ПРИМЕНЕНИЕ АДАПТИВНОГО СЕНСОРНОГО ИНТЕРФЕЙСА В ПРИЛОЖЕНИЯХ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Жернова К.Н.[1], Коломеец М.В.[2], Котенко И.В.[3] , Чечулин А.А.[4]

Цель статьи: *разработка подхода к созданию адаптивных интерфейсов на основе сенсорных экранов для приложений информационной безопасности.*

Метод исследования: *системный анализ современных «наилучших практик» для создания жестовых и графических интерфейсов, разработка собственного подхода и его экспериментальная проверка.*

Полученный результат: *предложен подход к формированию сенсорных интерфейсов для визуального анализа в приложениях информационной безопасности, в частности анализа безопасности для устройств Интернета вещей. Предложены алгоритмы адаптации интерфейсов под конкретные задачи информационной безопасности и модели взаимодействия пользователя с интерфейсом на бизнес уровне и на уровне реализации. Представлены результаты экспериментов по восприятию пользователями жестов на примерах визуальной аналитики состояния иерархической централизованной сети встроенных устройств и децентрализованной сенсорной сети.*

Область применения предложенного подхода – создание адаптивных сенсорных интерфейсов, которые могут использоваться для повышения эффективности взаимодействия оператора с приложениями информационной безопасности.

Ключевые слова: *пользовательский интерфейс, графический интерфейс пользователя, сенсорный интерфейс, адаптивные интерфейсы, предиктивные интерфейсы, информационная безопасность, сенсорные экраны.*

DOI:10.21681/2311-3456-2020-01-18-28

Введение

Один из способов анализа безопасности основан на применении методов визуальной аналитики. Визуальная аналитика использует визуализацию данных для обнаружения событий, интерпретации инцидентов и выработки контрмер. В информационной безопасности визуализация данных имеет множество применений: контроль и разграничение доступа в различных моделях безопасности; анализ состояния сетей, образованных устройствами Интернета вещей (IoT-устройствами); анализ метрик безопасности и другие.

Для решения задач такого рода используются различные модели визуализации, которые основаны на традиционных интерфейсах взаимодействия. Тем не менее, усложнение моделей визуализации требует новых форм взаимодействия, которые были бы более удобными для оператора и тем самым повышали бы скорость и качество принятия решений. Одним из таких решений являются интерфейсы на основе сенсорных экранов, однако они обычно не рассматриваются как инструмент взаимодействия аналитика и визуализации данных.

Другой проблемой является противоречие между реализованным функционалом и функционалом, необходимым пользователю для решения конкретных задач. Для решения этой проблемы применяются адаптивные и предиктивные интерфейсы, подстраивающиеся под конкретного пользователя и решаемую им задачу.

В данной статье предлагается подход к формированию пользовательских интерфейсов на основе сенсорных экранов и распознавания жестов оператора и его применение для приложений информационной безопасности. Преимуществом данного подхода, в сравнении с традиционным, является повышение оперативности выполнения аналитики событий информационной безопасности, упрощение взаимодействия с моделями визуализации и повышение качества принятия решений. Для этого предлагаются модели взаимодействия пользователя и алгоритмы адаптации интерфейсов пользователя для решения задач информационной безопасности, связанных с управлением иерархической централизованной сетью встроенных устройств и визуализацией децентрализованной сенсорной сети.

1 Жернова Ксения Николаевна, аспирант, ФГБУН Санкт-Петербургский институт информатики и автоматизации Российской академии наук, г. Санкт-Петербург, Россия. E-mail: zhernova@comsec.spb.ru

2 Коломеец Максим Вадимович, младший научный сотрудник, ФГБУН Санкт-Петербургский институт информатики и автоматизации Российской академии наук, г. Санкт-Петербург, Россия. E-mail: kolomeec@comsec.spb.ru

3 Котенко Игорь Витальевич, доктор технических наук, профессор, главный научный сотрудник и заведующий лабораторией проблем компьютерной безопасности, ФГБУН Санкт-Петербургский институт информатики и автоматизации Российской академии наук, г. Санкт-Петербург, Россия. E-mail: ivkote@comsec.spb.ru

4 Чечулин Андрей Алексеевич, кандидат технических наук, ведущий научный сотрудник, ФГБУН Санкт-Петербургский институт информатики и автоматизации Российской академии наук, г. Санкт-Петербург, Россия. E-mail: chechulin@comsec.spb.ru

Научная новизна данной работы заключается в (1) предложенном комбинированном способе реализации сенсорного интерфейса, основанного на адаптивной подстройке интерфейса под конкретного пользователя и решаемую задачу информационной безопасности и (2) использовании «наилучших практик» в разработке предиктивного жестового интерфейса. Вкладом данной работы является подход, включающий модели и алгоритмы адаптации сенсорного интерфейса под задачи информационной безопасности.

Во второй части статьи проводится обзор релевантных работ из области визуальной аналитики информационной безопасности и интерфейсов взаимодействия. В третьей части предлагается подход к разработке адаптивного интерфейса для информационной безопасности, в том числе модели взаимодействия пользователя и компонентов визуализации, алгоритм адаптации и алгоритм соответствия интерфейса с «наилучшими практиками». В четвертой части описываются эксперименты по восприятию пользователями жестов на примерах визуальной аналитики состояния иерархической централизованной сети встроенных устройств и децентрализованной сенсорной сети. В пятой части рассматриваются достоинства и недостатки предлагаемого подхода и описываются дальнейшие направления работы.

Обзор релевантных подходов

Интерфейсы взаимодействия человека с компьютером тесно связаны с моделями визуализации. Так, в зависимости от решаемой задачи, используются различные типы моделей визуализации, от которых зависит реализация интерфейсов. Например, с помощью графов можно визуализировать компьютерную сеть [1], сканирование портов [2], атаки и их маршруты [3-5], также есть возможность моделирования сценариев атак [6]. При этом методы визуализации могут сочетаться между собой. Например, с помощью графа типа «дерево» можно изобразить физическую иерархическую топологию компьютерной сети, радиальное дерево можно использовать для визуализации атак, диаграммы Корда для одновременного представления физической и логической топологии, а матрицы для отображения доступности сегментов сети для атакующего [7].

Для контроля и разграничения доступа используются способы визуализации, показывающие отношения субъекта и объекта в той или иной модели прав доступа. Так, для дискретных моделей доступа используются матрицы [8]. Для моделей доступа Take-Grant - графы [9], для иерархических моделей управления доступом на основе ролей (RBAC) - TreeMaps [10]. Кроме того, существуют сложные модели визуализации, предназначенные для анализа в комбинированных моделях безопасности. Например, треугольные матрицы [11] применяются для визуализации как матриц, так и деревьев.

