. Третий этап, заключающийся в оценивании удобства использования интерфейса, можно разделить на следующие составляющие: (1) подготовка к эксперименту, (2) проведение эксперимента, (3) проведение опроса и (4) статистический анализ результатов (рисунок 15).

1. На стадии подготовки к эксперименту требуется подготовить несколько сценариев различных ситуаций, типичных для информационной безопасности. Для каждого сценария следует разработать свой вариант

визуализации. Эти ситуации должны отвечать выбранной задаче информационной безопасности. Каждый сценарий должен включать в себя несколько заданий на взаимодействие, при этом должны быть задания на все разработанные типы жестов управления. В разных сценариях номера заданий на определённые типы жестов управления должны совпадать, для удобства дальнейшего анализа результатов.

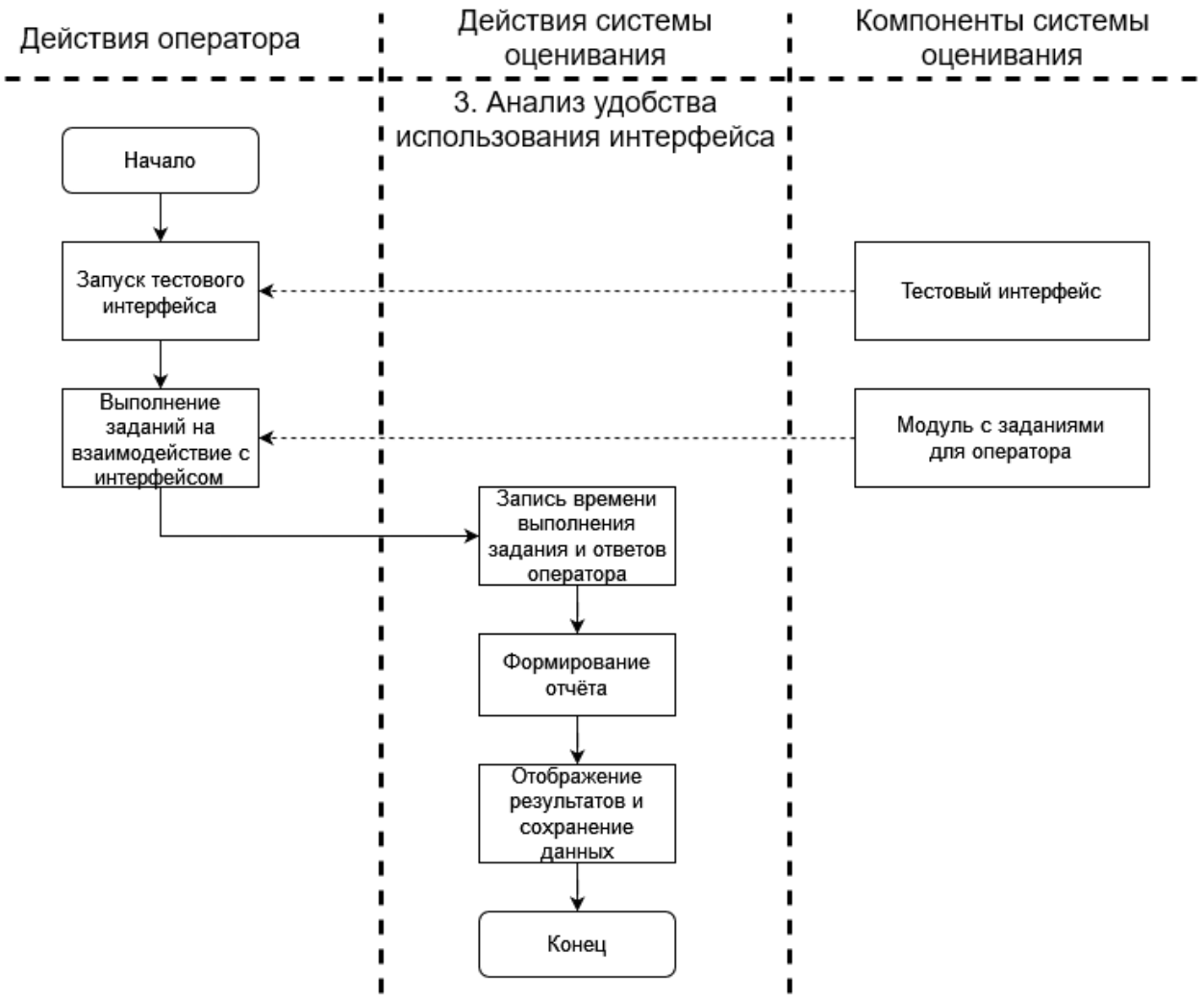


Рисунок 15 – Методика: этап 3

2. Вторая стадия заключается в проведении эксперимента с привлечением подопытных. Причём данный этап включает в себя предварительное обучение и непосредственно эксперимент. Предварительное обучение предполагает, что первые несколько вопросов теста будут направлены на привыкание подопытного к интерфейсу, и при анализе результата учитываться не будут. Сам эксперимент проводится по каждому

сценарию с использованием традиционного решения (управление с помощью клавиатуры и мыши) и оцениваемого (управление с помощью жестов управления на сенсорном экране). При этом определяются время выполнения каждого задания и точность выполнения каждого задания.

3. После теста проводится опрос испытуемых, включающий в себя вопросы об их субъективном отношении к оцениваемой системе. Вопросы могут быть следующими [134]:

(1) «Я буду часто использовать эту систему».

(2) «Эта система неоправданно сложная».

(3) «Я думаю, что систему было легко использовать».

(4) «Мне понадобится помощь технически грамотного человека, чтобы использовать эту систему».

(5) «В системе были хорошо реализованы различные функции».

(6) «Система сделана слишком непоследовательно».

(7) «Я могу предположить, что большинство людей быстро обучатся использовать эту систему».

(8) «Система оказалась очень громоздкой».

(9) «Я чувствовал(а) себя очень уверенно, используя эту систему».

(10) «Мне нужно было выучить много вещей прежде, чем я смог(ла) использовать эту систему».

Каждому вопросу соответствует шкала из пяти делений, пронумерованных от 1 до 5. Чтобы рассчитать оценку, сначала суммируются вклады баллов по каждому вопросу. Оценка по каждому вопросу будет варьироваться от 0 до 4. Для вопросов 1, 3, 5, 7, 9 значением вклада в балл будет позиция на шкале минус 1. Для пунктов 2, 4, 6, 8, 10 значение вклада составляет 5 минус позиция на шкале. Далее следует умножить сумму баллов на 2.5, чтобы получить общее значение оценки. Данная оценка имеет диапазон от 0 до 100. Система считается хорошей, если набрано 80 баллов. Если набрано больше 50 баллов, система дорабатывается. Если набрано 50 баллов и меньше, система отбраковывается.

4. Четвёртый этап предполагает статистический анализ результатов, полученных в ходе выполнения заданий по каждому сценарию. В ходе анализа сравнивается эффективность изначального интерфейса и интерфейса после принятия контрмер. При этом сравниваются распределения по скорости выполнения заданий у двух типов интерфейсов и распределения по точности выполнения заданий. Скорость выполнения задания рассчитывается как время, которое прошло от момента появления текста задания до момента перехода к следующему заданию. Точность выполнения задания рассчитывается как среднее количество ошибок, которые совершили испытуемые при выполнении задания. Чем выше среднее количество ошибок, тем ниже точность. Сравнение распределений для каждого задания производится по верхнему квантилю (Q3).

Третий этап методики предполагает не только формальную оценку полученных результатов, но также учитывает субъективные впечатления испытуемых о качестве взаимодействия с системой информационной безопасности. Поскольку интуитивность не является определяющим параметром в области информационной безопасности, в данной методике она не учитывается, что также позволяет приблизить распределения формальных показателей к нормальным. С помощью данной методики возможно оценить скорость выполнения заданий и точность выполнения заданий по отдельности, так как в некоторых задачах информационной безопасности более важна скорость, а в некоторых – точность.

Исходя из этого, входные данные предлагаемой методики, от которых будет зависеть оперативность и защищённость, представлены ниже:

1) параметры уязвимостей интерфейса;
2) уязвимости и угрозы интерфейсу;
3) тип визуализации (двухмерный или трёхмерный);
4) технология, используемая интерфейсом (сенсорные экраны или виртуальная реальность).

Выходные данные методики следующие:

1) «слабые места» в защищённости интерфейса (для которых применение уязвимостей наносит наибольший ущерб);

2) показатели безопасности для оценивания общего уровня защищённости интерфейса;

3) рекомендации по улучшению защищённости интерфейса.

Предлагаемая методика оценивания человеко-компьютерных интерфейсов позволяет добиться следующих результатов:

1) выявить «слабые места» в защищённости человеко-компьютерного интерфейса;

2) определить недостатки архитектуры человеко-компьютерного интерфейса с точки зрения его защищённости;

3) оценить, насколько сильно повлияет применение выбранных контрмер на общий уровень защищённости интерфейса;

4) оценить, понизилось ли (и насколько) удобство использования человеко-компьютерного интерфейса после принятия контрмер.

3.2 Архитектура и программная реализация системы оценивания защищённости человеко-компьютерных интерфейсов

Система оценивания защищённости человеко-компьютерных интерфейсов состоит из следующих компонентов (рисунок 16):

1) библиотека уязвимостей интерфейсов;

2) модуль выбора известных уязвимостей, находящихся в библиотеке;

3) модуль расчёта уровня критичности уязвимостей;

4) модуль расчёта суммарной защищённости интерфейса;

5) модель интерфейса, поступающая на вход системы;

6) модель уязвимостей, поступающая на вход системы;

7) тестовый интерфейс, защищённость которого необходимо оценить;

8) модуль с заданиями оператора.

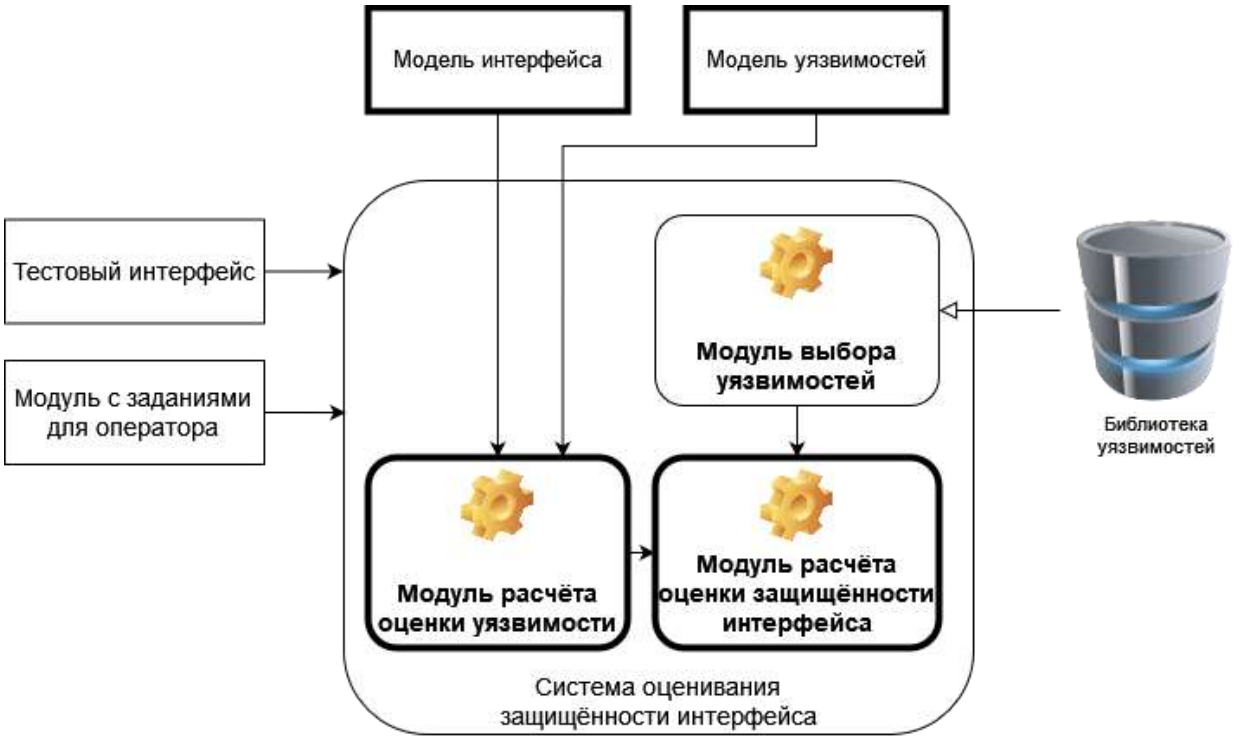


Рисунок 16 – Архитектура программного прототипа системы оценки защищённости человеко-компьютерного интерфейса

Интерфейс для приложения расчёта влияния уязвимости на систему, созданный с помощью языка разметки HTML и свободно распространяемой библиотеки CSS-стилей Bootstrap, представлен на рисунке ниже.

Рисунок 17 – Интерфейс для расчёта влияния уязвимости на систему

На рисунке 17 также представлен результат расчёта для набора входных параметров:

(1) Конфиденциальность – Нет ущерба,

(2) Целостность – Высокий,

(3) Доступность – Высокий,

(4) Урон оператору – Низкий,

(5) Сложность применения – Высокая,

(6) Требуемые привилегии – Высокие,

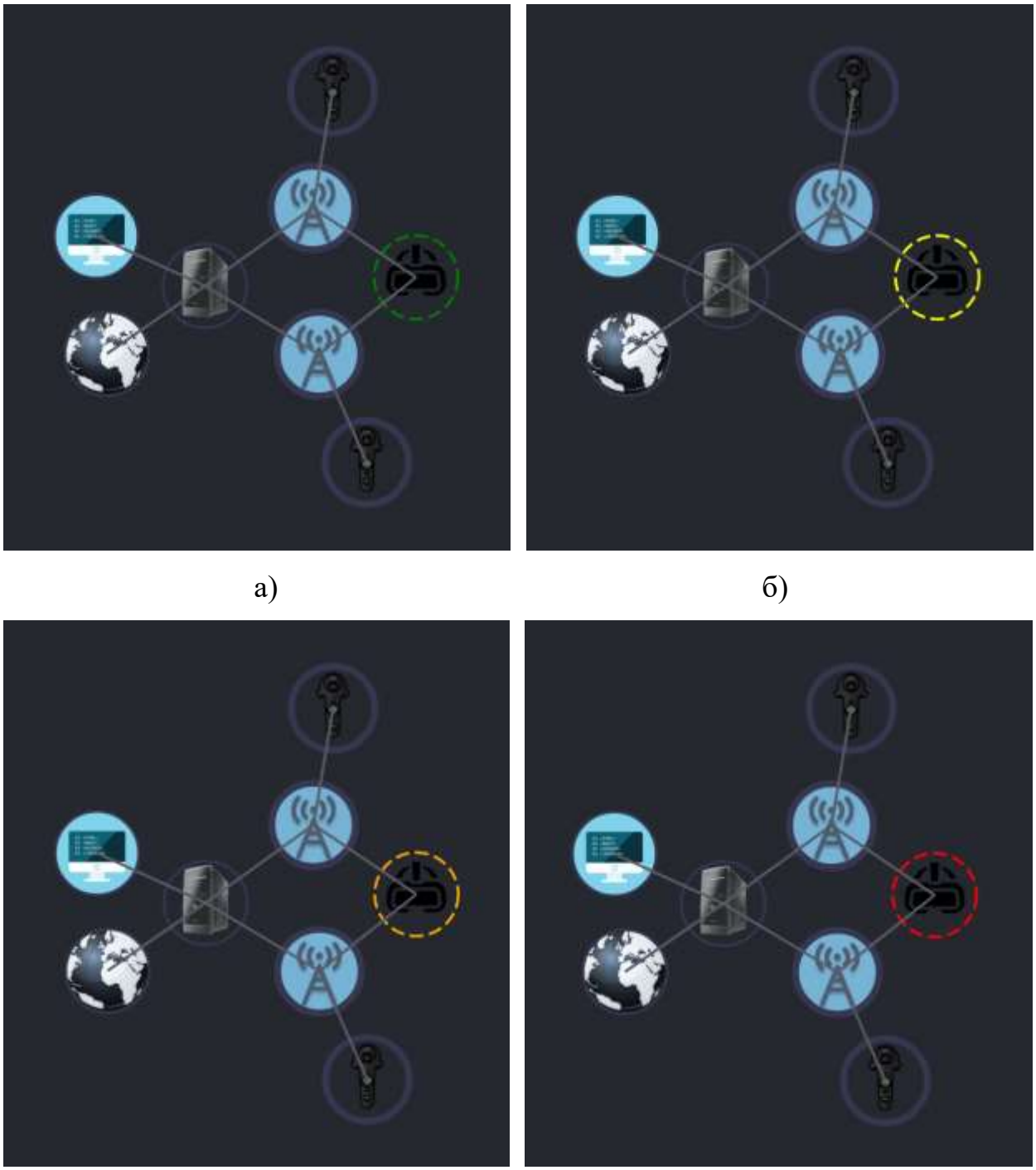
(7) Взаимодействие – Нет (Аудио).

Значение возможного ущерба равно 6.2 и входит в диапазон значений для среднего уровня критичности уязвимости.

Для оценки общего уровня защищённости интерфейса было разработано веб-приложение, реализующее данный алгоритм. Для разработки этого приложения использовался язык программирования JavaScript, для создания графического интерфейса пользователя применялись язык разметки HTML, язык таблиц стилей CSS, свободно распространяемая библиотека CSS-стилей Bootstrap. Реализованный графический интерфейс приложения можно видеть на рисунке 18.

Рисунок 18 – Граф, отображающий связи между элементами интерфейса

Модель оцениваемого интерфейса представлена в виде графа, реализованного по технологии «force-layout» (рисунок 19).



а) б)

в) г)

Рисунок 19 – Выбор состояния элемента интерфейса виртуальной реальности: а) нет уязвимости; б) низкий уровень уязвимости; в) средний уровень уязвимости; г) высокий уровень уязвимости

Взаимодействие с этой моделью происходит при помощи контекстного меню, которое можно вызвать нажатием правой клавиши контроллера

«компьютерная мышь» поверх нужного элемента интерфейса. Модель включает в себя все элементы модели интерфейса «виртуальная реальность»: (1) память компьютера, (2) монитор, (3) две базовые станции, (4) два контроллера, (5) очки виртуальной реальности.

При выборе определённого уровня уязвимости для каждого элемента интерфейса в контекстном меню происходит визуальное отображение взаимодействия оператора с интерфейсом. При выборе значения «Нет уязвимости» в контекстном меню элемент модели интерфейса выделяется зелёной рамкой, данные об уровне уязвимости вносятся в массив данных для дальнейшего расчёта. При выборе значения «Низкий уровень» вокруг элемента появляется жёлтая рамка. При выборе значения «Средний уровень» элемент выделяется оранжевой рамкой. При выборе значения «Высокий уровень» рамка вокруг элемента становится красной. Общий вид интерфейса приложения представлен на рисунке 20.