Каждая из существующих моделей используется в конкретном случае анализа и управления правами доступа. Чем сложнее модель визуализации и чем сложнее анализ модели безопасности, тем более сложные методы взаимодействия необходимы оператору. Например, в [8] представлена матрица доступа, в которой применяются механизмы фильтрации и группирования субъектов и объектов. Для этого используются классические средства, например, выпадающие списки. В TreeMaps можно фильтровать данные, показав только определенную часть дерева [10]. Для этого, как правило, необходимо при визуализации кликнуть на отображение корня искомого поддерева.

При анализе состояния сетей также используются графы, TreeMaps, матрицы и прочие модели визуализации [7]. Графы наиболее универсальны, и с их помощью можно визуализировать любую структуру сети [11]. TreeMaps подходят для визуализации иерархических сетей [3], а матрицы - для сетей топологии «каждый с каждым» [1]. Также для сетей, которые могут образовывать планарные структуры, используются карты Вороного [12]. Это справедливо, например, для отображения самоорганизующейся сенсорной сети, топология которой была урезана до планарной для сохранения энергии и уменьшения интерференции [12]. Каждый метод имеет свои достоинства и недостатки, поэтому часто их совмещают [13].

Вышеперечисленные модели визуализации применяются во многих областях. Как уже упоминалось – чем сложнее задача и чем больше метрик нужно визуализировать, тем более сложной становится модель визуализации. Например, в [14] представлен подход к объединению моделей визуализации для того, чтобы отображать больше метрик. С другой стороны, чем более сложная модель, тем больше инструментов взаимодействия необходимо оператору. Например, при реализации 3D-моделей нужны инструменты, реализующие поворот и масштабирование. В перегруженных графах масштабирование также необходимо [13]. При этом часто стандартных инструментов может быть недостаточно, и вместо стандартного масштабирования применяется «рыбий глаз» и дисторсия Кортезиана [15]. Всё это ведет к перегруженности интерфейса и усложнению работы оператора-аналитика.

При проектировании визуальных моделей рассматриваются только традиционные способы управления, основанные на использовании монитора, мыши, клавиатуры. Однако визуальную аналитику можно проводить также с помощью планшетов, смартфонов и других сенсорных устройств, поскольку они получают все большее распространение и обеспечивают большую мобильность оператора, например, на производстве.

В работах, посвященных визуальной аналитике информационной безопасности, не рассматриваются подходы к работе на сенсорных экранах и их влияние на процесс визуального анализа и принятия решений информационной безопасности. Приложения анализа безопасности, имеющие сенсорный интерфейс, как правило, не используются. Мы рассмотрели немногие из них (например, «Network Scanner», «Net Analyzer» и «IP Tools») и выяснили, что жесты чаще всего ограничены касанием одного пальца (редко – двух), при этом взаимодействие с моделями визуализации также ограничено нажатием и перетягиванием. Так, интерфейсы многих современных приложений служат простой имитацией взаимодействия с компьютерной мышью.

Для приложений информационной безопасности, ввиду сложности обрабатываемой информации и ком-

плексности и многоуровневости визуализации данных, может быть недостаточно одних стандартных жестов, имитирующих мышь и клавиатуру. Однако жесты также не должны быть слишком сложны для запоминания или неестественны для использования. Для использования таких жестов в интерфейсах безопасности в настоящей работе предлагается подход, включающий модели взаимодействия пользователя и компонентов визуализации, алгоритм адаптации и алгоритм соответствия жестов «наилучшим практикам».

Предлагаемый подход

Для понимания работы интерфейса приложений информационной безопасности следует обратиться к специфике интерфейсов данной предметной области. В информационной безопасности чаще всего присутствуют следующие элементы:

- использование цвета для разграничения по степени опасности (например, зелёный - безопасный, жёлтый - средняя опасность, красный - высшая степень опасности);
- вложенность (вызов дополнительных деталей по требованию, например, показать параметры устройства на графе);
- большое количество данных, которые нужно обработать (например, маршруты трафика);
- ситуационная осведомленность (выдача пользователю актуальных данных с привязкой ко времени и месту, например, при мониторинге сети);
- визуализация обрабатываемых данных (например, представление различных топологий сети).

Перечисленные элементы должны присутствовать в приложениях информационной безопасности, однако их визуальное представление и взаимодействие с ними может видоизменяться. Например, при частых сообщениях о рисках безопасности, помеченных красным цветом, пользователь может утомляться и начать их игнорировать. Данную проблему можно решить с помощью адаптивности интерфейса – через определенные промежутки времени изменять оттенок сообщения о тревоге в пределах красного цвета, например, использовать оттенок «маджента». Пользователь заметит изменения и вновь начнет обращать внимание на сообщения. Таким образом, адаптация интерфейса, то есть подстройка под нужды пользователя на основе его поведения при работе с приложением, также является необходимой частью его проектирования.

Кроме адаптивных интерфейсов, также существуют предиктивные интерфейсы, которые предсказывают, какое действие пользователь совершит в следующий момент, а также какой дизайн интерфейса окажется наиболее удобным для пользователя на основе его поведения. Реализация предиктивного интерфейса возможна, например, на основе нейросетей или сбора статистики действий пользователя. Простой пример такого интерфейса – предиктивный набор текста, при наличии которого система запоминает слова и сочетания слов, чаще всего употребляемые пользователем.

В данном разделе предлагается подход к построению адаптивного интерфейса приложений информационной безопасности, который дает возможность пользователю подстроить систему взаимодействия с приложением с учетом своих предпочтений (“под себя”) и минимизировать необходимость подстраиваться самому.

Для того чтобы понять, какое место интерфейсы занимают в процессе визуального анализа, необходимо определить модель взаимодействия человека и модуля визуализации. На уровне бизнес-логики модель выглядит следующим образом (рис. 1).

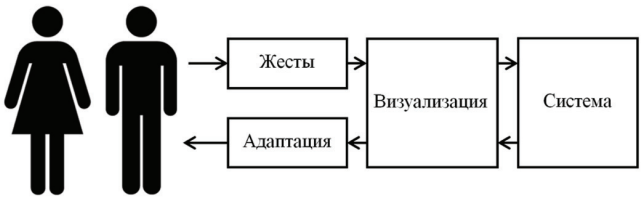

Рис. 1. Модель интерфейса на уровне бизнес-логики

Данная модель предполагает взаимодействие пользователя с модулем визуализации через жесты, система будет обрабатывать команды пользователя, осуществляемые посредством жестов. При этом каждый конкретный пользователь имеет свои особенности, которые система также будет обрабатывать, поэтому в итоге будет формироваться не только визуальное представление, но и результат адаптации под конкретного пользователя.

На уровне реализации модель выглядит, как показано на рис. 2.

Данные поступают от компьютерной системы и загружаются в приложение, далее эти данные обрабатываются, отображаются и прорисовываются, чтобы получилась окончательная визуализация. При этом пользователь воздействует на изображение жестами, которые система обработает, выполнит адаптацию под особенности конкретного пользователя и видоизменит изображение в соответствии с заложенным функционалом.

Как видно, ключевые элементы взаимодействия – это процессы вывода и ввода информации, осуществляющиеся при помощи визуализации и жестов соответственно. Для их адаптации процессы взаимодействия следует рассматривать на двух уровнях (направлениях) взаимодействия:

(1) машины с человеком;

(2) человека с машиной;

Идея алгоритма адаптации состоит в самостоятельном комбинировании приложением жестов и функций, наиболее удобных для пользователя в процессе работы. Алгоритм адаптации интерфейса можно разделить на два этапа.

Первый этап состоит из следующих шагов.

1. Подстройка в процессе инициализации под человека или группу.

Этап инициализации обычно подразумевает вход пользователя в систему, определение уровня подготовки пользователя, выдачу актуальной на данный момент информации. В качестве «наилучших подходов» с этой точки зрения можно упомянуть следующие правила [16]:

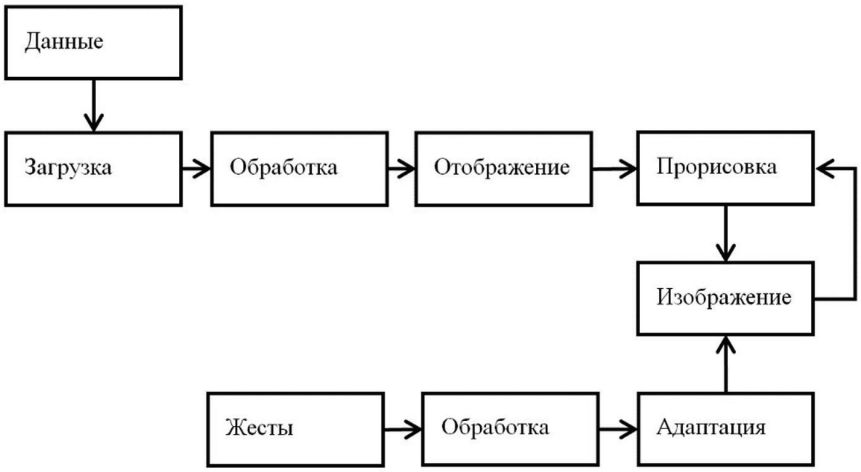

Рис. 2. Модель визуализации с учетом жестового интерфейса

- фокусировка внимания пользователя на том, с чего нужно начинать (какие элементы нужно делать более крупными, выделение заголовков и т.п.);
- правильная группировка элементов из соображений схожести, близости, замыкания, связей между ними и преемственности (группировка похожих функций, отделение функциональных элементов друг от друга пространством и т.п.);
- визуализация первоочередной информации на начальной странице;
- отображение множества способов выполнения задачи (например, возможность входа в систему по почте, логину или номеру телефона);
- предоставление подсказки требуемых действий (обозначение обязательных и необязательных действий, как должен выглядеть результат действий и т.п.);
- помощь пользователям в обходе ошибок (например, отображение доступных вариантов, структурирование текстовых полей, отображение легкого способа выхода из опции, информативные сообщения об ошибках).

2. Глобальная подстройка может помочь преодолеть ряд проблем, обусловленных социокультурными различиями. Например, восприятие семантики цвета может быть разным в разных культурных средах, могут использоваться разные символы и пиктограммы для одних и тех же целей. Таким образом, кроме смены языка приложения, можно реализовать изменение цветовой гаммы интерфейса, порядка расположения окон и т.п. Такая подстройка также необходима при некоторых физических затруднениях, например, расстройствах цветового восприятия [17].

3. Подстройка под функциональные обязанности. Целесообразно позволять пользователю выбирать вид интерфейса в соответствии со своими функциональными обязанностями, чтобы видеть детали, которые необходимы именно ему. Например, программист может запросить больше деталей, связанных с программным кодом. Специалист по информационной безопасности может не обладать навыками программирования, однако ему будут важны детали по безопасности.

4. Подстройка под конкретного человека заключается в индивидуальной манере пользователя работать с приложением. Эта манера может выражаться в специфическом выборе наиболее удобных пользователю моделей визуализации, индивидуальном восприятии жестов сенсорного интерфейса, способе воспринимать и анализировать поступающую информацию. Также сюда входит наиболее часто используемый функционал приложения и запрашиваемая информация.

Второй этап адаптации происходит в процессе работы. Его можно разделить на две составляющие.

1. Подстройка под взаимодействие со стороны компьютера касается визуализации информации, обрабатываемой программой. В качестве подстройки может выступать подбор наиболее комфортных для пользователя визуальных моделей, корректировка выбранной цветовой гаммы, подбор и настройка других сигналов, кроме визуальных, например, звуковых и вибрационных. При необходимости отражается более подробная информация, детали по требованию, подсказки и пр. На рис. 3 на основе представления диаграмм, созданных с помощью D3.js, продемонстрирован набор комплексных моделей визуализации, которые могут применяться для решения задач информационной безопасности.

2. Подстройка под взаимодействие со стороны человека касается того, каким образом человек сообщает свои намерения программно-аппаратной системе. На данном этапе система должна определять наиболее удобные пользователю жесты для определенных функций. Также она должна подстраиваться под исполнение жеста конкретного человека (например, различается время нажатия, пользователь может начать делать один жест, потом передумать и закончить другим).