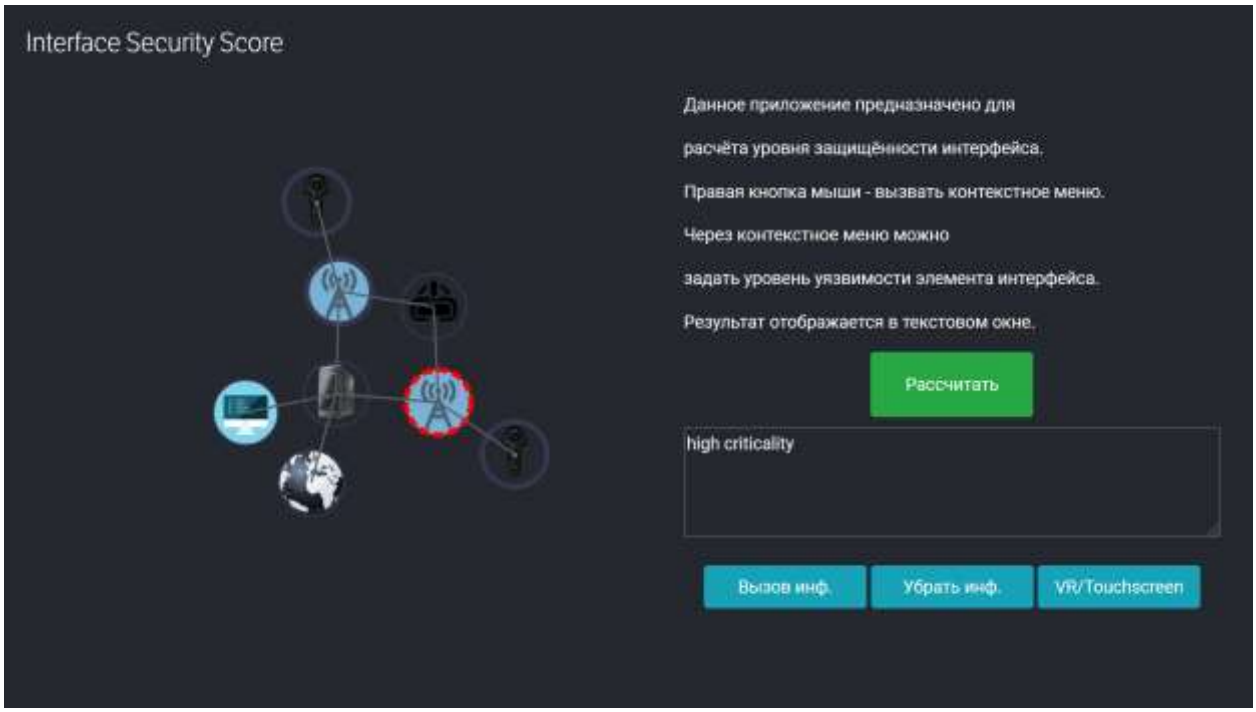


Рисунок 20 – Внешний вид графического интерфейса системы оценки защищённости интерфейса виртуальной реальности

Общий вид интерфейса системы оценки защищённости представлен на рисунке 21. Интерфейс визуально разделён на два блока. В левом блоке

содержится модель визуализации «граф», которая отображает модель интерфейса «виртуальная реальность» или «сенсорные экраны». Вершины графа содержат пиктограммы устройств, входящих в соответствующий интерфейс. Также рядом с каждой вершиной отображается название устройства.

Рисунок 21– Внешний вид графического интерфейса системы оценки защищённости интерфейса сенсорных экранов

Правый блок содержит следующие элементы.

1. Краткое описание приложения вместе с краткой инструкцией по работе с этим приложением.

2. Кнопка для расчёта общего уровня защищённости интерфейса.

3. Текстовое окно, в котором отображается результат после расчёта.

4. Кнопки управления пользовательским интерфейсом приложения:

4.1. Кнопка вызова дополнительной информации, при нажатии на которую рядом с пиктограммами устройств появляется информация об этих устройствах.

4.2. Кнопка, убирающая дополнительную информацию.

4.3. Кнопка для переключения между моделью интерфейса виртуальной реальности и моделью интерфейса на основе сенсорных экранов, при нажатии на которую меняется отображаемая модель интерфейса.

Чтобы рассчитать числовое значение уровня защищённости интерфейса, используется контекстное меню вверху страницы. Для этого выбираются значения рассчитанных ранее уровней критичности уязвимости, при нажатии на кнопку «Рассчитать» в текстовом поле под этой кнопкой отображается числовое значение и уровень уязвимости интерфейса.

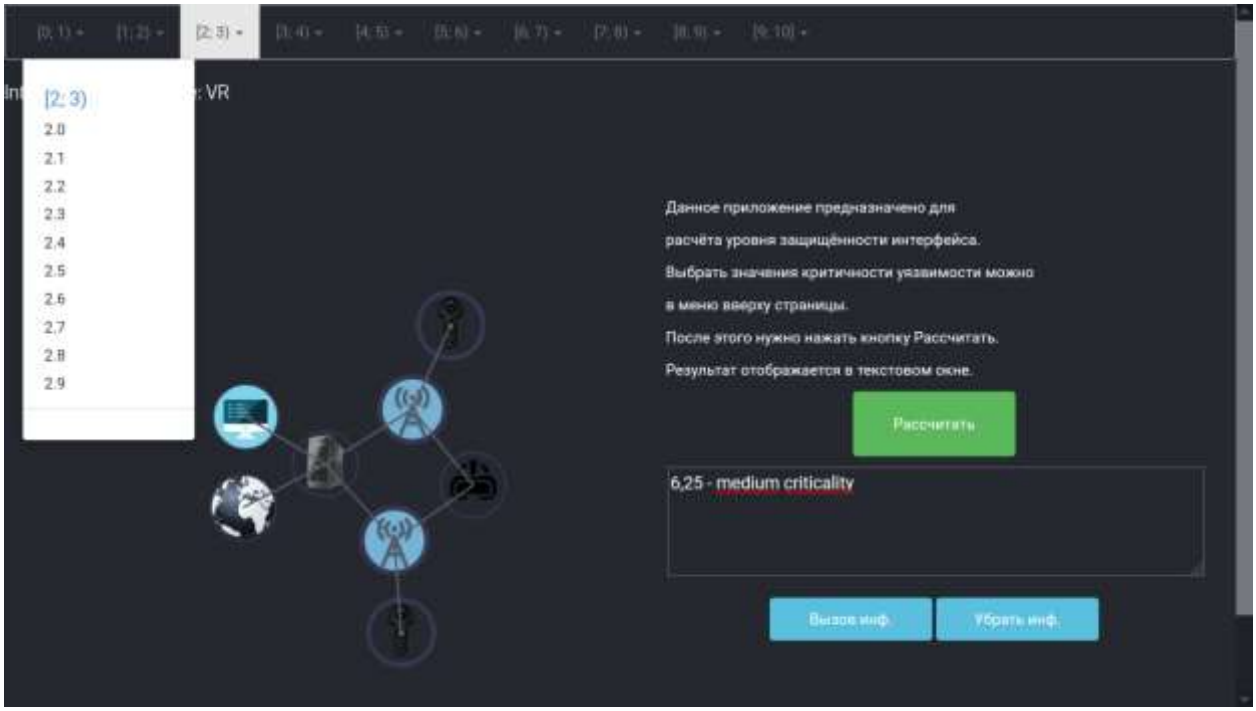


Рисунок 22– Внешний вид графического интерфейса для расчёта уровня уязвимости интерфейса виртуальной реальности

3.3 Экспериментальная оценка предложенной методики оценивания человеко-компьютерных интерфейсов

3.3.1 Описание тестового примера для оценки методики

Во время проведения экспериментов последовательно выполнялись шаги алгоритмов оценивания уязвимостей и человеко-компьютерного интерфейса для сенсорного интерфейса управления компьютерной сетью.

Для проведения экспериментов был разработан программно-аппаратный стенд на основе технологии сенсорных экранов.

3.3.1.1 Сенсорные экраны

Аппаратная часть представляла собой сенсорный монитор Dell P2418HT с возможностью регулирования высоты и наклона, а также персональный компьютер на базе процессора Intel Core i5 с установленной операционной системой Windows 10. Ниже представлено изображение прогрммно-аппаратного стенда (рисунок 23).

Рисунок 23 – Внешний вид программно-аппаратного стенда

Программная часть стенда представляла собой веб-приложение, которое открывалось по определённому электронному адресу в браузере. Приложение написано на языке JavaScript. Для реализации жестов использовалась открытая библиотека hammer.js, для визуализации графа компьютерной сети использовалась библиотека D3.js, для оформления интерфейса был использован набор графических средств Bootstrap.

Таким образом, приложение представляло собой модель визуализации для компьютерной сети и интерфейс управления, состоящий из кнопок перехода к заданиям теста. Интерфейс взаимодействия с самой визуальной

моделью осуществлён в двух вариантах в соответствии с тестами: (1) традиционный, с помощью кнопок, и (2) сенсорный, с помощью жестов на сенсорном экране.

Ниже (рисунок 24 и рисунок 25) представлены изображения интерактивной модели компьютерной сети.

Рисунок 24 – Визуальная модель компьютерной сети

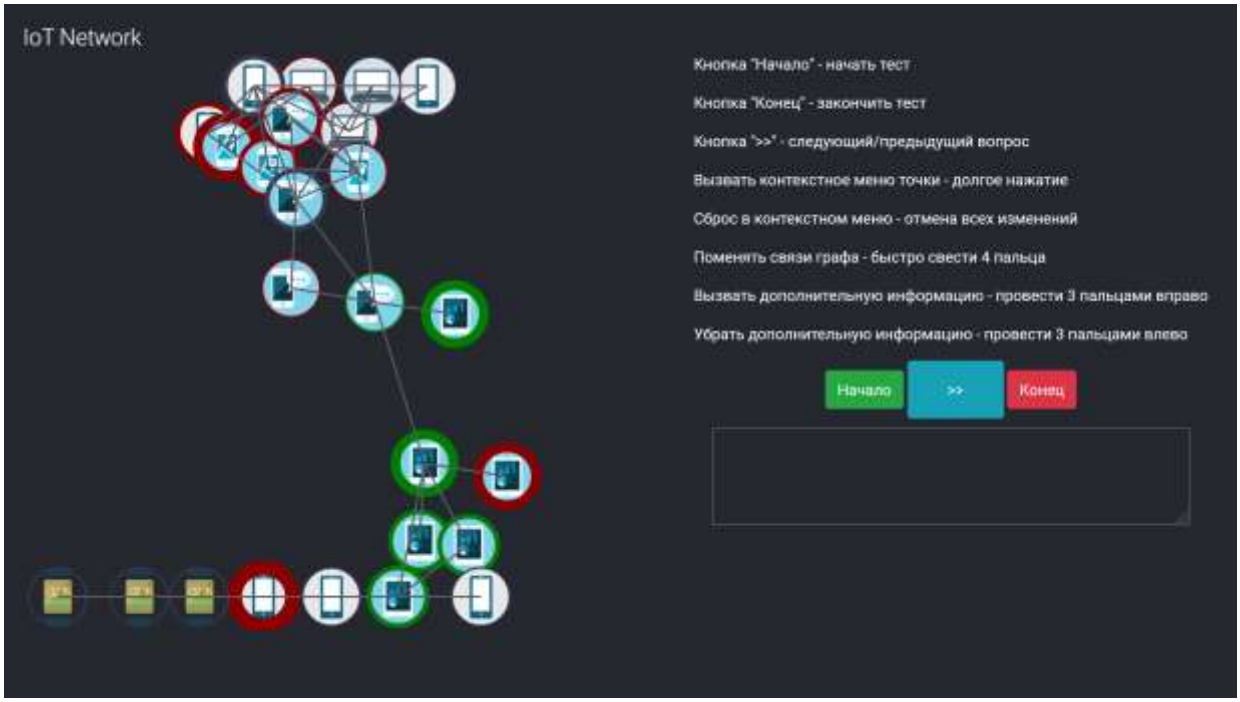


Рисунок 25 – Визуальная модель сети Интернета вещей

В качестве модели визуализации были выбраны силовые графы (force graph), так как они подходят для визуализации многих объектов информационной безопасности с множеством связей путем самоорганизации положения вершин и ребер графа. Силовые графы подходят для отображения сложных структур, таких как компьютерные сети, социальные сети и т.п. Для экспериментов были построены два вида графов: граф централизованной сети устройств и исправленный граф компьютерной сети.

На рисунке 24 представлена модель централизованной компьютерной сети. Модель представляет собой централизованный граф, вершинами которого являются отображения подключенных устройств, представленные в виде узнаваемых символов-изображений, центром этого графа является отображение глобальной сети Интернет. Как видно на рисунке, данная компьютерная сеть состоит из трёх подсетей, сообщающихся между собой через сеть Интернет.

На рисунке 25 представлена модель децентрализованной сети Интернета вещей. Модель представляет собой децентрализованный граф, вершинами которого также являются отображения подключенных устройств. Как видно на рисунке, данная сеть состоит из множества устройств, общающихся между собой.

Ниже (рисунки 26 и 27) представлены примеры взаимодействия с моделью компьютерной сети. На рисунке 26 представлен процесс помещения заражённого устройства в карантин с помощью контекстного меню. Для этого нужно сначала выбрать устройство с помощью долгого нажатия, а затем выбрать пункт «Поместить в карантин хост».

Также на рисунке 27 представлен процесс смены физических связей компьютерной сети на логические, т.е. с помощью этих связей показано, какие устройства общаются между собой. Таким образом, связанными остаются только хосты, другие «вспомогательные» устройства «отсоединены».

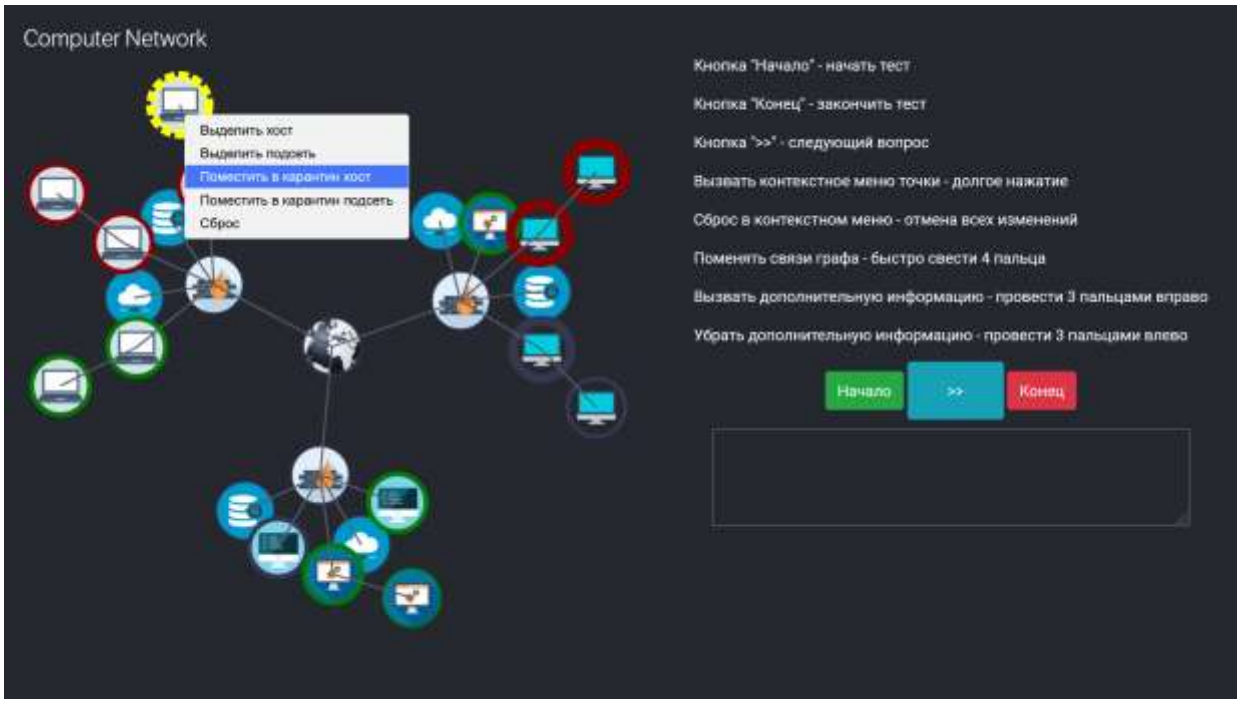


Рисунок 26 – Пример взаимодействия оператора с визуальной моделью компьютерной сети – помещение хоста в карантин

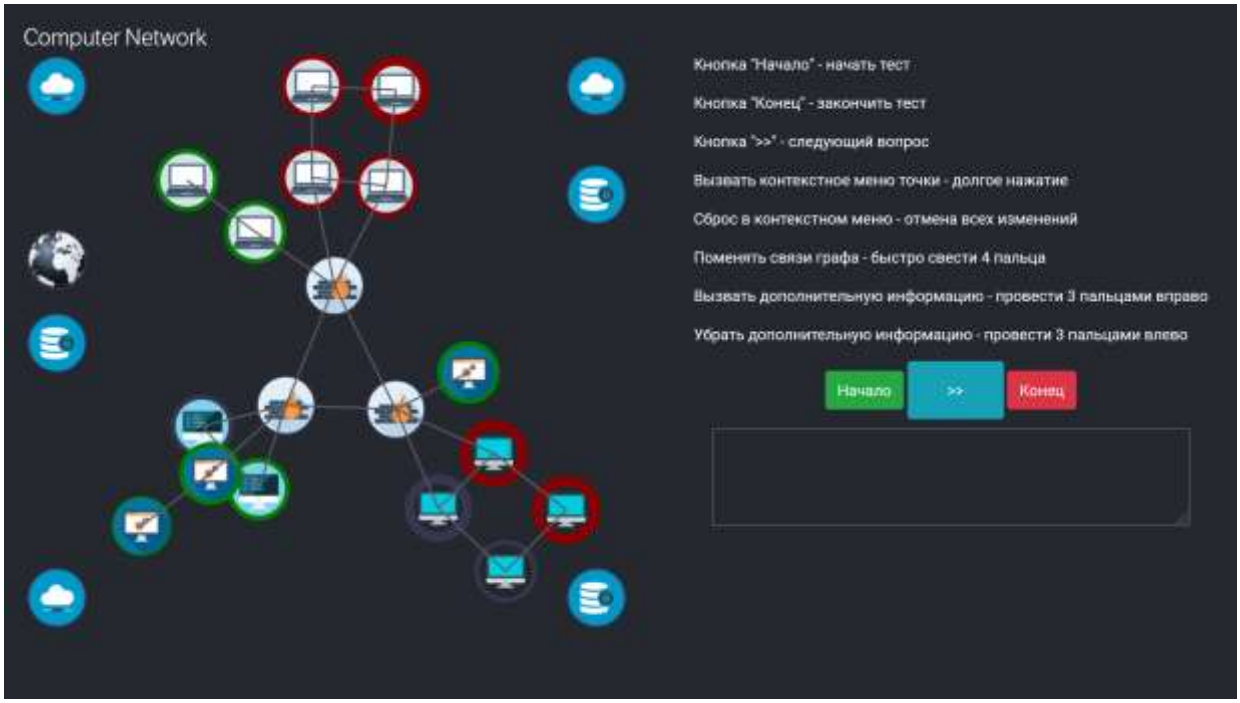


Рисунок 27 – Пример взаимодействия оператора с визуальной моделью компьютерной сети – смена топологии сети с физической на логическую

Результаты эксперимента записывались в файл формата *.json, и представляли собой многомерный массив, каждый элемент которого состоял из описания события и времени срабатывания этого события. Файлы сохранялись в соответствующую папку.

В окне приложения слева визуализация сети, справа панель управления и окно, где появляется текст заданий. В тесте содержатся 16 заданий. 7 первых заданий не учитываются и направлены на то, чтобы тестируемый ознакомился с интерфейсом, привык к нему. 9 заданий на три типа жестов: в 3 заданиях оценивается долгое нажатие, в 3 заданиях «swipe», в 3 заданиях «pinch».

3.3.1.2 Виртуальная реальность

Аппаратная часть представляла собой шлем виртуальной реальности HTC Vive и контроллеры, а также персональный компьютер на базе процессора Intel Core i5 и видеокарты Geforce 2060 Super с установленной операционной системой Windows 10. Ниже представлено изображение прогрммно-аппаратного стенда (рисунок 28).

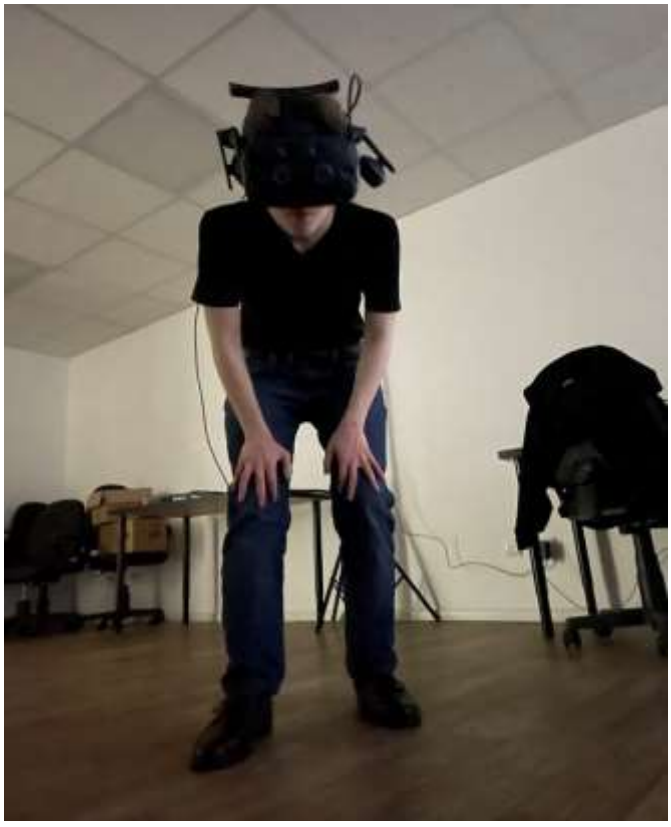


Рисунок 28 – Внешний вид программно-аппаратного стенда

Таким образом, интерфейс виртуальной реальности включал в себя следующие элементы: шлем виртуальной реальности для показа изображения, контроллеры для управления изображением, базовые станции

для координации шлема виртуальной реальности с контроллерами и контроля положения шлема виртуальной реальности в пространстве. Элементы интерфейса были подключены к персональному компьютеру. Потоки данных между элементами интерфейса можно видеть на рисунке 29:

1) оператор видит визуализацию;

2) оператор воздействует на визуализацию с помощью контроллеров;

3) визуализация отображается на мониторе;

4) визуализация генерируется из файла в памяти компьютера;

5) компьютер связывается с базовыми станциями;

6) базовые станции отслеживают шлем виртуальной реальности;

7) базовые станции связываются с контроллерами.

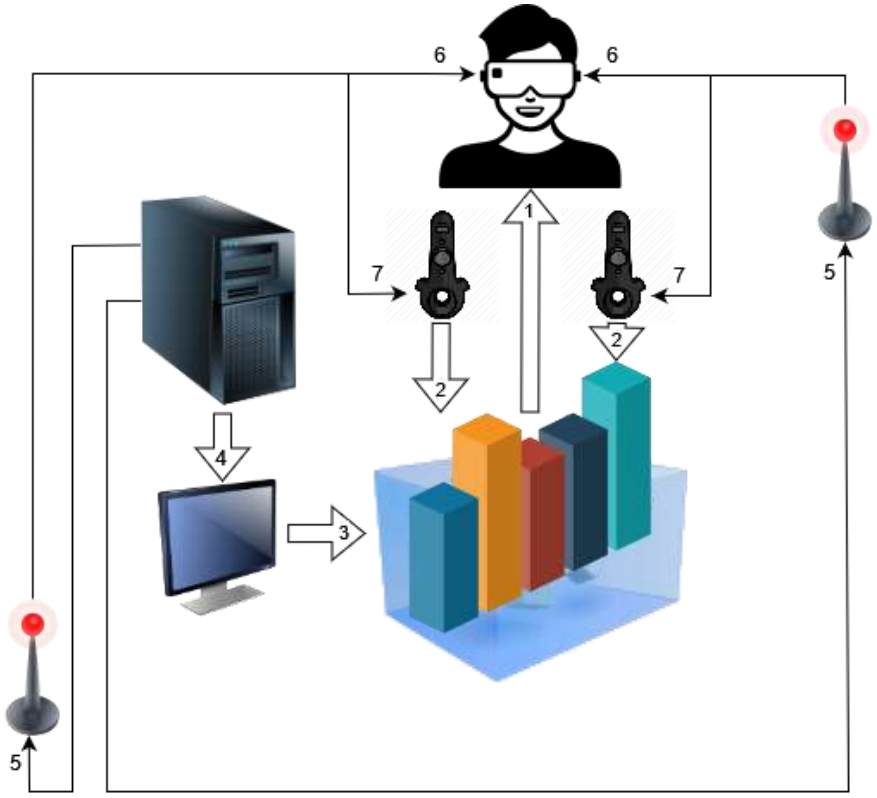


Рисунок 29 – Схема взаимодействия основных элементов интерфейса виртуальной реальности

Во время проведения экспериментов были разработаны различные модели визуализации, использующиеся в информационной безопасности.

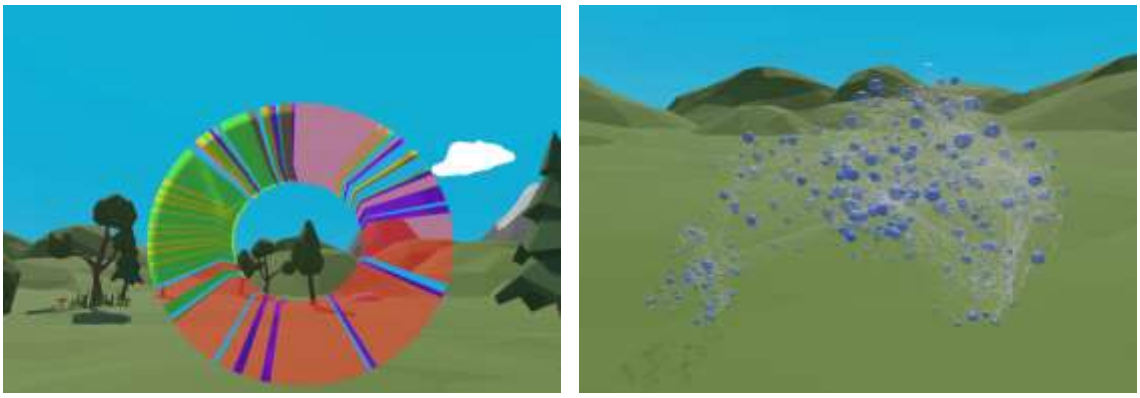


Рисунок 30 – Трёхмерные модели визуализации для визуального анализа данных безопасности

Результаты эксперимента записывались в файл формата *.json, и представляли собой многомерный массив, каждый элемент которого состоял из описания события и времени срабатывания этого события. Файлы сохранялись в соответствующую папку.

3.3.2 Экспериментальная оценка методики оценивания защищённости

В данном разделе будут рассмотрены выполнение поставленной задачи, а также основные нефункциональные требования, предъявляемые к методике. Эти требования разделены на две группы: требования к (1) оперативности и (2) ресурсопотреблению. Проверка соответствия поставленной задаче, а также требованиям оперативности и ресурсопотребления проводится только для оценивания защищённости интерфейса, так как её алгоритмы имеют программную реализацию, в то время как оценивание удобства использования требует проведения экспериментов, следовательно, это растянутый во времени процесс, оценивание которого приводится в отдельном разделе 3.3.2.4.

3.3.2.1 Оценка возможности поиска интерфейса с минимальной уязвимостью

В качестве примера можно рассмотреть интерфейсы, уязвимые к атакам через наблюдение (shoulder-surfing attacks). Данный тип атак

предполагает, что злоумышленник, подсматривая за действиями оператора, может раскрыть конфиденциальную информацию.

Сенсорные экраны. Рассмотрим сенсорный интерфейс управления безопасностью сети, использующий модели визуализации с изображениями подключенных к сети устройств. Чтобы снизить ущерб конфиденциальности в данном случае, проектировщик может убрать изображения и вместо них поставить цветные вершины. Ущерб конфиденциальности снизится за счёт того, что для совершения атаки злоумышленник должен заранее знать условные цветовые обозначения. Кроме того, атаку совершить становится сложнее, так как вершины в этом случае более мелкие, и потому злоумышленнику сложнее зафиксировать действия оператора. Более подробно пример рассмотрен ниже.

Имеется сенсорный человеко-компьютерный интерфейс, представляющий собой устройство с сенсорным экраном (например, планшет). Экран устройства отображает приложение компьютерной безопасности с визуализацией графа компьютерной сети. Хосты обозначены крупными вершинами с графическим изображением устройств. Такой интерфейс обладал двумя уязвимостями:

1) визуализация на сенсорном устройстве могла быть подвержена угрозе потери конфиденциальности (атаки через наблюдение);

2) частота обновления сенсорного экрана причиняла дискомфорт оператору.

Входные данные 1 уязвимости: есть угроза конфиденциальности, угрозы целостности нет, угрозы доступности нет, угрозы оператору нет, сложность выполнения низкая, требуемых привилегий нет, канал визуальный, взаимодействие «оператор». В этом случае результат базовой оценки уязвимости составил – 6.7 (раздел 2.3.1).

Входные данные 2 уязвимости: конфиденциальность – нет, целостность – нет, доступность – нет, урон оператору – низкий, сложность – высокая,

привилегии – высокие, взаимодействие – злоумышленник, канал – видео, базовая оценка– 2,5.

Расчёт по формуле 11 дал результат суммарной оценки уязвимости $BS_{\sum}(I_n)$ – также 6,725. Таким образом, общий уровень уязвимости интерфейса: средний.

Чтобы снизить возможный урон конфиденциальности, можно заменить крупные вершины с изображением устройств на вершины меньших размеров, отличающиеся друг от друга только цветом. При этом разными цветами обозначаются разные типы устройств, например: синяя вершина – сервер, зелёные – маршрутизаторы, фиолетовые – коммутаторы, жёлтые и белые – хосты в случае централизованной сети (рисунок 31).

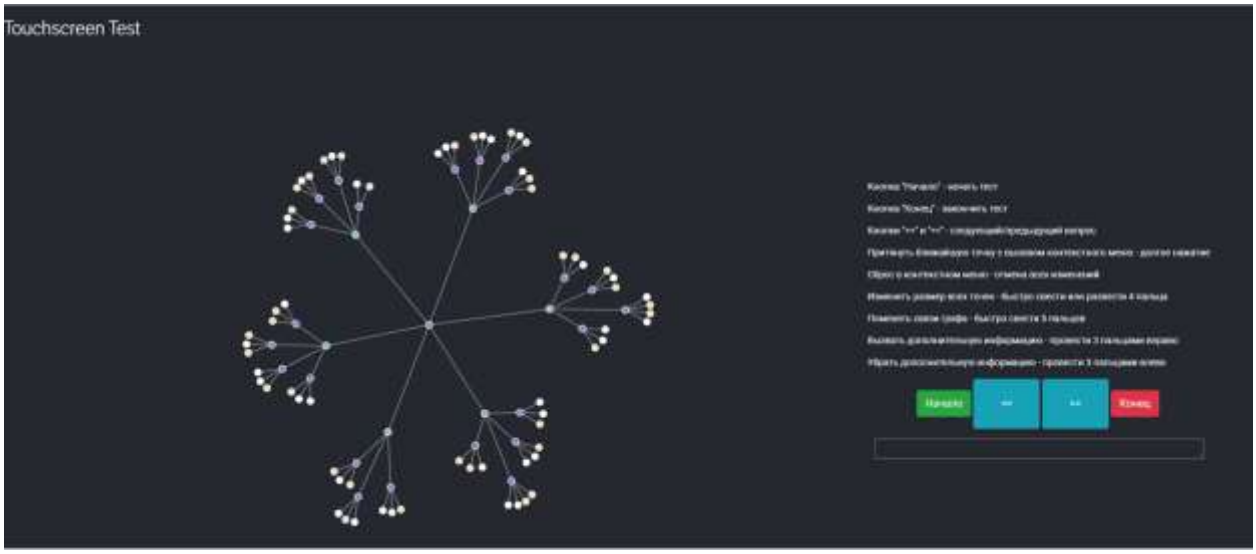


Рисунок 31 – Модель визуализации для отображения централизованной сети устройств после принятия контрмер

Подсоединённые устройства связаны между собой, не связанные устройства не подключены к сети и отображают разные типы устройств интернета вещей (рисунок 32). Ущерб конфиденциальности здесь возможен лишь в том случае, если злоумышленник заранее знаком с цветовыми обозначениями устройств. Кроме того, на устройстве настроили комфортную для оператора частоту обновления экрана. Таким образом, для обеих устранённых уязвимостей результат оценки равен 0. Поэтому расчёт по

формуле 11 дал результат суммарной оценки уязвимости $BS_{\sum}(I_n)$ – 0. Таким образом, из двух оцениваемых интерфейсов можно выбрать исправленный интерфейс, так как он обладает минимальной уязвимостью. Следовательно, разработчик приложения, воспользовавшись знаниями о возможной угрозе интерфейсу, может принять соответствующие контрмеры и ощутимо снизить угрозу. Общий уровень защищённости интерфейса: интерфейс защищён. Расчёты представлены в таблице 8.

Рисунок 32 – Модель визуализации для отображения децентрализованной сети устройств после принятия контрмер

Таблица 8 – Пример поиска наименее уязвимого интерфейса, сенсорные экраны

№	Название интерфейса	Тип	Кол-во уязв.	Тип уязв.	$BS_{in}(I_n)$	$BS_{\sum}(I_n)$
1	Граф компьютерной сети, с уязвимостью	СЭ	2	Уязвимость к наблюдению	6,7	6,25
				Некомфортная частота обновления экрана	2,5	
2	Граф компьютерной сети, исправленный	СЭ	0	-	**0**	**0**
Min уязвимость		**0**				

Виртуальная реальность. Потенциально интерфейс виртуальной реальности может быть более устойчив против атак через наблюдение, чем интерфейс на основе сенсорных экранов, так как оператор рассматривает

модели визуализации с помощью головного дисплея, мониторы на котором не видны посторонним. Однако изображение с головного дисплея также часто транслируется на обычный монитор персонального компьютера, на котором запущено приложение. Также злоумышленник может заменить конфигурационный файл виртуальной среды, а базовые станции подвержены атакам типа спуфинг. Таким образом, данный пример интерфейса обладает тремя уязвимостями:

1) уязвимость к наблюдению;

2) возможность подмены файла конфигурации;

3) уязвимость ИК-базовых станций к спуфингу.

Входные данные 1 уязвимости: есть угроза конфиденциальности, угрозы целостности нет, угрозы доступности нет, угрозы оператору нет, сложность выполнения низкая, требуемых привилегий нет, канал визуальный, взаимодействие «оператор». В этом случае результат оценки уязвимости с использованием алгоритма оценивания уязвимостей (раздел 2.3) – 6.7 (средний уровень уязвимости).

Входные данные 2 уязвимости: нет угрозы конфиденциальности, угроза целостности высокая, угрозы доступности нет, угроза оператору низкая, сложность выполнения высокая, требуемых привилегий нет, канал визуальный, взаимодействие «злоумышленник». В этом случае результат оценки уязвимости с использованием алгоритма оценивания уязвимостей (раздел 2.3) – 5.4 (средний уровень уязвимости).

Входные данные 3 уязвимости: угрозы конфиденциальности нет, угроза целостности высокая, угроза доступности высокая, угрозы оператору нет, сложность выполнения высокая, требуемых привилегий нет, канал визуальный, взаимодействие «злоумышленник». В этом случае результат оценки уязвимости с использованием алгоритма оценивания уязвимостей (раздел 2.3) – 7.1 (высокий уровень уязвимости).

Расчёт по формуле 11 также дал результат суммарной оценки уязвимости $BS_{\sum}(I_n)$ – 896,05. Таким образом, общий уровень уязвимости интерфейса: высокий.

Чтобы снизить возможный урон конфиденциальности, требуется сворачивать приложение на главном мониторе персонального компьютера, чтобы не оставить злоумышленнику возможности следить за действиями оператора. Кроме того, требуется настроить политики безопасности и аутентификации. Тогда уязвимости 1 и 3 окажутся исправлены, интерфейс станет уязвим только к спуфингу базовых станций. В этом случае, расчёт по формуле 11 также дал результат суммарной оценки уязвимости $BS_{\sum}(I_n)$ – 883,95. Оператор может выбрать исправленный интерфейс, так как он обладает минимальной уязвимостью из двух оцениваемых. Расчёты представлены в таблице 9.

Таблица 9 – Пример поиска наименее уязвимого интерфейса, виртуальная реальность

№	Название интерфейса	Тип	Кол-во уязв.	Тип уязв.	$BS_{in}(I_n)$	$BS_{\sum}(I_n)$
1	Трёхмерный граф компьютерной сети, с уязвимостью	ВР	3	Уязвимость к наблюдению	6,7	896,05
				Возможность подмены файла конфигурации	5,4	
				Уязвимость ИК-базовых станций к спуфингу	7,1	
2	Трёхмерный граф компьютерной сети, исправленный	ВР	1	Уязвимость ИК-базовых станций к спуфингу	7,1	883,95
Min уязвимость		**883,95**				

Из результатов эксперимента видно, что поиск наиболее защищённого интерфейса (с минимальной уязвимостью) произведён успешно, таким образом, поставленная задача исследования выполняется.

3.3.2.2 Оценка оперативности

Основные этапы методики оценивания защищённости:

(1) ввод данных оператором для оценки уязвимости

(2) расчёт оценки уязвимости;

(3) ввод данных оператором об уязвимостях системы;

(4) расчёт суммарной оценки защищённости интерфейса.