Особенностью сенсорных интерфейсов является взаимодействие через жесты. Жестовый интерфейс, как и графический, должен подчиняться принципу прямого взаимодействия [18], т.е. используемые жесты должны быть интуитивно понятны пользователю. Пример

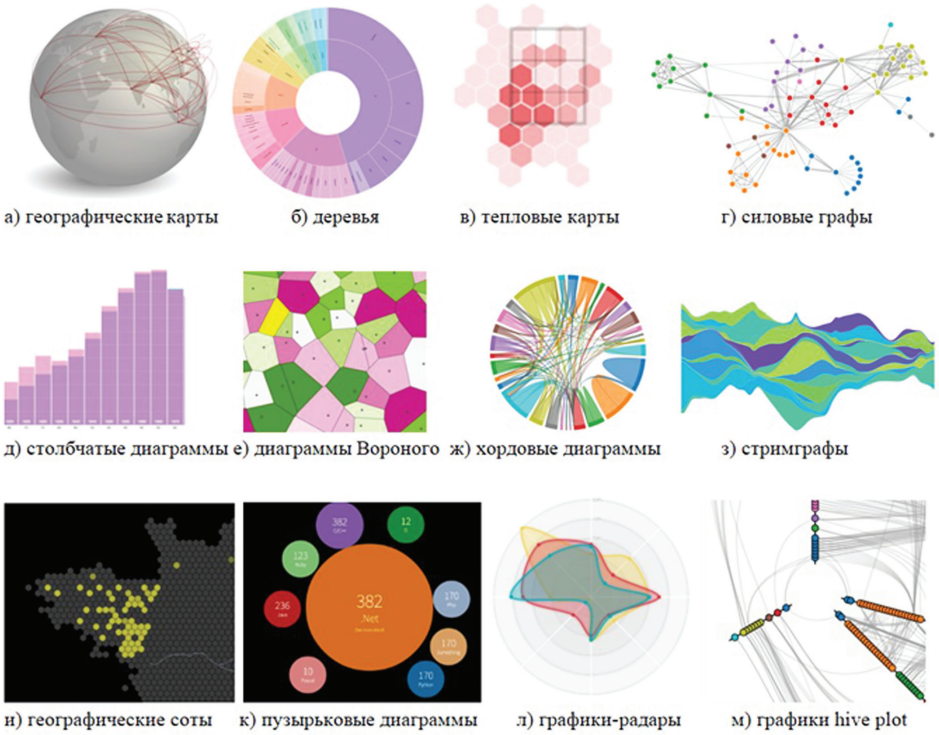

Рис. 3. Примеры диаграмм, созданных с помощью D3.js, которые могут использоваться в информационной безопасности

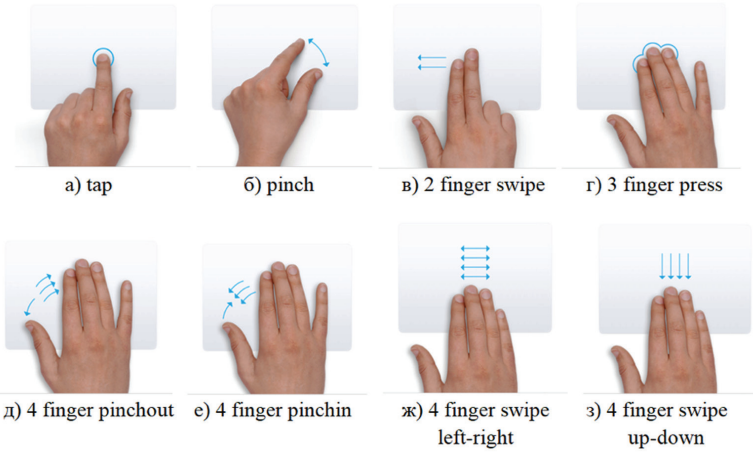

Рис. 4. Примеры простых и сложных жестов для сенсорных интерфейсов[19]

использования интуитивных жестов показан на рис. 4.

Для улучшения жестов предлагается алгоритм, позволяющий создавать жесты в соответствии со своими «наилучшими подходами» [20-22]:

(1) адаптация под мобильное устройство (изменение ширины страницы, размера текста и рисунков при изменении разрешения экрана, появление возможности листания либо прокручивания);

(2) создание удобных для жестов графических элементов (крупные кнопки, прокручивающиеся элементы, изображения высокого разрешения, достаточное расстояние между интерактивными элементами);

(3) использование стандартных жестов (нажатие, двойное нажатие, перетаскивание, листание, вращение, удержание и т.д.);

(4) использование интуитивно подходящих для функции жестов;

(5) отказ от традиционных жестов мыши (например, наведение);

(6) создание элементов интерфейса, которые не будут перекрываться руками пользователя.

Применение

Предлагаемые модели человеко-компьютерного взаимодействия были разработаны в качестве программного прототипа веб-приложения. Прототип был выполнен на языке JavaScript с применением языка разметки HTML5 и библиотек D3.js, hammer.js и свободно распространяемого пакета Bootstrap. Проект состоит из двух компонентов: компонента визуализации и компонента человеко-компьютерного взаимодействия.

Визуализация представляет собой силовой граф, сформированный из заранее загруженных данных. Всего использовалось три набора данных.

Набор 1 сформирован для визуализации имитационного моделирования децентрализованной сенсорной сети без усечения до планарности. В эксперименте использовались данные о симуляции децентрализованной сенсорной сети, состоящей из автономных устройств. В рамках моделирования учитывались следующие параметры устройств: заряд аккумулятора, уровень освещенности и звука. Некоторые из них находились вне помещения. Каждое устройство имеет критический уровень, рассчитанный на основе критичности активов [14], которые были расположены в этой области. Таким образом, потеря датчика означала бы потерю контроля над этим активом. Поскольку устройства являются автономными, они разряжаются, но их можно заряжать с помощью солнечных батарей.

Наборы 2 и 3 сформированы для визуализации имитационного моделирования иерархической централизованной сети комплексной системы безопасности, содержащей встроенные устройства [23]. Встроенные устройства оснащены набором сенсоров, представляющие собой датчики движения, считыватель RFID, датчик горючих газов, датчик температуры, влажности и освещенности. Встроенные устройства подсоединялись к концентратору, осуществляющему сбор, нормализацию и предварительную обработку получаемых данных. Концентраторы подключались к серверу, который хранил, обрабатывал и анализировал сообщения безопасности от устройств и состояние этих устройств.

В процессе анализа данных имелась возможность взаимодействия с визуализированной информацией посредством жестов на сенсорном экране.

В прототипе реализованы следующие жесты:

- притягивание ближайшей вершины графа (устройства) и вызов контекстного меню для этой вершины при касании пальцем, выбор опции контекстного меню повторным касанием, причем выделение отдельных вершин и групп вершин реализовано через контекстное меню;
- перемещение трех пальцев влево/вправо – вызов/скрытие дополнительной информации (показать/скрыть MAC-адреса, уровень заряда, количество переданных сообщений и т.д.);
- касание четырех пальцев – изменение фильтрации (отобразить цвет вершины как тип устройства, уровень заряда устройства, количество переданных сообщений, количество принятых сообщений и т.д.);
- сведение/разведение пяти пальцев – изменение связей графа (показать, как физически связаны устройства, их маршруты трафика).

Жесты были изначально закреплены за определенными функциями, которые выполняет приложение. На странице приложения справа от графа располагается пояснение о соответствии жестов функциям.

В качестве теста данного прототипа был предложен ряд заданий, основанных на доступных способах человеко-компьютерного взаимодействия.