Следовательно, количество времени, затрачиваемого на выполнение методики, складывается из отрезков времени, необходимых для выполнения этапов методики, можно представить в виде следующего выражения:

$$TIME = T_1 + T_2 + T_3 + T_4, \quad (12)$$

где T_i – отрезок времени, необходимый для выполнения одного этапа методики.

Для расчёта суммарного показателя времени применялись следующие действия.

(1) На первом этапе методики при первом нажатии оператором на кнопку записывалось время начала работы с программным алгоритмом, реализующим данный этап методики.

(2) На четвёртом этапе после завершения расчёта общего уровня защищённости интерфейса записывалось время окончания работы с программным алгоритмом, реализующим данный этап методики.

(3) Далее, на основании полученных данных о времени начала и окончания выполнения методики, рассчитывался временной отрезок по формуле

$$\Delta TIME = T_{end} - T_{start}, \quad (13)$$

где T_{start} – время начала выполнения методики, T_{end} – время окончания выполнения методики.

(4) Шаги с 1 по 3 повторялись 20 раз.

(5) В конце рассчитывалось максимальное и среднее значения времени, затраченного на выполнение методики.

На графике (рисунок 33) представлены значения временных отрезков. Отсюда видно, что время, затрачиваемое на выполнение методики, не

превышает заданного допустимого отрезка в 3 минуты и в среднем составляет 29,5 секунд.

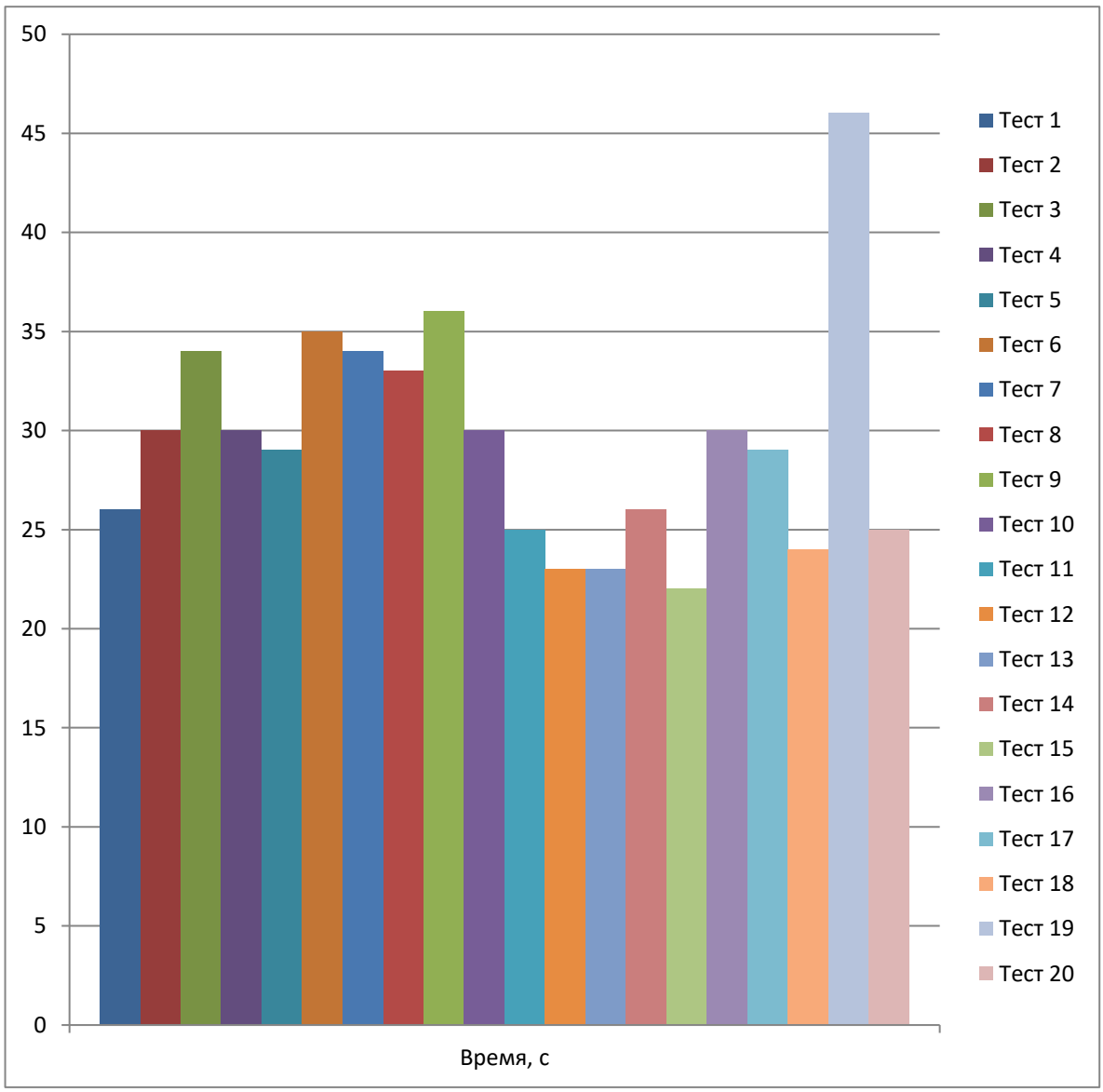


Рисунок 33 – Значения временных отрезков выполнения методики

Время, за которое программный прототип может выполнить методику, рассматривается как случайная величина, при этом её вероятность подчиняется нормальному закону распределения [142]. Для оценки времени, затраченного на выполнение методики, часто применяют бета-распределение на интервале $[t_{\min}, t_{\max}]$, плотность которого вычисляется с помощью набора условий [143]:

$$f(t) = \begin{cases} \frac{(t-t_{min})^{\alpha-1}(t_{max}-t)^{\beta-1}}{(t_{max}-t_{min})^{\alpha+\beta-1}B(\alpha,\beta)}, t_{min} \le t \le t_{max}, \\ 0, t_{max} \le t \le t_{min} \end{cases} \quad (14)$$

где:

- t – отрезок времени, за который методика будет выполнена полностью;
- $t_{\min}$ – минимальный отрезок времени, затраченный на выполнение методики;
- $t_{\max}$ – максимальный отрезок времени, затраченный на выполнение методики;
- $B(\alpha,\beta)$ – функция Эйлера, где $\alpha > 0$, $\beta > 0$ – параметры бета-распределения.

Расчёт предполагаемого времени, за которое методика будет выполнена, а также дисперсии предполагаемого отрезка времени, производится по двухоценочной методике [142]:

$$T_i = \frac{3T_i^{min}+2T_i^{max}}{5}, \quad (15)$$

$$\sigma^2(T_i) = 0{,}4(T_i^{max} + T_i^{min})^2. \quad (16)$$

Вероятность того, что отрезок времени, за которое выполняется методика, будет меньше или равен допустимому пороговому значению $T^{ДОП}$, рассчитывается по формуле:

$$P_{\text{СВ}}(T \le T^{\text{ДОП}}) = \Phi(Z), \quad (17)$$

где $\Phi(Z)$ – значение функции Лапласа. При этом:

$$z = \left(\frac{T^{\text{ДОП}} - \sum_{i=1}^{n} T_i}{\sqrt{\sum_{i=1}^{n} \sigma_i^2(T_i)}}\right). \quad (18)$$

Полученные в ходе экспериментов минимальный и максимальный отрезки времени приведены в таблице 10.

Таблица 10 – Показатели оперативности для методики оценивания защищённости интерфейса

T_i^{min}, сек	T_i^{max}, сек	$T_i = \frac{3T_i^{min} + 2T_i^{max}}{5}$	$\sigma^2(T_i) = 0{,}4(T_i^{max} + T_i^{min})^2$
22	*46*	*31,6*	*230,4*

Тогда функция Лапласа $\Phi(Z)$ для порогового значения $TIME^{\text{ДОП}} = 180$ секунд будет равна:

$$\Phi\left(\frac{TIME^{\text{ДОП}} - \sum_{i=1}^{1} T_i}{\sqrt{\sum_{i=1}^{1} \sigma_i^2(T_i)}}\right) = \left(\frac{60-31{,}6}{\sqrt{230{,}4}}\right) \approx 1{,}87 \qquad (19)$$

Из полученных значений функции Лапласа следует, что вероятность того, что методика будет полностью выполнена за заданное время, оказывается $P_{\text{ОП}}(TIME_I \leq TIME^{\text{ДОП}}) \geq 0{,}9911$. Данное значение соответствует требованиям, предъявляемым к *оперативности* ($P_{\text{ОП}}^{\text{ДОП}} = 0{,}99$). Таким образом, требование к оперативности $P_{\text{ОП}}(TIME_I \leq TIME^{\text{ДОП}}) \geq P_{\text{ОП}}^{\text{ДОП}}$, поставленное в разделе 1.4, выполняется.

3.3.2.3 Оценка ресурсопотребления

Свойство *ресурсопотребления* отражает, какие ресурсы должны быть затрачены на реализацию анализа защищённости интерфейса, в том числе программные, аппаратные и кадровые ресурсы, необходимые данные, и т.д., а также из количество. Для оценки ресурсопотребления требуется измерить ряд показателей:

(1) использование центрального процессора

$$R_{\text{ЦП}} = \frac{Q_{\text{ЦП}}^{\text{М}}}{Q_{\text{ЦП}}^{\text{ОБЩ}}}, \qquad (20)$$

где $Q_{\text{ЦП}}^{\text{М}}$ – процессорное время, которое было затрачено на выполнение программы, реализующей методику, $Q_{\text{ЦП}}^{\text{ОБЩ}}$ – общее доступное время центрального процессора;

(2) использование канала сети

$$R_{\text{СЕТЬ}} = \frac{Q_{\text{СЕТЬ}}^{\text{М}}}{Q_{\text{СЕТЬ}}^{\text{ОБЩ}}}, \qquad (21)$$

где $Q_{\text{СЕТЬ}}^{\text{М}}$ – объем данных, переданных и полученных по сети во время эксперимента, $Q_{\text{СЕТЬ}}^{\text{ОБЩ}}$ – максимально объем данных, которые возможно передать по сети за то же время;

(3) используемый объём жёсткого диска

$$R_{\text{ЖД}} = \frac{Q_{\text{ЖД}}^{\text{М}}}{Q_{\text{ЖД}}^{\text{ОБЩ}}}, \quad (22)$$

где $Q_{\text{ЖД}}^{\text{М}}$ – значение величины потребовавшегося при выполнении методики объема жесткого диска, $Q_{\text{ЖД}}^{\text{ОБЩ}}$ – суммарный объем жесткого диска;

(4) используемый объём оперативной памяти

$$R_{\text{ОП}} = \frac{Q_{\text{ОП}}^{\text{М}}}{Q_{\text{ОП}}^{\text{ОБЩ}}}, \quad (23)$$

где $Q_{\text{ОП}}^{\text{М}}$ – объем оперативной памяти, потребовавшийся для выполнения программной реализации методики, $Q_{\text{ОП}}^{\text{ОБЩ}}$ – общий объем оперативной памяти.

Значения, полученные при оценке ресурсопотребления, соответствуют тем значениям, которые были заданы в требованиях, если все четыре показателя удовлетворяют условию $r \leq R^{\text{ДОП}}$. Допустимое значение затрачиваемых ресурсов $R^{\text{ДОП}} = 0{,}3$, при том, что ещё около 30% ресурсов затрачивается на задачи операционной системы.

В качестве экспериментального стенда был задействован персональный компьютер, соответствующий стандартному рабочему месту. Экспериментальный стенд для проведения оценки обладал следующими характеристиками:

1. центральный процессор использовавшегося компьютера: Intel® Core(TM) i5;
2. объем всего жесткого диска: 1 Тб;
3. объем всей оперативной памяти: 16 Гб.
4. при этом максимальная пропускная способность канала сети составила: 100 Мб/с;

Во время экспериментальной оценки показателей ресурсопотребления были получены следующие значения:

1. Всего было использовано 4% ресурсов центрального процессора, то есть, показатель $R_{ЦП}$=0,04.

2. Канал сети использовался для обмена данных с веб-приложением, а также для получения данных об уязвимостях от базы данных. Максимальное значение 1 Мбит/с, то есть, значение показателя $R_{СЕТЬ}$=0,01.

3. Размер программной реализации предложенной методики составляет около 500 Кб, то есть, значение показателя $R_{ЖД}$≤0,000001.

4. Используемый объём оперативной памяти составил 8% (до запуска 27%, после запуска 35%). Таким образом, показатель ресурсопотребления оперативной памяти составил $R_{ОП}$=0,08.

Представленные выше значения показателей соответствуют поставленным в первой главе требованиям к ресурсопотреблению, поскольку $P_{\text{РЕС}}(r \le R^{\text{ДОП}})$=1, что соответствует требованию $P_{\text{РЕС}}(r \le R^{\text{ДОП}}) \ge P_{\text{РЕС}}^{\text{ДОП}}$, где $P_{\text{РЕС}}^{\text{ДОП}} = 0{,}99$. Следовательно, *ресурсопотребление* программной реализации методики удовлетворяет предъявленным требованиям.

Таким образом, все предъявляемые требования выполняются (таблица 11).

Таблица 11 – Требования к системе оценивания защищённости интерфейса

Свойство	**Показатели**	**Требования**
Оперативность	Вероятность того, что время, необходимое для получения результата оценки защищенности, не будет превышать допустимое значение.	$P_{\text{ОП}}(46\text{ с} \le 1\text{ мин}) = 0{,}9911 > 0{,}99$
Ресурсо-потребление	Вероятность того, что количество использованных ресурсов не будет превышать допустимое значение.	$P_{\text{РЕС}}(r \le 0{,}3) = 1 > 0{,}99$, где r складывается из: $R_{ЦП}$ = 0,04, $R_{СЕТЬ}$ = 0,01, $R_{ЖД} \le$ 0,000001, $R_{ОП}$=0,08.

3.3.2.4 Оценка удобства использования

Методы оценивания подразделяются на формализованные и неформализованные. Формализованные методы оценки включают в себя оценку скорости выполнения заданий, точности выбранных ответов, количества реализованных жестов.

Скорость. Оценивается количество времени, затраченное на выполнение каждого задания. При этом время отсчитывается от начала до момента нажатия кнопки перехода к следующему заданию.

Точность. Оценивается количеством правильных ответов, полученных во время выполнения заданий. В случае если ответ дан неверно или он верный лишь частично, следует проанализировать причину неточного ответа, вызвана ли она проблемами с жестовым управлением или неправильным пониманием задания, текст которого может быть неоднозначно истолкован.

Количество жестов. Количество жестов должно быть оптимальным, то есть обеспечивать основные функции приложения, и при этом жестов не должно быть слишком много, во избежание проблем с запоминанием жестов пользователем. Жесты должны включать в себя взаимодействие с одним объектом, взаимодействие с группой объектов, взаимодействие с дополнительной информацией (показать/убрать), включать в себя масштабирование и/или переход между уровнями. При этом количество жестов должно быть минимальным (в целях лёгкости запоминания) и достаточным (чтобы обеспечить весь необходимый функционал интерфейса). Управление включает в себя три основные функции: дополнительная информация, смена топологии, взаимодействие с одним объектом или с группой. Таким образом, должно быть не менее трёх жестов управления.

Однако смена топологии и работа с контекстной информацией подразумевают по два противоположных действия: смена топологии с физической на логическую и обратно и показать/убрать дополнительную информацию. Для удобства восприятия эти жесты с привязанными

противоположными функциями должны быть зеркальными по отношению друг к другу. Например, так следующие жесты реализуют данный принцип:

- показать дополнительную информацию – провести тремя пальцами по экрану вправо;

- убрать дополнительную информацию – провести тремя пальцами по экрану влево;

- поменять топологию сети с физической на логическую – свести четыре пальца на экране;

- поменять топологию сети с логической на физическую – развести четыре пальца на экране.

Таким образом, достигается более лёгкое запоминание жестов при достаточном их количестве, однако необходимое количество жестов возрастает до пяти.

Взаимодействие с одним объектом и группой объектов достигается с помощью вызова контекстного меню. Вызов контекстного меню можно реализовать, например, следующим образом:

- долгое нажатие на вершину графа – вызвать контекстное меню.

Масштабирование/переход с одного уровня на другой достигается сменой топологии сети с физической на логическую и обратно:

- свести/развести четыре пальца на экране – смена топологии сети.

Взаимодействие с дополнительной информацией реализовано с помощью листания тремя пальцами вправо/влево на экране – показать/убрать дополнительную информацию.

Неформализованные методы оценки могут быть представлены опросами, экспертными оценками, фокус-группами, сбором мнений и т.п. В диссертационном исследовании было принято решение остановиться на опросе испытуемых после проведения тестирования с помощью заранее заготовленных вопросов, на которые можно дать числовой ответ по шкале от 0 до 5.

Таким образом, алгоритм оценивания удобства использования интерфейсов взаимодействия состоял из следующих шагов.

1. При обработке данных, полученных при тестировании, должна проводиться оценка результатов тестирования по формальным признакам.

1.1. Количество реализованных жестов должно быть не менее трёх и не более пяти. Этот параметр оценивается ещё на этапе разработки.

1.2. После прохождения тестирования испытуемыми рассчитать скорость прохождения каждого задания тестирования. Время прохождения теста измеряется в секундах. Расчёты должны сравниваться с другим интерфейсом: традиционным либо предыдущей версией сенсорного/виртуальной реальности. Результаты сравниваются по верхнему квантилю Q3, при этом разница в 3 секунды не учитывается. На основе полученных результатов выясняется, какие жесты и при выполнении каких задач вызвали затруднения, и какой интерфейс оказался более удачным: новый или предыдущий.

1.3. После прохождения тестирования испытуемыми сравнить ответы на каждое задание, данные испытуемыми, с правильными. На основе полученных результатов выясняется, какие жесты и при выполнении каких задач вызвали затруднения.

2. При завершении тестирования среди испытуемых должен проводиться опрос с возможностью выбора количественного показателя от 0 до 5. Если достигается результат в 80% баллов, то интерфейс считается пригодным для использования оператором. Если достигается результат в 50% баллов, то считается возможным доработать интерфейс. Если результат в 50% баллов не достигается, то интерфейс считается неудачным.

Таким образом, алгоритм оценки удобства использования интерфейса будет включать в себя следующие шаги.

1. Определить, какие показатели наиболее важны для интерфейса: в данном случае, скорость и точность.

2. Определить, который из показателей наиболее критичен (так как для разных моделей это могут быть разные показатели).

3. Определить, в каких величинах измеряются показатели, так как показатель может трактоваться по-разному разными исследователями. В данном случае, скорость измеряется как время, за которое испытуемый проходит задание. Точность измеряется как процент ошибок, которые допущены испытуемым.

4. Определить допустимый порог измеряемых показателей, если оценивается интерфейс сам по себе. Либо, если один интерфейс сравнивается с другим, определить порог, начиная с которого различия считаются значительными. Например, для скорости прохождения заданий такой порог может быть установлен в 3 секунды.

5. Определить, какие у интерфейса существуют типы взаимодействия.

6. Дать испытуемым задания на каждый тип взаимодействия.

7. Измерить время, затрачиваемое испытуемыми на каждое задание.

8. Измерить количество ошибок, которое допускают испытуемые при выполнении каждого задания.

9. Определить среднее время выполнения каждого типа задания.

10. Определить среднее количество ошибок при выполнении каждого типа задания.

11. Сравнить полученные значения с допустимым порогом. Если сравнивались два интерфейса, то определить, у какого интерфейса показатели оказались лучше: у разрабатываемого интерфейса или у того, с которым сравнивают.

12. Определить типы взаимодействия, которые вызвали затруднения у испытуемых (т.е. перешли допустимый порог).

Применение данного алгоритма (рисунок 34) позволяет определить эффективность взаимодействия с человеко-компьютерным интерфейсом.

Рисунок 34 – Блок-схема алгоритма оценивания удобства использования

Для получения оценки удобства использования были проведены тесты:

1. Test 1 A – исправленный интерфейс и сенсорный экран;
2. Test 2 A – уязвимый интерфейс и сенсорный экран;
3. Test 1 B – исправленный интерфейс и клавиатура и мышь;
4. Test 2 B – уязвимый интерфейс и клавиатура и мышь.

Каждый тест содержал 13 заданий на взаимодействие с элементами модели визуализации. Задания были разделены на три группы (выбор, взаимодействие с меню и действие) и их комбинации (выбор + меню, выбор + действие). Вопросы, связанные с выбором, предполагали возможность выбора одного или нескольких элементов визуализации, взаимодействие с меню предполагало взаимодействие с опциями выпадающего списка, действие осуществлялось посредством более сложных жестов.

В результате проведения эксперимента были получены распределения скорости и точности выполнения заданий. Результаты были оценены по трем параметрам: по максимуму распределения, по верхнему квантилю (75% лучших показателей) и по среднему значению. Для этого графики распределения были визуализированы в виде ящиков с усами (рисунки 35, 36), в которых показатели определялись следующим образом.

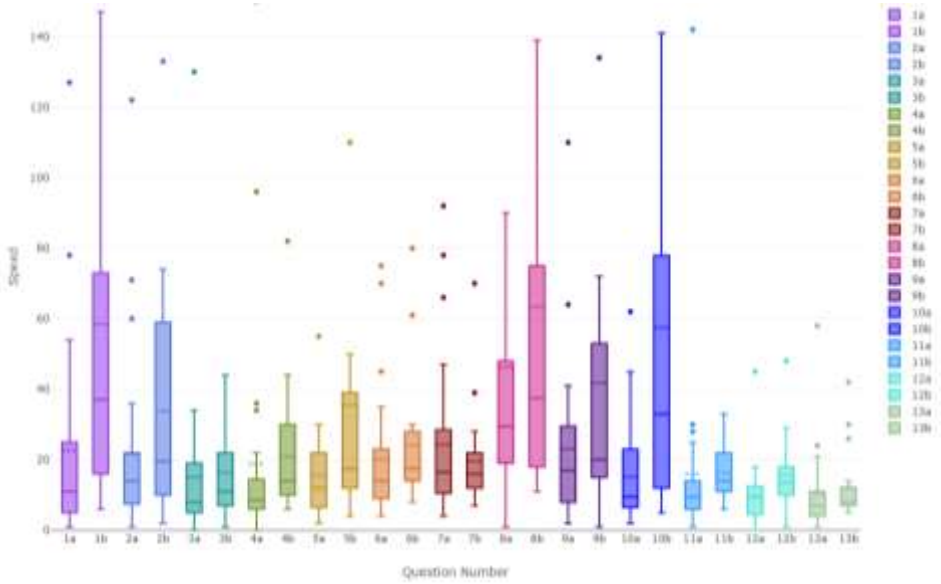


Рисунок 35 – Тест 1: скорость выполнения заданий в секундах для исправленного интерфейса, где *a* – сенсорные экраны, *b* – традиционный интерфейс

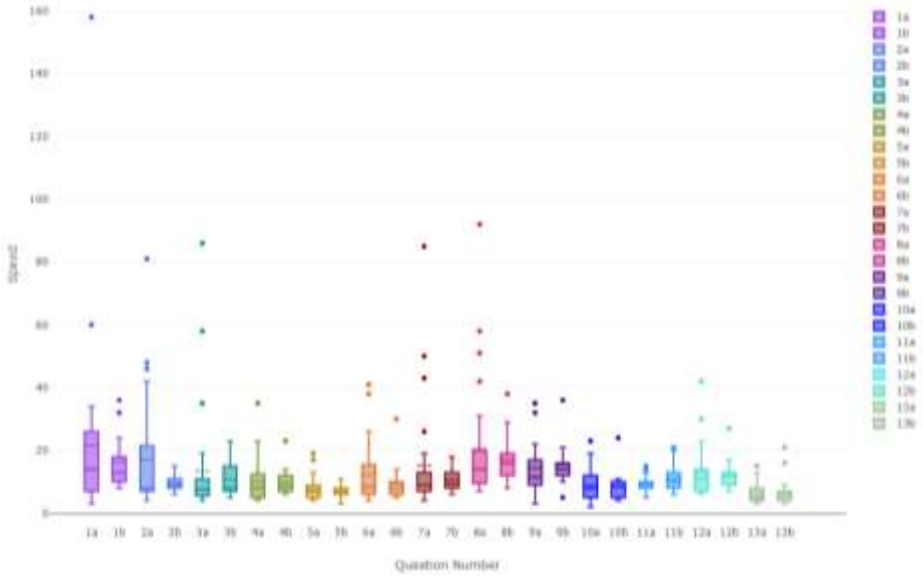


Рисунок 36 – Тест 2: скорость выполнения заданий в секундах для уязвимого интерфейса, где *a* – сенсорные экраны, *b* – традиционный интерфейс

(1) Скорость ответа на вопросы определялась как разность между началом ответа (когда в соответствующем окне появлялся текст задания) и окончанием ответа (когда пользователь нажимал на кнопку перехода к следующему вопросу).

(2) Время на графике измеряется в секундах.

На каждом графике сравниваются показатели тестов с сенсорным интерфейсом (номера вопросов с буквой “a”: 1a, 2a, и т.д.) и тестов с традиционным кнопочным интерфейсом (номера вопросов с буквой “b”: 1b, 2b, и т.д.). Одинаковые вопросы показаны одним и тем же цветом.

Ниже представлены таблицы 12 и 13 сравнения параметров временных распределений при выполнении заданий тестов. Сравниваются максимум распределения (Upper fence), верхний квантиль (Q3) и среднее значение (mean).

Сравнение выполнялось по следующим принципам.

1. В случае если разница во времени составляет **более 3 секунд в пользу сенсорных экранов** (время выполнения задания на сенсорном экране меньше, чем на традиционном интерфейсе), результаты признаются хорошими, и ячейки выделяются синим цветом.

2. В случае если разница во времени составляет **менее 3 секунд в пользу любого из двух тестов**, результаты признаются одинаковыми, и ячейки выделяются жёлтым цветом.

3. Если разница во времени составляет **более 3 секунд в пользу традиционного интерфейса** (время выполнения задания на сенсорном экране больше, чем на традиционном интерфейсе), результаты признаются неудовлетворительными, и ячейки выделяются красным цветом.

Итоговый результат отмечается определённым цветом в ячейке с номером задания по принципу мажоритарного элемента.

1. Если большинство параметров одинаковые, общий результат признаётся приемлемым и отмечается жёлтым цветом.

2. Если большинство параметров «хорошие», общий результат признаётся хорошим и отмечается синим цветом.

3. Если существует даже один «неудовлетворительный» параметр, результаты не могут признаться хорошими и считаются: (1) приемлемыми, если «неудовлетворительный» параметр один, (2) «неудовлетворительными», если «неудовлетворительных» параметров два или три.

Таблица 12 – Эффективность групп заданий в тестах 1 (исправленный интерфейс)

№ задания	Тип взаимодействия	Тест 1 Сенсорный		Тест 1 Традиционный	
		Параметр	Значение	Параметр	Значение
1	выбор	Upper fence	54	Upper fence	147
		Q3	25	Q3	73
		Mean	22.68	Mean	58.5
2	меню	Upper fence	36	Upper fence	74
		Q3	22	Q3	59
		Mean	21.68	Mean	33.83
3	выбор + меню	Upper fence	34	Upper fence	44
		Q3	19	Q3	22
		Mean	15.07	Mean	16.44
4	выбор + меню	Upper fence	22	Upper fence	44
		Q3	14.5	Q3	30
		Mean	18.82	Mean	21
5	выбор + меню	Upper fence	30	Upper fence	50
		Q3	22	Q3	39
		Mean	15.5	Mean	35.5
6	выбор + меню	Upper fence	35	Upper fence	30
		Q3	23	Q3	28
		Mean	20	Mean	24.17
7	действие	Upper fence	47	Upper fence	28
		Q3	28.5	Q3	22
		Mean	24.29	Mean	19.56
8	выбор + меню	Upper fence	90	Upper fence	139
		Q3	48	Q3	75
		Mean	46.29	Mean	63.33
9	выбор + меню	Upper fence	41	Upper fence	72
		Q3	29.5	Q3	53
		Mean	23.11	Mean	41.78
10	действие	Upper fence	45	Upper fence	141
		Q3	23	Q3	78
		Mean	15.21	Mean	57.5
11	выбор + меню	Upper fence	25	Upper fence	33
		Q3	14	Q3	22
		Mean	16	Mean	16.39
12	выбор + меню	Upper fence	18	Upper fence	29
		Q3	12.5	Q3	18
		Mean	9.75	Mean	15.78
13	действие	Upper fence	21	Upper fence	14
		Q3	11	Q3	12
		Mean	10.48	Mean	12.22

Таблица 13 – Эффективность групп заданий в тестах 2 (уязвимый интерфейс)

№ задания	Тип взаимодействия	Тест 2 Сенсорный		Тест 2 Традиционный	
		Параметр	Значение	Параметр	Значение
1	выбор	Upper fence	34	Upper fence	24
		Q3	26	Q3	18
		Mean	21.68	Mean	16.39
2	меню	Upper fence	42	Upper fence	15
		Q3	21.5	Q3	11
		Mean	17.07	Mean	9.56
3	выбор + меню	Upper fence	19	Upper fence	23
		Q3	11	Q3	15
		Mean	13.42	Mean	10.78
4	выбор + меню	Upper fence	23	Upper fence	14
		Q3	12.5	Q3	12
		Mean	9.93	Mean	9.89
5	выбор + меню	Upper fence	13	Upper fence	11
		Q3	9	Q3	8
		Mean	7.75	Mean	7.06
6	выбор + меню	Upper fence	26	Upper fence	14
		Q3	15.5	Q3	10
		Mean	12.25	Mean	9.67
7	выбор + меню	Upper fence	19	Upper fence	18
		Q3	13	Q3	13
		Mean	15.14	Mean	10.83
8	выбор	Upper fence	31	Upper fence	29
		Q3	20	Q3	19
		Mean	20.43	Mean	17.06
9	выбор + меню	Upper fence	22	Upper fence	21
		Q3	17	Q3	16
		Mean	14.29	Mean	15.44
10	действие	Upper fence	19	Upper fence	11
		Q3	12	Q3	10
		Mean	9.04	Mean	8.5
11	выбор + меню	Upper fence	13	Upper fence	20
		Q3	10	Q3	13
		Mean	9.11	Mean	11.11
12	выбор + меню	Upper fence	23	Upper fence	17
		Q3	14	Q3	13
		Mean	13	Mean	12.11
13	действие	Upper fence	13	Upper fence	9
		Q3	8	Q3	7
		Mean	6.11	Mean	6.89

Для исправленного графа сенсорный интерфейс показал лучший результат почти во всех категориях тестов. Исключение составило задание №7 “Поменять связи у графа”, которое выполнялось сведением/разведением нескольких пальцев по экрану. В остальном результаты лучше, либо сравнимы с традиционным интерфейсом.

Для графа с уязвимостью сенсорный интерфейс показал преимущественно равный результат. Исключение составило задание №1 “ Притянуть и зафиксировать любую точку”, №2 “Увеличить выбранную точку” и №6 “ Выделить все зелёные точки”. В остальном результаты признаны одинаковыми.

Точность на графике отображена таким образом, что ответы с наибольшей точностью имеют величину 0, чем больше отклонение от нуля – тем ниже точность ответа (рисунок 37). Например, вопрос с самой высокой точностью ответа - №13 “Убрать дополнительную информацию на графе”, т.к. не имеет отклонений от нуля ни в одном тестировании. Самым сложным вопросом оказался вопрос №8 (“Зафиксировать три тёмно-серые точки и поменять связи у графа”), т.к. имеется самое большое отклонение от нуля – 4, сразу по двум тестам.

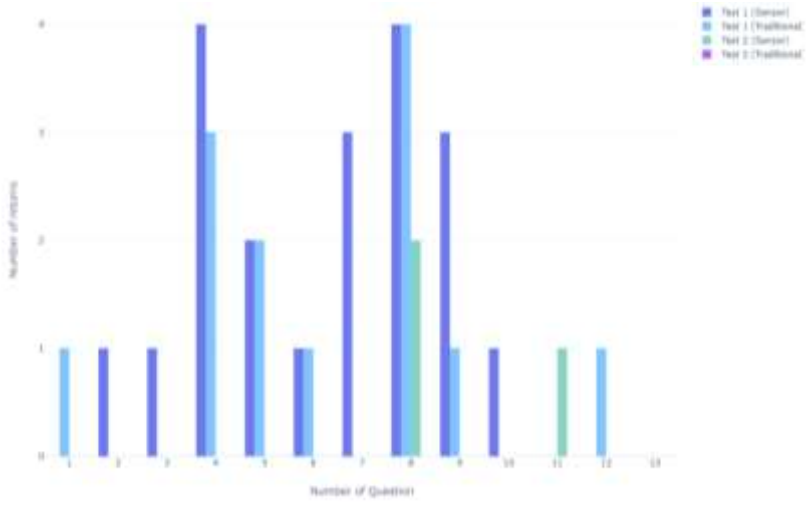


Рисунок 37 – Измерение точности. Синий – Test 1a (сенсорный интерфейс на исправленном графе), голубой – Test 1b (традиционный интерфейс на исправленном графе), бирюзовый – Test 2a (сенсорный интерфейс на уязвимом графе), фиолетовый – Test 2b (традиционный интерфейс на уязвимом графе)

Точность была визуализирована с помощью столбчатой диаграммы, так как количество ошибок при прохождении тестов было небольшим, и применить графики рассеивания было невозможно. Каждый столбец отображает суммарное количество ошибок, полученных всеми испытуемыми.

Результаты точности для сенсорного и традиционного интерфейса приблизительно одинаковы. В тесте 2 (уязвимый интерфейс) практически не возникало ошибок. В тесте 1 (исправленный интерфейс), количество ошибок для обоих интерфейсов сопоставимо. Таким образом, преимущества в точности у определённого интерфейса не выявлено.

Однако можно оценить эффективность модели выше даже при неудовлетворительной точности, если на выполнение заданий затрачено намного меньше времени. В качестве примера представим ситуацию, в которой испытуемые допускают ошибку при выполнении задания №1 с вероятностью 30% на сенсорном интерфейсе и выполняют задания №1 на кнопочном интерфейсе безошибочно. В качестве значения времени возьмём верхний предел выполнения задания по времени в секундах.

Таблица 15 – Пример расчёта эффективности модели, исходя из двух показателей качества

Показатель	Точность, %	Время, с	u(x)
Модель 1	80	54	0,163w
Модель 2	100	147	0,152w
Min	0	5	
Max	100	300	
Цель	Max	Min	

В строке «Цель» таблицы 15 указаны показатели, один из которых должен максимизироваться (целевой показатель), другой должен минимизироваться (ресурсный показатель). Для достижения компромисса между этими двумя показателями можно использовать средневзвешенную суммарную функцию полезности. Функция полезности представлена ниже:

$$u(x) = w_a \cdot u_a(x) + w_t \cdot u_t(x), \qquad (24)$$

где:

– u(x) – общая полезность модели;

– u_a (x) – полезность, измеряемая в точности;

– w_a – важность показателя точности в оценке общей полезности модели;

– u_t (x) – полезность, измеряемая по времени;

– w_t – важность времени выполнения заданий в общей оценке модели.

Чтобы рассчитать общую полезность по формуле (24), нужно нормировать полезности моделей по точности и скорости в соответствии с измерениями, представленными в таблице 15. Для моделей 1 и 2 были рассчитаны значения полезности по точности и времени:

$$u_a(1) = \frac{80}{100-0} = 0{,}8,$$
$$u_t(1) = \frac{300-54}{300-5} = 0{,}83,$$
$$u_a(2) = \frac{100}{100-0} = 1,$$
$$u_t(2) = \frac{300-147}{300-5} = 0{,}52.$$

Если подставить полученные значения в формулу, то получится:

$$u(x)_1 = 0{,}8w_a + 0{,}83w_t,$$
$$u(x)_2 = w_a + 0{,}52w_t.$$

Дальнейшая оценка будет зависеть от того, какой показатель важнее – точность или скорость, т.е. от процентного соотношения wa и wt соответственно. Если эти показатели равны, т.е. $w_a = w_t = 0{,}5$, то $u(x)_1 = 0{,}163w$, $u(x)_2 = 0{,}152w$, и, таким образом, эффективность модели с сенсорным интерфейсом несколько выше.

Однако, из данного примера видно, что расчёт эффективности может зависеть от ситуации, то есть, от того, какие из исследуемых параметров более важны для модели.

3.4 Сравнение предложенной методики оценивания человеко-компьютерных интерфейсов с существующими аналогами

Поскольку не было выявлено полных аналогов предлагаемой методики, в данном разделе обсуждаются наиболее близкие методики и технологии. Существующие системы оценки интерфейсов оценивают показатели удобства использования. Такие методики включают в себя эвристический анализ, пользовательские исследования, также может применяться количественная оценка формальных показателей (таких как точность и скорость) [135, 140, 141].

Ближайшие аналоги можно разделить на две группы: (1) оценивающие уязвимость (такие как общая система оценки уязвимостей CVSS и система оценки уязвимостей Microsoft) и (2) оценивающие удобство использования (например, система оценивания интерфейса Нильсена [135, 140, 141]). Системы из первой группы не оценивают интерфейс, в то время как системы из второй группы, оценивающие интерфейс, оценивают только удобство использования, но не защищённость.

Как было показано в разделе 1.3, существующие исследования, посвящённые повышению уровня защищённости современных типов человеко-компьютерного интерфейса, сосредоточены на закрытии какой-либо конкретной уязвимости или предотвращении какой-либо конкретной угрозы или ряда угроз, в то время как оценивание уровня защищённости интерфейса от этих угроз не предполагается.

Особенность предлагаемой методики состоит в том, что она оценивает как уровень защищённости интерфейса, так и удобство его использования после принятия мер безопасности. Таким образом, достигается повышение осведомленности оператора об уровне защищённости используемой системы человеко-компьютерного взаимодействия без существенного снижения удобства использования такой системы после принятия контрмер.

Для сравнения показателей *защищённости* были отобраны методики, наиболее часто используемые для оценивания интерфейсов, а также система

оценивания уязвимостей CVSS, так как она обладает открытой документацией и потому доступна для анализа.