Перечислим эти задания для набора данных 1:

(1) зафиксировать устройства с определенными MAC-адресами (MAC-адреса скрыты и показываются по определенному жесту);

(2) выделить разряженные устройства (значения «высокий уровень заряда», «устройство почти разряжено», «устройство разряжено и отключилось» – задаются с помощью цвета);

(3) выделить определённый тип устройств (тип устройства задаётся цветом);

(4) выделить устройства, не подключенные к самоорганизующейся сети (вершина без ребер);

(5) выделить почти разряженные устройства с высокой критичностью (параметр «устройство почти разряжено» задается цветом, критичность актива задается размером вершины).

Перечислим эти задания для наборов данных 2 и 3:

(1) зафиксировать все выключенные устройства (включенные устройства или выключенные – задается цветом);

(2) выделить все концентраторы (тип устройства задается цветом);

(3) выделить все RFID-сканнеры и датчики дыма, исходя из цвета вершины (тип датчика задается цветом);

(4) выделить концентраторы, на которые пришло наибольшее количество сообщений (чем больше сообщений, тем крупнее вершина);

(5) выделить устройства, которые сгенерировали наибольшее количество сообщений (чем больше сообщений, тем крупнее вершина).

Прототип был запущен через браузер на компьютере, имеющем сенсорный дисплей (рис. 5, рис. 6). На рис. 5 отображен граф, который сформирован на основе моделирования децентрализованной сенсорной сети без усечения до планарности. Вершины графа – это автономные устройства. Цвет показывает тип устройства. Сеть является самоорганизующейся, объединенные между собой устройства имеют связи, не присоединённые устройства не имеют связей.

На рис. 6 отображен силовой граф, образованный при моделировании интегрированной системы безопасности иерархической централизованной сети. Вершины графа – это датчики (белые и желтые), встроенные устройства (фиолетовые), концентраторы (зеленые) и сервер (синий).

При этом управление осуществлялось через дисплей посредством жестов.

Проверка выполнялась следующим образом:

(1) пользователь вставал перед сенсорным дисплеем;

(2) запускался проверяемый прототип;

DOI:10.21681/2311-3456-2020-01-18-28

(3) для начала работы следовало коснуться кнопки «Начало», и в текстовом окне появлялся первый вопрос теста;

(4) управление вопросами осуществлялось кнопками «Назад» и «Вперед»;

(5) во время выполнения очередного задания требовалось взаимодействовать с компонентами визуализации;

(6) после выполнения последнего задания следовало коснуться кнопки «Конец», это действие инициировало загрузку текстового файла с логами выполнения заданий на компьютер;

(7) далее собранные логи анализировались на предмет времени, затрачиваемого на каждое задание, а также качества выполнения заданий (правильность выполнения).

Проверка прототипа показала, что в начале теста интервал между выполняемыми заданиями был больше. Это означает, что испытуемым требуется время, чтобы понять и запомнить предустановленные жесты, если их нельзя задать самостоятельно перед началом работы. В приложениях информационной безопасности, как правило, нет возможности настроить взаимодействие с приложением. Кроме того, некоторые жесты могут не подходить под конкретную модель визуализации метрик безопасности, т.е. отсутствует интуитивность понимания конкретного жеста.

При реализации интерфейса с возможностью предварительной настройки жестов способы взаимодействия с приложением запоминаются пользователем гораздо быстрее, так как он устанавливает их самостоятельно, в соответствии со своим собственным пониманием об удобстве взаимодействия.

Эксперименты показали, что представленный подход позволяет быстрее осуществлять аналитику самоорганизующейся сенсорной сети и иерархической сети. Жесты дают возможность быстро и интуитивно переключаться между метриками, фиксировать интересующие вершины или группы вершин, и переключаться между представлениями графа. Таким образом, улучшается качество и повышается скорость принятия решений при управлении мобильными сетями. Интуитивность жестов позволяет запомнить больше команд, что дает возможность анализировать большее количество метрик, так как между ними становится проще переключаться. Там образом в процессе управления мобильной сетью появляется больше используемой информации для принятия решений.

Предполагается в дальнейшем реализовать адаптивный интерфейс на основе сбора статистики действий пользователя, например, какие жесты для каких функций он использует чаще всего.

Анализ достоинств и недостатков предлагаемого подхода

Настоящие исследования сосредоточены на жестах сенсорных экранов как способе улучшить взаимодействие между пользователем и системами информационной безопасности. Дальнейшая работа предполагает разработку методики создания адаптивных интерфейсов кибербезопасности для сенсорных экранов.

Выделим следующие достоинства предлагаемого подхода.

1. Сохраняются все предыдущие настройки под конкретного пользователя.

2. Настроенный на конкретного пользователя интерфейс становится более удобным для этого пользователя, следовательно, работа с программным средством также происходит более оперативно и с меньшим количеством ошибок.

3. Создание адаптивного жестового интерфейса позволит пользователю самому привязывать определенные удобные ему жесты к существующим функциям приложения.

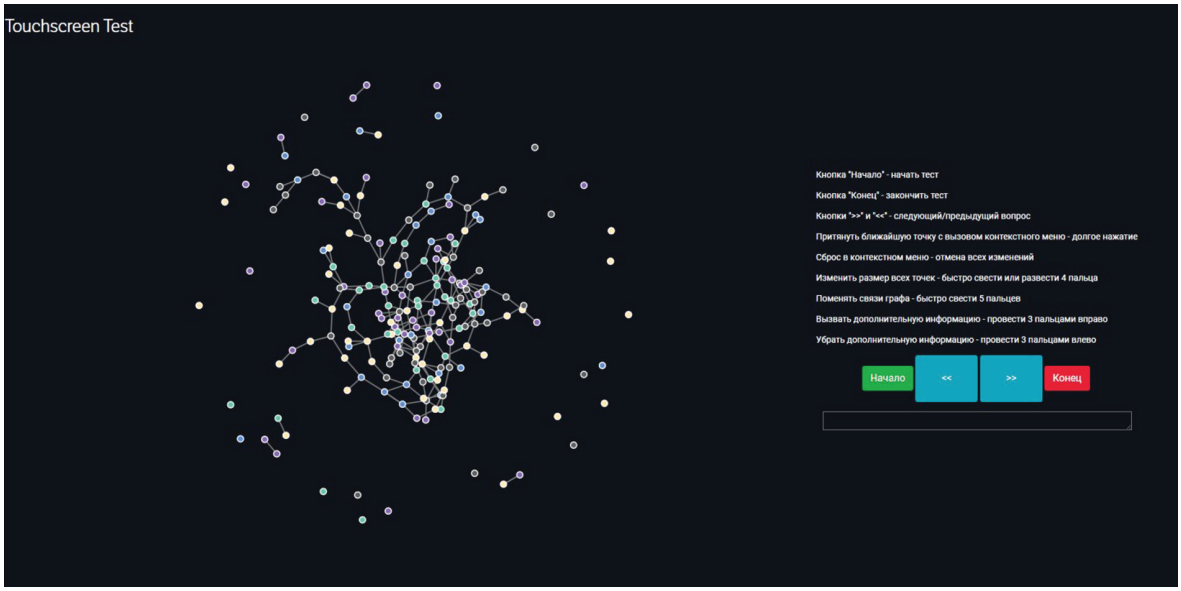

Рис. 5. Внешний вид реализованного веб-приложения при использовании набора данных 1

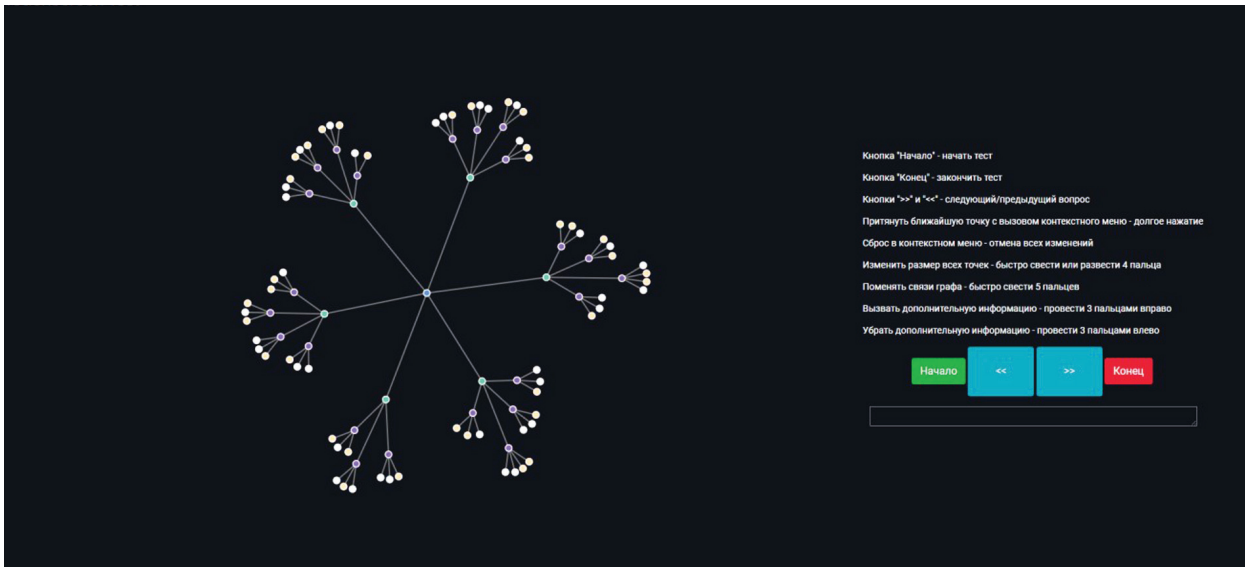

Рис. 6. Внешний вид реализованного веб-приложения при использовании набора данных 2.

4. Такое улучшение позволит повысить скорость обучения интерфейсу нового приложения, а также повысить скорость и эффективность дальнейшей работы оператора с приложением информационной безопасности.

Основными возможными недостатками подхода являются следующие:

1. Другому пользователю будет сложно начинать работу на том же устройстве. В случае работы нескольких людей по очереди за одним устройством данный подход будет скорее недостатком. Однако чаще всего у каждого работника имеется свое индивидуальное рабочее место.

2. Потребуется некоторое время, пока приложение наберет необходимую статистику для адаптации под конкретного пользователя.

С учетом описанных выше недостатков, последующая работа будет включать в себя исследование нескольких важных решений: какие жесты целесообразно назначать заранее, какие из них требуют подстройки, с какими жестами выполнять привязанные к ним функции в конце жеста, с какими – в начале, и какие жесты целесообразно использовать для визуального отображения процесса выполнения функции.

Предложенная модель адаптации может применяться для контроля и разграничения доступа, где система будет подбирать наиболее адекватные ситуации и особенностям пользователя модели визуализации, позволять сортировать колонки и строки матриц удобными для пользователя жестами, масштабировать деревья [24]. Также модель применима для контроля над самоорганизующимися сенсорными сетями, где, в соответствии с ситуацией, будет приниматься решение об отображении сети с помощью графа, TreeMaps, карт Вороного или какой-либо их комбинации, а также будет осуществляться подбор жестов, в наибольшей степени соответствующих выбранным моделям визуализации [25].

В целом подход позволяет увеличить оперативность процессов принятия решений и повысить их качество при настройке сетей мобильных устройств. Например, при анализе сетей, использование жестов позволяет быстрее и более интуитивно переключаться между метриками, фиксировать интересуемые вершины или группы вершин, и переключаться между представлениями графа. Также жесты можно использовать при управлении контролем доступа (например, при управлении разрешений между мобильными устройствами) и оценке рисков (например, при оценке риска и стоимости потери устройства). Отдельно стоит отметить ценность жестов при использовании на планшетах и мобильных устройствах, что востребовано на производстве – когда специалисту необходимо настраивать сети мобильных устройств в полевых условиях. Таким образом, подход также расширяет возможности использования визуальной аналитики для ситуаций, когда использование персональных компьютеров затруднительно.

Заключение

В статье предложен подход к человеко-компьютерному взаимодействию с интерфейсами приложений информационной безопасности на основе сенсорных экранов. Представлены модели взаимодействия пользователя и компонентов визуализации, алгоритм адаптации и алгоритм соответствия интерфейса «наилучшим практикам». Методики, модели, алгоритмы и архитектуры, выбранные на основе существующих «наилучших практик» создания жестовых интерфейсов, могут быть применены для создания адаптивного сенсорного интерфейса приложений информационной безопасности. При этом в предложенном подходе адаптивность интерфейса достигается с помощью следующих процедур:

- подстройки в процессе инициализации под человека или группу;
- подстройки под социокультурные особенности пользователя;

- подстройки под функциональные обязанности пользователя;
- подстройки под индивидуальные физические особенности пользователя.

Кроме того, адаптация происходит как на уровне вывода информации от компьютера, так и на уровне ввода информации в компьютер пользователем.

Основой для реализации жестового интерфейса послужили открытые библиотеки D3.js (для визуализации), hammer.js (для реализации мультитач-жестов), а также открытый набор библиотек и шаблонов Bootstrap (для оформления внешнего вида web-приложения). В качестве стенда выступал дисплей с сенсорным экраном с возможностью расположить его горизонтально.