В таблице 16 были использованы следующие обозначения: «+» показывает, что система полностью удовлетворяет заданным условиям; «-» означает, что заданное условие не удовлетворяется рассматриваемой системой. Например, система CVSS оценивает уязвимости, однако не отображает уровень защищённости человеко-компьютерного интерфейса. При этом CVSS не учитывает параметры, характерные для человеко-компьютерных интерфейсов, такие как урон самому оператору от атаки на интерфейс. Эвристический подход Нильсена оценивает удобство использования интерфейса, однако не учитывает возможные уязвимости.

По результатам экспериментов, показатели *защищённости* предлагаемой методики превзошли показатели рассматриваемых ближайших аналогов.

Таблица 16 – Сравнение предлагаемой методики с ближайшими аналогами

Параметры	**Реализация предлагаемой методики**	**CVSS**	**Nielsen Heuristics**
Оценивает уязвимости	+	+	-
Оценивает защищённость интерфейса	+	-	-
Учитывает урон, наносимый пользователю	+	-	-
Учитывает ущерб, наносимый данным пользователя	+	+	-
Оценивает удобство использования	+	-	+

Также, результаты экспериментов показали, что предлагаемая методика расширяет знания оператора или разработчика об уязвимостях компьютерной системы на величину $VULN_I$. Таким образом, по значениям показателей защищённости предлагаемая методика не уступает существующим методикам оценивания уязвимостей и удовлетворяет предъявляемым требованиям.

Таким образом, результаты экспериментальной проверки и сравнение с ближайшими аналогами предлагаемой методики и программного прототипа показали, что поставленная задача поиска интерфейса с минимальной уязвимостью выполняется, а значение дополнительных свойств оперативности и ресурсопотребления соответствует предъявляемым требованиям.

3.5 Предложения по практическому использованию результатов исследования

Цель настоящего диссертационного исследования заключается в создании моделей и алгоритмов, а также основанной на них методики, которая позволяет оценивать общий уровень защищённости человеко-компьютерного интерфейса. За счёт этого предполагается достичь повышения безопасности компьютерной системы, так как предложенная методика расширяет возможности оценивания уязвимости компьютерной системы: общепризнанные методики оценивания уязвимостей оценивают уровень защищённости сети, в то время как предлагаемая система, в дополнение к ним, оценивает уровень защищённости взаимодействия пользователя с компьютерной системой.

Для достижения поставленной цели система оценивания защищённости человеко-компьютерного интерфейса должна успешно решать ряд задач, представленных ниже:

1) получение данных о реальном состоянии уровня защищённости человеко-компьютерного интерфейса;

2) проведение обоснованного анализа безопасности защищаемого человеко-компьютерного интерфейса;

3) принятие эффективных решений для обеспечения безопасности человеко-компьютерного интерфейса;

4) своевременное снижение и/или устранение рисков для безопасности человеко-компьютерного интерфейса.

Данная методика должна лечь в основу создания системы оценивания защищённости человеко-компьютерного интерфейса и защиты интерфейса от атак, которая: (1) основывается на общепринятых показателях, используемых в таких системах, как, например, CVSS; (2) учитывает показатели, специфичные для человеко-компьютерных интерфейсов; (3) позволяет оценить уровень защищённости человеко-компьютерного интерфейса, основанного на технологиях сенсорных экранов и виртуальной реальности.

В качестве пользователей данных систем выступают операторы приложений, взаимодействие с которыми должно быть защищено (например, банковских приложений или приложений информационной и компьютерной безопасности), а также разработчики данных приложений, предназначенных для компьютерных систем, основанных на технологиях сенсорных экранов и виртуальной реальности. Применение данной методики позволит уменьшить количество успешных атак на современные типы человеко-компьютерных интерфейсов и, следовательно, уменьшить финансовые риски и потери.

Результаты данной работы могут быть применены также для оценивания защищённости человеко-компьютерного взаимодействия с устройствами, использующими технологии сенсорных экранов и виртуальной реальности и принадлежащими коммерческим и государственным организациям. Поскольку новые типы интерфейсов начинают повсеместно внедряться, применение данной методики может быть особенно актуально. Кроме того, в различных организациях всегда остаётся риск инсайдерских атак.

Рядовые пользователи с помощью данной методики также смогут оценить, насколько защищено их взаимодействие с интерфейсами. Это может быть важно, так как в настоящее время широко используются портативные компьютерные системы, такие как смартфоны и планшеты.

Кроме того, пользователями данной методики могут выступать исследователи, выполняющие проекты по заказу Министерства Обороны РФ, так как защищённость взаимодействия с интерфейсом компьютерной

системы может быть актуальна для военной сферы. По этой причине предлагаемая методика может представлять интерес для научно-исследовательских институтов города Санкт-Петербург.

Таким образом, учитывая всё вышеописанное, планируется дальнейшее взаимодействие и работа в сотрудничестве с различными организациями Санкт-Петербурга для внедрения результатов, полученных при проведении данного исследования, что позволит ввести предлагаемую методику в повседневную практику.

3.6 Выводы по главе 3

1. Предложена методика оценивания человеко-компьютерного взаимодействия оператора с новыми типами пользовательских интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности. Данная методика использует разработанные модели интерфейсов и уязвимостей этих интерфейсов, а также применяет разработанные алгоритмы оценивания уязвимостей и уровня защищённости интерфейса. Данная методика оценивания интерфейсов отличается от существующих возможностью оценивания уровня защищённости интерфейса. Особенностью данной методики является также то, что с её помощью можно оценить удобство использования интерфейса после принятия мер по защите интерфейса от возможных угроз.

2. Представлена архитектура, а также разработан программный прототип, реализующий предложенную методику. Прототип содержит все разработанные модели и алгоритмы.

3. Были проведены эксперименты для оценки предлагаемой методики и её программного прототипа. Для определения успешности достижения предъявляемых требований были выбраны свойства оперативности и ресурсопотребления. Результаты, полученные в ходе проведения экспериментов, показали, что предъявляемые требования были удовлетворены.

ЗАКЛЮЧЕНИЕ

В диссертационной работе решена научная задача разработки комплекса моделей, алгоритмов и методики оценивания человеко-компьютерных интерфейсов, повышающих их защищённость, за счет чего была достигнута поставленная в исследовании цель – повышение защищённости человеко-компьютерных интерфейсов. Решенная задача имеет важное значение для развития теории взаимодействия оператора и компьютерных систем, использующих технологии сенсорных экранов и виртуальной реальности, и разработке алгоритмов оценивания уязвимостей и оценивания общего уровня защищённости человеко-компьютерных интерфейсов, а также методик, использующих алгоритмы оценивания человеко-компьютерных интерфейсов.

Итоги исследования включают нижеперечисленные научные результаты:

1. Разработана аналитическая модель уязвимостей человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, которая использует теоретико-множественный подход к описанию параметров этих уязвимостей. Для работы с моделью уязвимостей интерфейсов были также разработаны следующие модели человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности: (1) концептуальная модель человеко-компьютерного интерфейса, учитывающая когнитивный аппарат оператора и (2) теоретико-множественная модель интерфейса, учитывающая параметры безопасности. Теоретико-множественный подход позволяет описать человеко-компьютерный интерфейс и его уязвимости как набор данных, которые могут быть использованы при выполнении алгоритмов оценивания защищённости человеко-компьютерного интерфейса.

2. Разработан алгоритм оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов

и виртуальной реальности, по комплексному показателю, включающий расчёт показателя его уязвимости. Основным отличием от ближайших аналогов является учёт показателей, характерных для человеко-компьютерного интерфейса, таких как канал восприятия и урон, наносимый оператору.

3. Разработана методика оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, в которую входят оценивание общего уровня защищённости человеко-компьютерного интерфейса и оценивание удобства его использования. Особенностью данной методики является повышение осведомленности оператора об уровне защищённости используемой компьютерной системы.

4. Разработана архитектура и программная реализация системы оценивания защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности, использующая разработанную методику. Проведённые эксперименты показали, что требования к показателям оперативности и ресурсопотребления выполняются, и методика позволяет выполнить задачу поиска интерфейса с минимальной уязвимостью.

Также были даны **рекомендации** по практическому использованию научных результатов: результаты, полученные при проведении данного диссертационного исследования, могут быть использованы для разработки систем оценивания защищённости человеко-компьютерных интерфейсов, учитывающих удобство использования после принятия контрмер. Таким образом, задача, поставленная в диссертационном исследовании, была успешно решена.

Использование предложенной методики позволит повысить защищённость взаимодействия оператора с компьютерными системами, основанными на технологиях сенсорных экранов и виртуальной реальности.

Перспективы дальнейшей разработки темы состоят в улучшении алгоритмов оценивания защищённости интерфейсов, уточнении показателей уязвимости, а также изучении новых типов интерфейсов.

Полученные результаты соответствуют специальности 2.3.6. «Методы и системы защиты информации, информационная безопасность».

СПИСОК ЛИТЕРАТУРЫ

1. Большая Российская Энциклопедия [Электронный ресурс] // Интерфейс: [сайт]. [2019]. URL: https://bigenc.ru/technology_and_technique/text/4426494/ (дата обращения: 03.11.2021).

2. Котенко И. В., Коломеец М.В., Жернова К.Н., Чечулин А.А. Визуальная аналитика для информационной безопасности: области применения, задачи и модели визуализации // Вопросы кибербезопасности. – 2021. – Т. 44. – №. 4. – С. 2-15.

3. Papetti S. Design and Perceptual Investigations of AudioTactile Interactions //Proceedings of AIA DAGA. – 2013.

4. Котенко И., Левшун Д., Чечулин А., Ушаков И., Красов А. Комплексный подход к обеспечению безопасности киберфизических систем на системе микроконтроллеров // Вопросы кибербезопасности. – 2018. – Т. 3. – № 27. – С. 29-38.

5. Best D., Bohn S., Love D., Wynne A., Pike W. Real-time visualization of network behaviors for situational awareness // Proceedings of the seventh international symposium on visualization for cyber security. – ACM, 2010. – С. 79-90.

6. Choi H., Lee H., Kim H. Fast detection and visualization of network attacks on parallel coordinates // Computers & security. – 2009. – Т. 28. – No. 5. С. 276-288.

7. Котенко И., Степашкин М., Дойникова Е. Анализ защищенности автоматизированных систем с учетом социо-инженерных атак // Проблемы информационной безопасности. Компьютерные системы. – 2011. – № 3. – С.40-57.

8. Дойникова Е., Котенко Д., Котенко И. Реагирование на компьютерные вторжения с использованием графов атак и графов зависимостей сервисов // 21-я научно-техническая конференция «Методы и технические средства

обеспечения безопасности информации». 24 июня – 29 июня 2012 г. Санкт-Петербург. Материалы. Издательство Политехнического университета. – 2012 – С.45-47.

9. Ingols K., Lippmann R., Piwowarski K. Practical attack graph generation for network defense // 2006 22nd Annual Computer Security Applications Conference (ACSAC'06). – IEEE, 2006. – С.121-130.

10. Apple [Электронный ресурс]: Use Multi-Touch gestures on your Mac – Apple Support: [сайт]. [2022]. URL: https://support.apple.com/ en-us/HT204895 (дата обращения: 03.01.2022).

11. Коломеец М. В., Чечулин А. А., Котенко И. В. Обзор методологических примитивов для поэтапного построения модели визуализации данных //Труды СПИИРАН. – 2015. – Т. 42. – №. 5. – С. 232-257.

12. Hutchins E. L., Hollan J. D., Norman D. A. Direct manipulation interfaces //Human-computer interaction. – 1985. – Т. 1. – №. 4. – С. 311-338.

13. Котенко И. В. Коломеец М. В., Комашинский В. И., Бушуев С. Н., Гельфанд А. М. Модель человеко-машинного взаимодействия на основе сенсорных экранов для мониторинга безопасности компьютерных сетей //Региональная информатика (РИ-2018). XVI Санкт-Петербургская. – 2018. – С. 149.

14. Котенко И. В. Коломеец М.В., Бушуев С.Н., Гельфанд А.М. Методы человеко-машинного взаимодействия на основе сенсорных экранов в ситуационных центрах безопасности //Информационные технологии в управлении (ИТУ-2018). – 2018. – С. 554-558.

15. TofAR [Электронный ресурс] // Виртуальная реальность: [сайт]. URL: http://tofar.ru/article/virtualnaya-realnost.htm (дата обращения 10.06.2020).

16. Igor Kotenko, Andrey Chechulin. Fast Network Attack Modeling and Security Evaluation based on Attack Graphs // Journal of Cyber Security and Mobility. – 2014 – Т. 3. – No.1. – С. 27–46.

17. Noel S., Jacobs M. Kalapa P., Jajodia S. Multiple coordinated views for network attack graphs. // Visualization for Computer Security, 2005. (VizSEC 05). IEEE Workshop on. – IEEE, 2005.

18. Kolomeec M., Gonzalez-Granadillo G., Doynikova E., Chechulin A., Kotenko I., Debar H. Choosing Models for Security Metrics Visualization // International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security. – Springer, Cham, 2017. – С. 75-87.

19. Maxim Kolomeets, Andrey Chechulin, Igor Kotenko. Visualization Model for Monitoring of Computer Networks Security Based on the Analogue of Voronoi Diagrams. // The International Cross Domain Conference and Workshop (CD- ARES 2016). August 31- September 2, 2016. Salzburg, Austria. F. Buccafurri et al. (Eds.): CD-ARES 2016, LNCS 9817. – 2016 – С.141–157.

20. Balzer M., Deussen O., Lewerentz C. Voronoi treemaps for the visualization of software metrics. // Proceedings of the 2005 ACM symposium on Software visualization. – ACM, 2005.

21. Котенко И.В., Новикова Е.С. Визуальная аналитика на страже информационной безопасности. // Positive Technologies. PHD 2014. – 2014. – URL: https://vimeo.com/97908798 (дата обращения: 03.11.2021).

22. Сергеев С. Ф. Методологические проблемы человеко-машинного интерфейса //Москва. – 2014. – Т. 16. – С. 19.

23. Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия //СПб.: Символ-Плюс. – 2009.

24. Магазанник В. Человеко-компьютерное взаимодействие. – Litres, 2017.

25. machinelearningmastery.ru [Электронный ресурс] // 3 правила для интерактивной визуализации данных: [сайт]. [2018]. URL: https://www.machinelearningmastery.ru/3-rules-for-interactive-data-visualizations-a-showcase-with-r-and-highcharts-e452f5c37f44/ (дата обращения: 08.12.2020).

26. Bošnjak L., Brumen B. Shoulder surfing experiments: A systematic literature review //Computers & Security. – 2020. – Т. 99. – С. 102023.

27. Albinsson P. A., Zhai S. High precision touch screen interaction //Proceedings of the SIGCHI conference on Human factors in computing systems. – 2003. – С. 105-112.

28. Farhad M., MacKenzie I. S. Evaluating tap-and-drag: A single-handed zooming method //International Conference on Human-Computer Interaction. – Springer, Cham, 2018. – С. 233-246.

29. Sarcar S., Jokinen J. P. P., Oulasvirta A., Wang Z., Silpasuwanchai C., Ren X.. Ability-based optimization of touchscreen interactions //IEEE Pervasive Computing. – 2018. – Т. 17. – №. 1. – С. 15-26.

30. Gordon M. L., Zhai S. Touchscreen haptic augmentation effects on tapping, drag and drop, and path following //Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. – 2019. – С. 1-12.

31. Garg S. Comparative Studies of Gesture-Based and Sensor-Based Input Methods for Mobile User Interfaces. – 2021.

32. Goguey A., Casiez G., Cockburn A, Gutwin C. Storyboard-based empirical modeling of touch interface performance //Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. – 2018. – С. 1-12.

33. Liu D., Cuervo E, Pistol V, Scudellari R, Co L.P. Screenpass: Secure password entry on touchscreen devices //Proceeding of the 11th annual international conference on Mobile systems, applications, and services. – 2013. – С. 291-304.

34. Johansen U. A. Keystroke dynamics on a device with touch screen : дис. – 2012.

35. Ahmad N., Szymkowiak A., Campbell P. Keystroke dynamics in the pre-touchscreen era //Frontiers in human neuroscience. – 2013. – Т. 7. – С. 835.

36. Milgram P., Kishino F. A taxonomy of mixed reality visual displays //IEICE TRANSACTIONS on Information and Systems. – 1994. – Т. 77. – №. 12. – С. 1321-1329.

37. Mann S., Furness T., Yuan Y., Iorio J., Wang Z. All reality: Virtual, augmented, mixed (x), mediated (x, y), and multimediated reality //arXiv preprint arXiv:1804.08386. – 2018.

38. Azuma R., Bailot Y., Behringer R., Feiner S.J., MacIntyre B. Recent advances in augmented reality //IEEE computer graphics and applications. – 2001. – Т. 21. – №. 6. – С. 34-47.

39. Van Krevelen D. W. F., Poelman R. A survey of augmented reality technologies, applications and limitations //International journal of virtual reality. – 2010. – Т. 9. – №. 2. – С. 1-20.

40. Rabbi I., Ullah S. A survey on augmented reality challenges and tracking //Acta graphica: znanstveni časopis za tiskarstvo i grafičke komunikacije. – 2013. – Т. 24. – №. 1-2. – С. 29-46.

41. Billinghurst M., Clark A., Lee G. A survey of augmented reality. – 2015.

42. Sicaru I. A., Ciocianu C. G., Boiangiu C. A. A survey on augmented reality //Journal of Information Systems & Operations Management. – 2017. – С. 263-279.

43. Moline J. Virtual reality for health care: a survey //Studies in health technology and informatics. – 1997. – С. 3-34.

44. Psotka J. Immersive training systems: Virtual reality and education and training //Instructional science. – 1995. – Т. 23. – №. 5-6. – С. 405-431.

45. Roldán J. J., Roldan J.J., Pena-Tapia1 E., Garzon-Ramos D., de Leon J., Garzon M., del Cerro J., Barrientos A. Multi-robot systems, virtual reality and ROS: developing a new generation of operator interfaces //Robot Operating System (ROS). – Springer, Cham, 2019. – С. 29-64.

46. Erra U., Malandrino D., Pepe L. Virtual reality interfaces for interacting with three-dimensional graphs //International Journal of Human–Computer Interaction. – 2019. – Т. 35. – №. 1. – С. 75-88.

47. Ott R., Gutiérrez M., Thalmann D., Vexo F. Advanced virtual reality technologies for surveillance and security applications //Proceedings of the 2006

ACM international conference on Virtual reality continuum and its applications. – 2006. – С. 163-170.

48. Anand V. Security approaches for virtual reality transactions : заяв. пат. 15184759 США. – 2017.

49. Yelizarov A., Gamayunov D. Adaptive Security Event Visualization for Continuous Monitoring //UMAP Workshops. – 2013.

50. I.D. Brown. Driver fatigue // Human Factors The Journal of the Human Factors and Ergonomics Society. – 1994. – Т. 36. – №2. – С. 298-314.