На реализованном программно-аппаратном стенде проведены эксперименты по восприятию пользователями жестов на примерах визуальной аналитики состояния иерархической централизованной сети встроенных устройств и децентрализованной сенсорной сети. Основным результатом эксперимента служит то, что с помощью представленного подхода аналитика самоорганизующейся сенсорной сети и иерархической сети осуществляется быстрее, что приводит к улучшению качества и повышению скорости принятия решений при управлении мобильными сетями.

Методология, предложенная в данной статье, может быть использована для создания новых моделей взаимодействия с сенсорным интерфейсом в процессе оценки рисков. Также подход к разработке адаптивного интерфейса может быть использован в образовательных и исследовательских целях в области информационной безопасности, визуализации данных и человеко-компьютерного взаимодействия.

Дальнейшие исследования будут направлены на изучение естественности жестов на сенсорных экранах в восприятии пользователей, а также изучение наилучшего соответствия жестов визуальному отображению метрик информационной безопасности.

Рецензент: *Молдовян Александр Андреевич, доктор технических наук, профессор, заведующий отделом проблем информационной безопасности ФГБУН Санкт-Петербургский институт информатики и автоматизации Российской академии наук, Санкт-Петербург, Россия. E-mail: maal305@yandex.ru*

Работа выполнена при частичной финансовой поддержке РФФИ (проект № 18-07-01488-а) и бюджетной темы 0073-2019-0002.

Литература

1. Котенко И., Левшун Д., Чечулин А., Ушаков И., Красов А. Комплексный подход к обеспечению безопасности киберфизических систем на системе микроконтроллеров // Вопросы кибербезопасности. 2018. № 3 (27). С.29-38. DOI: 10.21681/2311-3456-2018-3-29-38
2. 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. P. 79-90.
3. Choi H., Lee H., Kim H. Fast detection and visualization of network attacks on parallel coordinates // Computers & security. 2009. Vol. 28. No. 5. P. 276-288.
4. 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.
5. 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. С.45-47.
6. Ingols K., Lippmann R., Piwowarski K. Practical attack graph generation for network defense // 2006 22nd Annual Computer Security Applications Conference (ACSAC'06). IEEE, 2006. P.121-130.
7. Коломеец М., Чечулин А., Котенко И. Обзор методологических примитивов для поэтапного построения модели визуализации данных // Труды СПИИРАН. 2015. Вып. 42. С.232-257.
8. Heitzmann A., Palazzi B., Papamanthou C., Tamassia R. Effective visualization of file system access-control // International Workshop on Visualization for Computer Security. Springer, Berlin, Heidelberg, 2008. P. 18-25.
9. Bishop M. Conspiracy and information flow in the take-grant protection model // Journal of Computer Security. 1996. Vol. 4. No. 4. P. 331-359.
10. Kim D., Ray I., France R., Li N. Modeling role-based access control using parameterized UML models // International Conference on Fundamental Approaches to Software Engineering. Springer, Berlin, Heidelberg, 2004. P. 180-193.
11. Kolomeets M., Chechulin A., Kotenko I., Saenko I. Access Control Visualization Using Triangular Matrices // 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP). IEEE, 2019. С. 348-355.
12. Kolomeets M., Chechulin A., Kotenko I., Strecker M. Voronoi Maps for Planar Sensor Networks Visualization // International Symposium on Mobile Internet Security. Springer, Singapore, 2017. P. 96-109.
13. Roberts J. Guest editor's introduction: special issue on coordinated and multiple views in exploratory visualization // Information Visualization. 2003. Vol. 2. No. 4. P. 199-200.
14. Коломеец М., Чечулин А., Дойникова Е., Котенко И. Методика визуализации метрик кибербезопасности // Изв. вузов. Приборостроение, Т.61, № 10, 2018, С.873-880.

15. Sarkar M., Brown M. Graphical fisheye views // Communications of the ACM. 1994. Vol. 37. No. 12. P. 73-83.
16. Kolenda N. Psychology & Business [Электронный ресурс]: A List of UX/UI Best Practices for Websites / Режим доступа: https://www.nickkolenda.com/user-experience/# (дата обращения 05.06.2019)
17. Ananto B., Sari R., Harwahyu R. Color transformation for color blind compensation on augmented reality system // 2011 International Conference on User Science and Engineering (i-USEr). IEEE, 2011. P. 129-134.
18. Hutchins E., Hollan J., Norman D. Direct manipulation interfaces // Human-computer interaction. 1985. Vol. 1. No. 4. P. 311-338.
19. Apple [Электронный ресурс]: Use Multi-Touch gestures on your Mac – Apple Support / Режим доступа: https://support.apple.com/en-us/HT204895 (дата обращения 05.06.2019).
20. Apple Developer [Электронный ресурс]: UI Design Do's and Don'ts – Apple Developer / Режим доступа: https://developer.apple.com/design/tips/ (дата обращения 02.06.2019).
21. Apple Developer [Электронный ресурс]: Gestures – User Interaction – iOS – Human Interface Guidelines – Apple Developer / Режим доступа: https://developer.apple.com/design/human-interface-guidelines/ios/user-interaction/gestures/ (дата обращения 05.06.2019).
22. World Usability Congress [Электронный ресурс]: Touch Screen Usability Best Practices When Designing Automation User Interfaces (UI) – World Usability Congress / Режим доступа: https://worldusabilitycongress.com/touch-screen-usability-best-practices-when-designing-automation-user-interfaces-ui/ (дата обращения 05.06.2019).
23. Desnitsky V., Levshun D., Chechulin A., Kotenko I. Design Technique for Secure Embedded Devices: Application for Creation of Integrated Cyber-Physical Security System // JoWUA. 2016. Vol. 7. No. 2. P. 60-80.
24. Милославская Н., Толстой А., Бирюков А. Визуализация информации при управлении информационной безопасностью информационной инфраструктуры организации // Научная визуализация. 2014. Т. 6. №. 2. С. 74-91.
25. Боронин П., Кучерявый А. Интернет вещей как новая концепция развития сетей связи // Информационные технологии и телекоммуникации. 2014. №. 3. С. 7.