51. Гришин О. В., Гришин В. Г., Смирнов С. В. Актиграфия в диагностике засыпания человека-оператора // Сибирский научный медицинский журнал. – 2012. – Т. 32. – №2. – С. 94-98.

52. George C., Khamis M., von Zezschwitz E., Burger M., Alt H.S.F., Hussmann H. Seamless and secure vr: Adapting and evaluating established authentication systems for virtual reality. – NDSS, 2017.

53. Mathis F., Vaniea K., Khamis M. RepliCueAuth: Validating the Use of a lab-based Virtual Reality Setup for Evaluating Authentication Systems //Proceedings of the 39th Annual ACM Conference on Human Factors in Computing Systems (Yokohama, Japan)(CHI'21). ACM, New York, NY, USA. – 2021.

54. Olade I., Fleming C., Liang H. N. BioMove: Biometric User Identification from Human Kinesiological Movements for Virtual Reality Systems //Sensors. – 2020. – Т. 20. – №. 10. – С. 2944.

55. Lu Y., Gao B., Long J., Weng J. Hand Motion with Eyes-free Interaction for Authentication in Virtual Reality //2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW). – IEEE, 2020. – С. 715-716.

56. Li S., Savaliya S., Marino L., Leider A. M., Tappert C. C. Brain signal authentication for human-computer interaction in virtual reality //2019 IEEE International Conference on Computational Science and Engineering (CSE) and

IEEE International Conference on Embedded and Ubiquitous Computing (EUC). – IEEE, 2019. – С. 115-120.

57. Li X., Chen Y., Patibanda R. vrCAPTCHA: Exploring CAPTCHA Designs in Virtual Reality //arXiv preprint arXiv:2102.12313. – 2021.

58. Visoottiviseth V., Phungphat A., Puttawong N., Chantaraumporn P., Haga J. Lord of secure: the virtual reality game for educating network security //2018 seventh ict international student project conference (ict-ispc). – IEEE, 2018. – С. 1-6.

59. Seo J. H., Bruner M., Payne A., Gober N., McMullen D., Chakravorty D.K. Using virtual reality to enforce principles of cybersecurity //The Journal of Computational Science Education. – 2019. – Т. 10. – №. 1.

60. Andrabi S. J., Reiter M. K., Sturton C. Usability of augmented reality for revealing secret messages to users but not their devices //Eleventh Symposium On Usable Privacy and Security ({SOUPS} 2015). – 2015. – С. 89-102.

61. Tijsmans L. Collaborative work with Augmented and Virtual Reality-A secure network connection in Unity.

62. Yarramreddy A., Gromkowski P., Baggili I. Forensic analysis of immersive virtual reality social applications: A primary account //2018 IEEE Security and Privacy Workshops (SPW). – IEEE, 2018. – С. 186-196.

63. Huang Z., Chen D., Wang M. Design and application of intelligent patrol system based on virtual reality //2017 IEEE International Conference on Robotics and Biomimetics (ROBIO). – IEEE, 2017. – С. 1194-1199.

64. Lv Z. Virtual reality in the context of Internet of Things //Neural Computing and Applications. – 2020. – Т. 32. – №. 13. – С. 9593-9602.

65. Lee H., Cha W. C. Virtual reality-based ergonomic modeling and evaluation framework for nuclear power plant operation and control //Sustainability. – 2019. – Т. 11. – №. 9. – С. 2630.

66. ГОСТ Р ИСО/МЭК 15408-1-2008. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки

безопасности информационных технологий. Часть 1. Введение и общая модель. – Введ. 2009-05-25. – М. : Изд-во стандартов.

67. ГОСТ Р ИСО/МЭК 15408-1-2008. Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Функциональные требования безопасности. – Введ. 2009-10-01. – М. : Изд-во стандартов.

68. Sproul J., Ledger S., MacCallum J. A review of digital media guidelines for students with visual light sensitivity //International Journal of Disability, Development and Education. – 2021. – Т. 68. – №. 2. – С. 222-239.

69. South L., Saffo D., Borkin M. A. Detecting and Defending Against Seizure-Inducing GIFs in Social Media //Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. – 2021. – С. 1-17.

70. Angelini M., May T., Santucci G., Schulz HJ. On Quality Indicators for Progressive Visual Analytics //EuroVA@ EuroVis. – 2019. – P. 25-29.

71. Culén A. L., Bratteteig T. Touch-screens and elderly users: a perfect match //Changes. – 2013. – Т. 7. – С. 15.

72. Luo T., Jin X., Ananthanarayanan A., Du W. Touchjacking attacks on web in android, ios, and windows phone //International Symposium on Foundations and Practice of Security. – Springer, Berlin, Heidelberg, 2012. – С. 227-243.

73. Delgado-Santos P., Delgado-Santos P., Stragapede G., Tolosana R., Guest R., Deravi F., Vera-Rodriguez R. A Survey of Privacy Vulnerabilities of Mobile Device Sensors //ACM Computing Surveys (CSUR). – 2022.

74. Miguel-Hurtado O., Stevenage S.V., Bevan C., Guest R. Predicting Sex as a Soft-biometrics from Device Interaction Swipe Festures. // Pattern Recognition Letters. – 2016. – № 79. – С. 44–51.

75. Jain A., Kanhangad V. Gender Recognition in Smartphones using Touchscreen Gestures. // Pattern Recognition Letters. – 2019. – № 125. – С. 604-611.

76. Acien A., Morales A., Fierrez J., Vera-Rodriguez R., Hernandez-Ortega J. Active Detection of Age Groups Based on Touch Interaction. // IET Biom. – 2019 – № 8.1. – С. 101-108.

77. Nguyen T., Roy A., Memon N. Kid on the Phone! Toward Automatic Detection of Children on Mobile Devices. // Computers & Security. – 2019. – № 84. – С. 334–348.

78. Arroyo-Gallego T., Ledesma-Carbayo M. J., Sanchez-Ferro A., Butterworth I., Mendoza C. S., Matarazzo M., Montero P., Lopez-Blanco R., Puertas-Martin V., Trincado R., Giancardo L. Detection of Motor Impairment in Parkinson's Disease Via Mobile Touchscreen Typing. // IEEE Trans. Biomed. Eng. – 2017. – № 64.9. – С. 1994–2002.

79. Ahvanooey M. T., Li Q., Rabbani M., Rajput A.R. A survey on smartphones security: software vulnerabilities, malware, and attacks //arXiv preprint arXiv:2001.09406. – 2020.

80. Lee J., Park S., Kim Y.G., Lee E.K., Jo J. Advanced Authentication Method by Geometric Data Analysis Based on User Behavior and Biometrics for IoT Device with Touchscreen //Electronics. – 2021. – Т. 10. – №. 21. – С. 2583.

81. Aviv A. J., Gibson K., Mossop E., Blaze M., Smith J.M.. Smudge attacks on smartphone touch screens //4th USENIX Workshop on Offensive Technologies (WOOT 10). – 2010.

82. IPhone Fingerprint Sensor Hacked with a Finger Made of Clay at MWC.: [сайт]. [2016]. URL: http://www.techworm.net/2016/02/iphone-fingerprint-sensor-hacked-finger-made-clay-mwc-2016.html (дата обращения 17.03.2022).

83. Ming Z., Visani M., Luqman M.M., Burie J.C. A survey on anti-spoofing methods for facial recognition with rgb cameras of generic consumer devices //Journal of Imaging. – 2020. – Т. 6. – №. 12. – С. 139.

84. Li X., Yan F., Zuo F., Zeng Q., Luo L. Touch Well Before Use: Intuitive and Secure Authentication for IoT Devices. // Proceedings of the 25th Annual International Conference on Mobile Computing and Networking, Cabo San Lucas, Mexico, 21–25 October 2019. – 2019. – С. 1–17.

85. Roesner F., Kohno T., Molnar D. Security and privacy for augmented reality systems //Communications of the ACM. – 2014. – Т. 57. – №. 4. – С. 88-96.

86. Epilepsy Foundation Was Targeted in Mass Strobe Cyberattack. [Электронный ресурс] // The New York Times (nytimes.com): [сайт]. [2019]. URL: https://www.nytimes.com/2019/12/16/us/strobe-attack-epilepsy.html (дата обращения 03.01.2022).

87. Gulhane A., Vyas A., Mitra R., Oruche R., Hoefer G., Valluripally S., Calyam P., Hoque K.A. Security, Privacy and Safety Risk Assessment for Virtual Reality Learning Environment Applications //2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC). – IEEE, 2019. – С. 1-9.

88. Hamed A., Khalek A. A. Acoustic Attacks in the Era of IoT-A Survey //2019 Amity International Conference on Artificial Intelligence (AICAI). – IEEE, 2019. – С. 855-858.

89. Diehl E. A Threat Analysis for Virtual Reality Media //SMPTE Motion Imaging Journal. – 2019. – Т. 128. – №. 7. – С. 38-44.

90. Yildirim C. Don’t make me sick: investigating the incidence of cybersickness in commercial virtual reality headsets //Virtual Reality. – 2019. – С. 1-9.

91. Weech S., Kenny S., Barnett-Cowan M. Presence and cybersickness in virtual reality are negatively related: a review //Frontiers in psychology. – 2019. – Т. 10. – С. 158.

92. Sevinc V., Berkman M. I. Psychometric evaluation of Simulator Sickness Questionnaire and its variants as a measure of cybersickness in consumer virtual environments //Applied ergonomics. – 2020. – Т. 82. – С. 102958.

93. Dremliuga R., Dremliuga O., Iakovenko A. Virtual Reality: General Issues of Legal Regulation //J. Pol. & L. – 2020. – Т. 13. – С. 75.

94. Casey P., Baggili I., Yarramreddy A. Immersive virtual reality attacks and the human joystick //IEEE Transactions on Dependable and Secure Computing. – 2019.

95. Gonzalez-Franco M., Lanier J. Model of illusions and virtual reality //Frontiers in psychology. – 2017. – T. 8. – C. 1125.

96. Gonzalez-Franco M., Cohn B., Burin D., Ofek E., Maselli A. The self-avatar follower effect in virtual reality //2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). – IEEE, 2020. – C. 18-25.

97. De Guzman J. A., Thilakarathna K., Seneviratne A. Security and privacy approaches in mixed reality: A literature survey //ACM Computing Surveys (CSUR). – 2019. – T. 52. – №. 6. – C. 1-37.

98. Reilly D., Salimian M., MacKay B., Mathiasen N., Edwards W.K., Franz J. SecSpace: prototyping usable privacy and security for mixed reality collaborative environments //Proceedings of the 2014 ACM SIGCHI symposium on Engineering interactive computing systems. – 2014. – C. 273-282.

99. Rafique M. U., Sen-ching S. C. Tracking Attacks on Virtual Reality Systems //IEEE Consumer Electronics Magazine. – 2020. – T. 9. – №. 2. – C. 41-46.

100. Lebeck K., Kohno T., Roesner F. How to safely augment reality: Challenges and directions //Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications. – 2016. – C. 45-50.

101. Lebeck K., Ruth K., Kohno T., Roesner F. Securing augmented reality output //2017 IEEE symposium on security and privacy (SP). – IEEE, 2017. – C. 320-337.

102. Srinivasa R. R., Veluchamy U. P., Bose J. Augmented Reality adaptive web content //2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC). – IEEE, 2016. – C. 107-110.

103. Shrestha P., Saxena N. An offensive and defensive exposition of wearable computing //ACM Computing Surveys (CSUR). – 2017. – T. 50. – №. 6. – C. 1-39.

104. Ahn S., Gorlatova M., Naghizadeh P., Chiang M., Mittal P. Adaptive fog-based output security for augmented reality //Proceedings of the 2018 Morning Workshop on Virtual Reality and Augmented Reality Network. – 2018. – C. 1-6.

105. Valluripally S., Gulhane A., Mitra R., Hoque K. A., Calyam P. Attack trees for security and privacy in social virtual reality learning environments //2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC). – IEEE, 2020. – С. 1-9.

106. George C., Khamis M., Buschek D., Hussmann H. Investigating the third dimension for authentication in immersive virtual reality and in the real world //2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). – IEEE, 2019. – С. 277-285.

107. Mathis F., Williamson J., Vaniea K., Khamis M. RubikAuth: fast and secure authentication in virtual reality //Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems. – 2020. – С. 1-9.

108. Mathis F., Williamson J., Vaniea K., Khamis M. Fast and Secure Authentication in Virtual Reality using Coordinated 3D Manipulation and Pointing //ACM Transactions on Computer-Human Interaction (ToCHI). – 2021. – Т. 28. – №. 1. – С. 1-44.

109. Holland A., Morelli T. Dynamic keypad-digit shuffling for secure pin entry in a virtual world //International Conference on Virtual, Augmented and Mixed Reality. – Springer, Cham, 2018. – С. 102-111.

110. Lantz P., Johansson B., Hell M., Smeets B. Visual cryptography and obfuscation: A use-case for decrypting and deobfuscating information using augmented reality //International Conference on Financial Cryptography and Data Security. – Springer, Berlin, Heidelberg, 2015. – С. 261-273.

111. Mathis F., Fawaz H. I., Khamis M. Knowledge-driven Biometric Authentication in Virtual Reality //Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems. – 2020. – С. 1-10.

112. Голушко Д. А., Затылкин А. В. Алгоритм разграничения доступа операторов автоматизированных рабочих мест к средствам разведки машин управления //Вопросы радиоэлектроники. – 2019. – №. 12. – С. 34-40.

113. Longley D., Shain M. The Data and Computer Security Dictionary of Standards //Concepts, and Terms. – 1990.

114. Krsul I., Spafford E., Tripunitara M. Computer vulnerability analysis //COAST Laboratory, Purdue University, West Lafayette, IN, Technical Report. – 1998.

115. Schou C. Handbook of INFOSEC Terms, Version 2.0 //CD-ROM (Idaho State University & Information Systems Security Organization). – 1996.

116. ГОСТ Р. 50922-2006. Национальный стандарт РФ //Защита информации. Основные термины и определения. – 2006.

117. Котенко, И.В. , Е.В. Дойникова, А.А. Чечулин. Общее перечисление и классификация шаблонов атак (CAPEC): описание и примеры применения // Защита информации. Инсайд. – СПб., 2012. – № 4. – С. 54-66.

118. Common Platform Enumeration [Электронный ресурс] // CPE: [сайт]. [2014]. URL: http://cpe.mitre.org/ (дата обращения 20.01.2022).

119. Zhong Y., Deng W. Towards transferable adversarial attack against deep face recognition //IEEE Transactions on Information Forensics and Security. – 2020. – Т. 16. – С. 1452-1466.

120. Song, L., Mittal, P. Poster: inaudible voice commands. // Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. – 2017. – С. 2583–2585.

121. Dodiya B., Singh U. K., Gupta V. Trend Analysis of the CVE Classes Across CVSS Metrics //International Journal of Computer Applications. – 2021. – Т. 975. – С. 8887.

122. Scarfone K., Mell P. An analysis of CVSS version 2 vulnerability scoring //2009 3rd International Symposium on Empirical Software Engineering and Measurement. – IEEE, 2009. – С. 516-525.

123. First.org [Электронный ресурс] // Common Vulnerability Scoring System: [сайт]. [2022]. URL: http://www.first.org/cvss (дата обращения: 01.02.2022).

124. Андрианов Ю. М., Суббето А. И. Квалиметрия в приборостроении. – Л. : Машиностроение, 1990. – 216 с.

125. Варжапетян А. Г. Квалиметрия : учеб. пособие. – СПб. : СПбГУАП, 2005. – 176 с.

126. Азгальдов Г. Г., Костин А. В., Садовов В. В. Квалиметрия для всех : учеб. пособие. – М. : Информ-Знание. 2012. – 165 с.

127. Баранов С. Н. Разработка и сертификация программного обеспечения для авиационных бортовых систем и оборудования : учеб. пособие. – СПб. : Изд-во ГУАП, 2017. – 175 с.

128. Стандарт ИСО 8402–94. Управление качеством и обеспечение качества – Словарь. – 29 с.

129. Липаев В. В. Тестирование компонентов и комплексов программ : учебник. – М. : Синтег, 2010. – 392 с.

130. Баранов С. Н., Тележкин А. М. Метрическое обеспечение программных разработок // Труды СПИИРАН. – 2014. – Т.5. – № 36. – С. 5-27.

131. Микони С. В., Соколов Б. В., Юсупов Р. М. Квалиметрия моделей и полимодельных комплексов. – 2018.

132. Борисевич А. В. Теория автоматического управления: элементарное введение с применением MATLAB. – Издательство СПбГПУ, 2011.

133. Соколов Б. В., Юсупов Р. М. Концептуальные основы оценивания и анализа качества моделей и полимодальных комплексов // Известия РАН. Теория и системы управления. – 2004. – № 6. – С. 5–16.

134. Brooke J. SUS: a “quick and dirty” usability //Usability evaluation in industry. – 1996. – С. 189.

135. Weichbroth P. Usability of mobile applications: a systematic literature study //IEEE Access. – 2020. – Т. 8. – С. 55563-55577.

136. Schmettow M. Sample size in usability studies //Communications of the ACM. – 2012. – Т. 55. – №. 4. – С. 64-70.

137. Lewis J. R., Sauro J. Usability and user experience: Design and evaluation //Handbook of Human Factors and Ergonomics. – 2021. – С. 972-1015.

138. Sossa J. W. Z. et al. Delphi method in technological foresight studies: an approach to calculating the number of experts and the application of the competence coefficient" k" expert //Biotecnología en el Sector Agropecuario y Agroindustrial: BSAA. – 2017. – Т. 15. – №. 1. – С. 105-115.

139. Johnson B. et al. Are security experts useful? Bayesian Nash equilibria for network security games with limited information //European Symposium on Research in Computer Security. – Springer, Berlin, Heidelberg, 2010. – С. 588-606.

140. Nielsen J. Ten usability heuristics. – 2005.

141. Nielsen J. Usability inspection methods //Conference companion on Human factors in computing systems. – 1994. – С. 413-414.

142. Balzarotti D., Monga M., Sicari S. Assessing the risk of using vulnerable components // Quality of protection: security measurements and metrics, Advances in Information Security 23. – Springer, New York, 2006. – С. 65–77.

143. Основы теории управления в системах военного назначения. Часть 2 / Под ред. А. Ю. Рунеева и И. В. Котенко. – СПб. : ВУС, 2000. – 158 с.

144. Fedorchenko, A. Integrated repository of security information for network security evaluation / A. Fedorchenko, I. Kotenko, A. Chechulin // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. – 2015. – Т. 6. – № 2. – С. 41-57.

145. Федорченко, А. В. Исследование открытых баз уязвимостей и оценка возможности их применения в системах анализа защищенности компьютерных сетей / А. В. Федорченко, А. А. Чечулин, И. В. Котенко // Информационно-управляющие системы. – 2014. – Т. 72 – № 5. – С. 72-79.

146. NIST [Электронный ресурс] // NVD Dashboard: [сайт]. [2022]. URL: https://nvd.nist.gov/general/nvd-dashboard (дата обращения: 16.08.2022).

147. Данелян Т. Я. Формальные методы экспертных оценок //Статистика и экономика. – 2015. – №. 1. – С. 183-187.