USE OF ADAPTIVE TOUCH INTERFACE IN INFORMATION SECURITY APPLICATIONS

Zhernova K.N.[5], Kolomeec M.V.[6], Kotenko I.V.[7], Chechulin A.A.[8]

The purpose of the article: *development of an approach to creating adaptive interfaces based on touch screens in information security applications.*

Research method: *analysis of modern "best practices" for creating gestural and graphical interfaces, development of our own approach and its experimental verification.*

The result obtained: *an approach to the formation of adaptive touch interfaces for visual analysis of IoT security is proposed. Algorithms for adapting interfaces for specific tasks of information security and models of user interaction with the interface at the business level and implementation level are proposed. The results of experiments on the perception of gestures by users on the examples of visual analytics of the state of a hierarchical centralized network of embedded devices and a decentralized sensor network are presented.*

The area of use of the proposed approach is the creation of adaptive touch interfaces that can be used to increase the efficiency of operator interaction with information security applications.

Keywords: *user interface, graphical user interface, touch interface, adaptive interfaces, predictive interfaces, information security, touch screens.*

References

5 Kseniia Zhernova, Ph.D. student at St. Petersburg Institute for Informatics and Automation of Russian Academy of Science, St. Petersburg, Russia. E-mail: zhernova@comsec.spb.ru

6 Maxim Kolomeec, Junior Researchir fellow at St. Petersburg Institute for Informatics and Automation of Russian Academy of Science, St. Petersburg, Russia. E-mail: kolomeec@comsec.spb.ru

7 Igor Kotenko, Dr.Sc. , Professor, Head of Laboratory of Computer Security Problems at St. Petersburg Institute for Informatics and Automation of Russian Academy of Science, St. Petersburg, Russia. E-mail: ivkote@comsec.spb.ru

8 Andrey Chechulin, Ph.D, Leading Researchir fellow at St. Petersburg Institute for Informatics and Automation of Russian Academy of Science, St. Petersburg, Russia. E-mail: chechulin@comsec.spb.ru

1. Kotenko I., Levshun D., Chechulin A., Ushakov I., Krasov A. An Integrated Approach to Provide Security of Cyber-Physical Systems Based on Microcontrollers // Cybersecurity issues. 2018. No. 3 (27). P. 29-38. DOI: 10.21681/2311-3456-2018-3-29-38
2. 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. P. 79-90.
3. Choi H., Lee H., Kim H. Fast detection and visualization of network attacks on parallel coordinates // Computers & security. 2009. Vol. 28. No. 5. P. 276-288.
4. Kotenko I, Stepashkin M., Doynikova E. Security Analysis of Computer-aided Systems taking into account Social Engineering Attacks. Problems of information security. Computer systems. 2011, № 3. P. 40-57.
5. Doynikova E., Kotenko D., Kotenko I. Analysis of the intrusions using attack and service dependency graphs. 21th All-Russian Conference “Methods and technical tools of information security” (MTTIS 2012). Proceedings. St.Petersburg, Russia. June 24-29, 2012. P.45-47.
6. Ingols K., Lippmann R., Piwowarski K. Practical attack graph generation for network defense // 2006 22nd Annual Computer Security Applications Conference (ACSAC’06). IEEE, 2006. P.121-130.
7. Kolomeec M., Chechulin A., Kotenko I. V. Methodological Primitives for Phased Construction of Data Visualization Models // J. Internet Serv. Inf. Secur. 2015. Vol. 5. No. 4. P. 60-84.
8. Heitzmann A., Palazzi B., Papamanthou C., Tamassia R. Effective visualization of file system access-control // International Workshop on Visualization for Computer Security. Springer, Berlin, Heidelberg, 2008. P. 18-25.
9. Bishop M. Conspiracy and information flow in the take-grant protection model // Journal of Computer Security. 1996. Vol. 4. No. 4. P. 331-359.
10. Kim D., Ray I., France R., Li N. Modeling role-based access control using parameterized UML models // International Conference on Fundamental Approaches to Software Engineering. Springer, Berlin, Heidelberg, 2004. P. 180-193.
11. Kolomeets M., Chechulin A., Kotenko I., Saenko I. Access Control Visualization Using Triangular Matrices // 2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP). IEEE, 2019. C. 348-355.
12. Kolomeets M., Chechulin A., Kotenko I., Strecker M. Voronoi Maps for Planar Sensor Networks Visualization // International Symposium on Mobile Internet Security. Springer, Singapore, 2017. P. 96-109.
13. Roberts J. C. Guest editor’s introduction: special issue on coordinated and multiple views in exploratory visualization // Information Visualization. 2003. Vol. 2. No. 4. P. 199-200.
14. Kolomeec M, Chechulin A., Doynikova E., Kotenko I. Technique of Security Metrics Visualization // Journal of Instrument Engineering. 2018. Vol. 61 No. 10. P.873-880.
15. Sarkar M., Brown M. H. Graphical fisheye views // Communications of the ACM. 1994. Vol. 37. No. 12. P. 73-83.
16. Kolenda N. Psychology & Business [online]: A List of UX/UI Best Practices for Websites / URL: https://www.nickkolenda.com/user-experience/# (Access date 05.06.2019)
17. Ananto B. S., Sari R. F., Harwahyu R. Color transformation for color blind compensation on augmented reality system // 2011 International Conference on User Science and Engineering (i-USEr). IEEE, 2011. P. 129-134.
18. Hutchins E. L., Hollan J. D., Norman D. A. Direct manipulation interfaces // Human-computer interaction. 1985. Vol. 1. No. 4. P. 311-338.
19. Apple [online]: Use Multi-Touch gestures on your Mac – Apple Support / URL: https://support.apple.com/en-us/HT204895 (Access date 05.06.2019).
20. Apple Developer [online]: UI Design Do’s and Don’ts – Apple Developer / URL: https://developer.apple.com/design/tips/ (Access date 02.06.2019).
21. Apple Developer [online]: Gestures – User Interaction – iOS – Human Interface Guidelines – Apple Developer / URL: https://developer.apple.com/design/human-interface-guidelines/ios/user-interaction/gestures/ (Access date 05.06.2019).
22. World Usability Congress [online]: Touch Screen Usability Best Practices When Designing Automation User Interfaces (UI) – World Usability Congress / URL: https://worldusabilitycongress.com/touch-screen-usability-best-practices-when-designing-automation-user-interfaces-ui/ (Access date 05.06.2019).
23. Desnitsky V., Levshun D., Chechulin A., Kotenko I. Design Technique for Secure Embedded Devices: Application for Creation of Integrated Cyber-Physical Security System // JoWUA. 2016. Vol. 7. No. 2. P. 60-80.
24. N. Miloslavskaya, A. Tolstoy, A. Birjukov Information visualization in information security management for enterprise’s information infrastructure // Scientific visualization. 2014. V. 6. No. 2. P. 74-91.
25. Boronin P., Kucheryavy A. Internet of things as a new concept for the development of communication networks // Information Technologies and Telecommunications. 2014. No. 3 P. 7.