Приложение А. Список публикаций соискателя по теме диссертации

Публикации в журналах из перечня рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание учёной степени кандидата наук, на соискание учёной степени доктора наук:

1. Жернова К.Н., Коломеец М.В., Котенко И.В., Чечулин А.А. Применение адаптивного сенсорного интерфейса в приложениях информационной безопасности // Вопросы кибербезопасности. 2020. 1, 35. С. 18-28. DOI: 10.21681/2311-3456-2020-01-18-28
2. Жернова К.Н. Тенденции и проблемы развития естественности человеко-машинных интерфейсов // Информатизация и связь. 2020. №2. С. 84-95. DOI: 10.34219/2078-8320-2020-11-2-84-95
3. Котенко И.В., Коломеец М.В., Жернова К.Н., Чечулин А.А. Визуальная аналитика для информационной безопасности: области применения, задачи и модели визуализации // Вопросы кибербезопасности. 2021. 4(44). С. 2-15. DOI: 10.21681/2311-3456-2021-4-2-15
4. Котенко И.В., Коломеец М.В., Жернова К.Н., Чечулин А.А. Визуальная аналитика для информационной безопасности: оценка эффективности и анализ методов визуализации // Вопросы кибербезопасности. 2021. 6(46). С. 2-15.

Жернова К.Н. Использование интерфейсов виртуальной реальности в области информационной безопасности // Информатизация и связь. 2021. 2. С. 118-127. DOI: 10.34219/2078-8320-2021-12-2-118-127

В зарубежных изданиях, индексируемых в WoS/Scopus:

1. Zhernova Ksenia, Kolomeets Maxim, Kotenko Igor, Chechulin Andrey. Adaptive Touch Interface: Application for Mobile Internet Security // Communications in Computer and Information Science. 2020. pp. 53-72. DOI: 10.1007/978-981-15-9609-4_5

2. Kolomeets Maxim, Chechulin Andrey, Zhernova Ksenia, Kotenko Igor, Gaifulina Diana. Augmented reality for visualizing security data for cybernetic and cyberphysical systems // 2020 28th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), Västerås, Sweden. 2020. DOI: 10.1109/PDP50117.2020.00071

3. Kolomeets Maxim, Zhernova Ksenia, Chechulin Andrey. Unmanned Transport Environment Threats // Proceedings of 15th International Conference on Electromechanics and Robotics "Zavalishin's Readings", Ufa, Russia, 15–18 April 2020 / Smart Innovation, Systems and Technologies. 2020. 187. pp. 395-408. DOI: 10.1007/978-981-15-5580-0_32

4. Zhernova Ksenia, Chechulin Andrey. Overview of Vulnerabilities of Decision Support Interfaces based on Virtual and Augmented Reality Technologies // Proceedings of 5th International Scientific Conference “Intelligent Information Technologies for Industry”, Sochi, Russia, 1 October 2021

В других изданиях:

1. Котенко И.В., Жернова К.Н. Визуальная модель компьютерной сети с поддержкой жестового интерфейса управления // Перспективные направления развития отечественных информационных технологий. 2019 (РИНЦ)

2. Коломеец М.В., Чечулин А.А., Жернова К.Н. Использование графовых алгоритмов для анализа социальных сетей // XI Санкт-Петербургская межрегиональная конференция «Информационная безопасность регионов России» (ИБРР-2019). 2019 (РИНЦ)

3. Жернова К.Н., Коломеец М.В. Когнитивные особенности цветового восприятия пользователями приложений информационной безопасности // XI Санкт-Петербургская межрегиональная конференция «Информационная безопасность регионов России» (ИБРР-2019). 2019 (РИНЦ)

4. Жернова К.Н., Чечулин А.А. Модели и алгоритмы визуализации данных для выявления и противодействия нежелательной информации //

Материалы V Межрегиональной научно-практической конференции: Перспективные направления развития отечественных информационных технологий. 2019. С. 325-327 (РИНЦ)

5. Жернова К.Н., Коломеец М.В., Чечулин А.А. Обзор методов человеко-машинного взаимодействия в системах противодействия сомнительной и нежелательной информации // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2019). VIII Международная научно-техническая и научно-методическая конференция. 2019. С. 449-454 (РИНЦ)

6. Виткова Л.А., Десницкий В.А., Жернова К.Н., Чечулин А.А. Обзор способов человеко-компьютерного взаимодействия для сетевой безопасности // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2019). VIII Международная научно-техническая и научно-методическая конференция. 2019. С. 218-223 (РИНЦ)

7. Коломеец М.В., Жернова К.Н. Визуальный анализ ботов социальной сети в дополненной реальности // XVII Санкт-Петербургская международная конференция «Региональная информатика (РИ-2020)», Санкт-Петербург, Россия, 28-30 октября, 2020.. 2020. 1. С. 141-142 (РИНЦ)

8. Коломеец М.В., Жернова К.Н. Виртуальная реальность в визуальной аналитике графов // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2020). IX Международная научно-техническая и научно-методическая конференция. 26–27 февраля 2020. Санкт-Петербург.. 2020. 1. С. 460-462 (РИНЦ)

9. Израилов К.Е., Жернова К.Н. Интеллектуальные методы классификаций угроз транспортной инфраструктуры умного города // Межрегиональная научно-практическая конференция «Перспективные направления развития отечественных информационных технологий» (ПНРОИТ 2020). Севастополь, Россия, 22 - 26 Сентября.. 2020. С. 160-161 (РИНЦ)

10. Жернова К.Н. Методика оценки моделей визуализации для систем компьютерной безопасности // Межрегиональная научно-практическая конференция «Перспективные направления развития отечественных информационных технологий» (ПНРОИТ 2020). Севастополь, Россия, 22 - 26 Сентября.. 2020. С. 221-222 (РИНЦ)

11. Бахтин Ю. Е., Бушуев С. Н., Гайфулина Д. А., Жернова К. Н., Иванов А. Ю., Комашинский В. И., Котенко И. В. Методика экспериментальной оценки эффективности человеко-компьютерного взаимодействия в визуальной аналитике // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2020). IX Международная научно-техническая и научно-методическая конференция. 26–27 февраля 2020. Санкт-Петербург.. 2020 (РИНЦ)

12. Жернова К.Н., Комашинский Н.А., Котенко И.В. Модели визуального человеко-компьютерного взаимодействия с сетью устройств интернета вещей // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2020). IX Международная научно-техническая и научно-методическая конференция. 26–27 февраля 2020. Санкт-Петербург.. 2020 (РИНЦ)

13. Жернова К.Н., Коломеец М.В. Обзор методик оценки эффективности систем визуальной аналитики // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2020). IX Международная научно-техническая и научно-методическая конференция. 26–27 февраля 2020. Санкт-Петербург.. 2020. 1. С. 463-466 (РИНЦ)

14. Жернова К.Н., Гайфулина Д.А., Иванов А.Ю., Комашинский В.И. Управление данными визуализации мобильной сети с использованием сенсорных экранов // XVII Санкт-Петербургская международная конференция «Региональная информатика (РИ-2020)», Санкт-Петербург, Россия, 28-30 октября, 2020.. 2020 (РИНЦ)

15. Жернова К.Н., Котенко И.В., Чечулин А.А. Методика оценки жестовых интерфейсов для управления компьютерной безопасностью //

Межрегиональная научно-практическая конференция «Перспективные направления развития отечественных информационных технологий» (ПНРОИТ 2020). Севастополь, Россия, 22 - 26 Сентября.. 2020. С. 219-220 (РИНЦ)

16. Жернова К.Н. Методика повышения надёжности оператора при работе с приложениями информационной безопасности на сенсорных экранах // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2021). X Международная научно-техническая и научно-методическая конференция. 26–27 февраля 2021. Санкт-Петербург.. 2021 (РИНЦ)

17. Жернова К.Н., Коломеец М.В. Уязвимости интерфейсов «оператор – искусственный интеллект» в беспилотной транспортной среде // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2021). X Международная научно-техническая и научно-методическая конференция. 26–27 февраля 2021. Санкт-Петербург.. 2021 (РИНЦ)

Регистрация результатов интеллектуальной деятельности:

1. Жернова К.Н., Котенко И.В. Компонент распознавания мультитач жестов для сенсорного экрана // Свидетельство № 2019666461. Зарегистрировано в Реестре программ для ЭВМ 10.12.2019. // 2019

2. Жернова К.Н., Котенко И.В. Сенсорный интерфейс взаимодействия для мониторинга безопасности компьютерной сети // Свидетельство № 2019666536. Зарегистрировано в Реестре программ для ЭВМ 11.12.2019. // 2019

3. Коломеец М.В., Чечулин А.А., Жернова К.Н. Система оценки визуального восприятия пользователя в виртуальной реальности // Свидетельство № 2019664065. Зарегистрировано в Реестре программ для ЭВМ 30.10.2019. // 2019

4. Чечулин А.А., Котенко И.В., Жернова К.Н. Компонент обнаружения многошаговых сетевых атак на основе аналитического

моделирования. // Свидетельство № 2020660703. Зарегистрировано в Реестре программ для ЭВМ 10.09.2020. // 2020

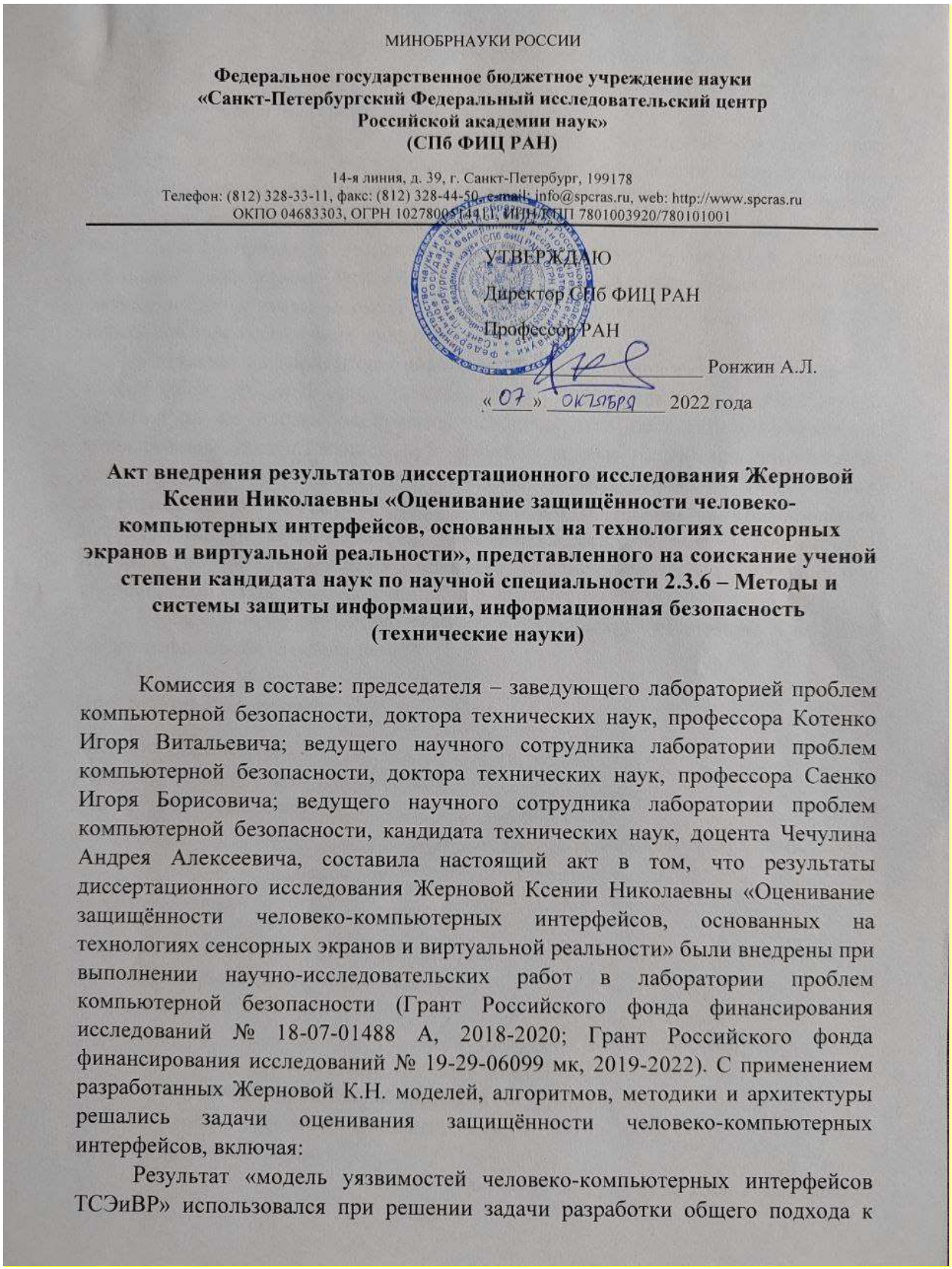
5. Жернова К.Н. Компонент распознавания жестов для управления безопасностью компьютерной сети. Свидетельство № 2020665761. Зарегистрировано в Реестре программ для ЭВМ 01.12.2020. // 2020

6. Жернова К.Н., Коломеец М.В. Компонент реализации круговой диаграммы для отображения данных в виртуальной и дополненной реальности // Свидетельство № 2020660603. Зарегистрировано в Реестре программ для ЭВМ 01.09.2020. // 2020

7. Жернова К.Н., Котенко И.В. Программный комплекс для оценки эффективности человеко-машинного взаимодействия с помощью сенсорных экранов. Свидетельство № 2020665837. Зарегистрировано в Реестре программ для ЭВМ 01.12.2020. // . 2020

8. Коломеец М.В., Чечулин А.А., Жернова К.Н. «Компонент определения усталости оператора на основе данных от датчиков жизнедеятельности». // Свидетельство № 2022612329. Зарегистрировано в Реестре программ для ЭВМ 10.02.2022.

Приложение Б. Акты о внедрении полученных научных результатов



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное учреждение науки
«Санкт-Петербургский Федеральный исследовательский центр
Российской академии наук»
(СПб ФИЦ РАН)

14-я линия, д. 39, г. Санкт-Петербург, 199178
Телефон: (812) 328-33-11, факс: (812) 328-44-50, e-mail: info@spcras.ru, web: http://www.spcras.ru
ОКПО 04683303, ОГРН 1027800514411, ИНН/КПП 7801003920/780101001

УТВЕРЖДАЮ
Директор СПб ФИЦ РАН
Профессор РАН
_____ Ронжин А.Л.
«07» октября 2022 года

Акт внедрения результатов диссертационного исследования Жерновой Ксении Николаевны «Оценивание защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности», представленного на соискание ученой степени кандидата наук по научной специальности 2.3.6 – Методы и системы защиты информации, информационная безопасность (технические науки)

Комиссия в составе: председателя – заведующего лабораторией проблем компьютерной безопасности, доктора технических наук, профессора Котенко Игоря Витальевича; ведущего научного сотрудника лаборатории проблем компьютерной безопасности, доктора технических наук, профессора Саенко Игоря Борисовича; ведущего научного сотрудника лаборатории проблем компьютерной безопасности, кандидата технических наук, доцента Чечулина Андрея Алексеевича, составила настоящий акт в том, что результаты диссертационного исследования Жерновой Ксении Николаевны «Оценивание защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности» были внедрены при выполнении научно-исследовательских работ в лаборатории проблем компьютерной безопасности (Грант Российского фонда финансирования исследований № 18-07-01488 А, 2018-2020; Грант Российского фонда финансирования исследований № 19-29-06099 мк, 2019-2022). С применением разработанных Жерновой К.Н. моделей, алгоритмов, методики и архитектуры решались задачи оценивания защищённости человеко-компьютерных интерфейсов, включая:

Результат «модель уязвимостей человеко-компьютерных интерфейсов ТСЭиВР» использовался при решении задачи разработки общего подхода к

исследований № 19-29-06099 мк «Разработка методов поиска уязвимостей интерфейсов взаимодействия человека с искусственным интеллектом транспортной среды “умного города”».

Результат «методика оценивания защищённости человеко-компьютерных интерфейсов ТСЭиВР» использовался при решении задачи проектирования человеко-компьютерных интерфейсов для систем информационной безопасности, основанных на сенсорных экранах, в проекте: Грант Российского фонда финансирования исследований № 18-07-01488 А «Модели, методы, методики и алгоритмы человеко-машинного взаимодействия для поддержки визуальной аналитики сетевой безопасности критических инфраструктур с использованием сенсорных мультитач-экранов».

Результат «алгоритм оценивания защищённости человеко-компьютерных интерфейсов ТСЭиВР по комплексному показателю» и «обобщённая архитектура комплекса оценивания защищённости человеко-компьютерных интерфейсов» использовались при решении задачи разработки алгоритмов, архитектуры и программных прототипов безопасных человеко-компьютерных интерфейсов в проекте: Грант Российского фонда финансирования исследований № 19-29-06099 мк «Разработка методов поиска уязвимостей интерфейсов взаимодействия человека с искусственным интеллектом транспортной среды “умного города”».

Комиссия отмечает теоретическую, практическую значимость и новизну полученных в работе результатов.

Председатель комиссии:

Заведующий лаборатории
проблем компьютерной безопасности,
доктор технических наук,
профессор — Котенко Игорь Витальевич

Члены комиссии:

Ведущий научный сотрудник лаборатории
проблем компьютерной безопасности,
доктор технических наук,
профессор — Саенко Игорь Борисович

Старший научный сотрудник лаборатории
проблем компьютерной безопасности,
кандидат технических наук, — Виткова Лидия Андреевна

GLORYSTORY:
репутационное агентство

\ ООО «Жасмин»
\ Инн7810880035 КПП 781001001
\ тел.: +7 812 655 0558
\ www.glorystory.ru

«04» октября 2022 г.

АКТ

о внедрении результатов диссертационной работы
Жерновой Ксении Николаевны
«Оценивание защищённости человеко-компьютерных интерфейсов, основанных на технологиях сенсорных экранов и виртуальной реальности»

Комиссия в составе:

1. Мартьянова Жанна Андреевна – Генеральный директор;
2. Намятова Ксения Андреевна – Заместитель генерального директора;
3. Каминская Любовь Олеговна – Менеджер проектов, технический специалист.

Составила настоящий акт о том, что результатов диссертационной работы Жерновой Ксении Николаевны, а именно:

- модель уязвимостей человеко-компьютерных интерфейсов ТСЭиВР;
- алгоритм оценивания защищённости человеко-компьютерных интерфейсов ТСЭиВР по комплексному показателю;
- методика оценивания защищённости человеко-компьютерных интерфейсов ТСЭиВР;
- архитектура и программная реализация системы оценивания уровня защищённости человеко-компьютерных интерфейсов ТСЭиВР

используются при оценке пользовательских интерфейсов для представления результатов заказчикам РА GloryStory. Также результаты диссертационной работы и программные прототипы компонентов архитектуры используются анализа защищённости работы оператора с интерфейсом.

Комиссия отмечает практическую значимость и новизну полученных в работе результатов.

Председатель комиссии:

_____ Мартьянова Ж.А.

Члены комиссии:

_____ Намятова К. А,

_____ Каминская Л. О.

"Жасмин"
М.П. (печать организации)