

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Политехнический институт
Факультет приборостроения,
информационных технологий и
электроники

Специальность 10.05.03 —

Специализация —

Кафедра

«Информационная безопасность систем
и технологий»

Информационная безопасность
автоматизированных систем
Защищенные автоматизированные
системы управления

ДИПЛОМНЫЙ ПРОЕКТ

на тему

ПОДСИСТЕМА ВИЗУАЛЬНОГО АНАЛИЗА ДАННЫХ ОБ ИНЦИДЕНТАХ
ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ В АВТОМАТИЗИРОВАННОЙ
СИСТЕМЕ ОБРАБОТКИ ИНЦИДЕНТОВ ИНФОРМАЦИОННОЙ
БЕЗОПАСНОСТИ

Студент

Теплов Никита Константинович

Руководитель

Шумкин С.Н.

Нормоконтролёр

Фатеев А.Г.

Рецензент начальник отдела ООО «НПФ «Кристалл»

Лакин К.А.

Работа допущена к защите (протокол заседания кафедры от 22.06.18 № 16)

Заведующий кафедрой

Зефилов С.И.

Работа защищена с отметкой отлично (протокол заседания ГЭК от 27.06.18 № 11)

Секретарь ГЭК

Иванов А.П.

Пенза 2018

Проверено — Ю.Б. Яковлев
22.06.18.

Оглавление

Введение.....	5
1 Обзор и сравнительный анализ подходов, средств и систем по обработке инцидентов информационной безопасности и визуального представления информации об инцидентах информационной безопасности	8
1.1 Анализ подходов по обработке инцидентов информационной безопасности	8
1.2 Обзор и сравнительный анализ средств и систем по обработке инцидентов ИБ и визуального представления информации об инцидентах ИБ	19
1.3 Результаты обзора и сравнительного анализа.....	37
2 Описание Единой системы мониторинга информационной безопасности и места подсистемы визуального анализа в Единой системе мониторинга информационной безопасности.....	38
2.1 Общее описание Единой системы мониторинга информационной безопасности	38
2.2 Описание места подсистемы визуального анализа в Единой системе мониторинга информационной безопасности	42
2.3 Необходимость использования подсистемы визуального анализа	47
3 Разработка архитектуры подсистемы визуального анализа с использованием UML-диаграмм и методологии функционального моделирования IDEF	49
4 Разработка программного обеспечения подсистемы визуального анализа	60
4.1 Общие принципы разработки.....	60
4.2 Описание этапов разработки программного обеспечения подсистемы визуального анализа	62
4.3 Описание разработанного программного обеспечения подсистемы визуального анализа	65

5	Тестирование разработанного программного обеспечения подсистемы визуального анализа	67
5.1	Разработка схемы тестирования разработанного программного обеспечения подсистемы визуального анализа.....	67
5.2	Функциональное тестирование программного обеспечения подсистемы	68
5.3	Тестирование производительности программного обеспечения подсистемы	70
6	Вредные психофизиологические факторы, влияющие на опорно-двигательный аппарат специалиста по защите информации, методы и средства физической культуры, снижающие их воздействие	72
	Заключение	75
	Список используемых источников	77
	Приложение А Коды программного обеспечения подсистемы визуального анализа данных.....	79
	Приложение Б Алгоритм работы подсистемы визуального анализа данных об инцидентах информационной безопасности.....	98

Обозначение и сокращения

АРМ — автоматизированное рабочее место;

АС — автоматизированная система;

БД — база данных;

ЕСМИБ ТУ — единая система мониторинга информационной безопасности;

ЗИ — защита информации;

ИБ — информационная безопасность;

КСИБ — коррелированные события информационной безопасности;

ПО — программное обеспечение;

ТЗ — техническое задание.

Введение

За последнее время количество угроз ИБ возросло в десятки раз, а преступники уже не являются хакерами одиночками, как 15 лет назад, а представляют собой мощные группировки хорошо организованного и технически оснащенного криминала с огромными многомиллионными оборотами денежных средств. Очень тревожной выглядит статистика инцидентов ИБ, которая собирается в крупных организациях. И это несмотря на то, что в организациях работают службы ИБ, которые вооружены программно-техническими средствами в области защиты информации (средствами и системами защиты информации, антивирусным ПО и т.д.). Одним из решений является разработка систем мониторинга информационной безопасности.

В территориальных управлениях Банка России действует единая система мониторинга информационной безопасности (ЕСМИБ ТУ). Она обеспечивает следующие виды деятельности по регистрации, обработке, анализу и хранению данных о событиях ИБ и коррелированных событиях ИБ (КСИБ):

- процесс мониторинга ИБ объекта автоматизации;
- контроль действий пользователей АС;
- взаимоувязанный анализ событий ИБ от разнотипных объектов из состава АС и системЗИ;
- формирование сообщений о событиях ИБ и сообщений о КСИБ;
- регистрация и анализ событий ИБ и КСИБ;
- хранение событий ИБ и КСИБ;
- обработка и доведение до персонала объекта автоматизации информации о КСИБ, своевременный учет и выработка рекомендаций для принятия мер по уменьшению негативного влияния КСИБ;
- формирование отчетов и их графическое представление.

Целью дипломного проекта является разработка подсистемы визуального анализа инцидентов ИБ, которая в последующем будет внедрена в ЕСМИБ ТУ и в последующем будет использоваться для анализа инцидентов ИБ.

Требования для разрабатываемой подсистемы следующие:

- анализ информации об инцидентах ИБ должен осуществляться по технологии OLAP на базе компонентов DevExpress;
- в качестве информации, анализируемой с использованием разрабатываемых OLAP-компонентов, выступает база данных о зарегистрированных инцидентах ИБ, включающая в себя:
 - время регистрации;
 - тип зарегистрированного инцидента;
 - задействованные информационные активы;
 - вовлеченные лица;
 - статус (открыт, закрыт, эскалирован);
- результаты анализа данных в разрабатываемой подсистеме должны быть представлены в табличном и графическом виде.

Таким образом, разработанная подсистема визуального анализа инцидентов ИБ должна выполнять следующие функции:

- запрос информации об инцидентах ИБ из БД об инцидентах;
- агрегация запрошенной информации за определенный период времени;
- задание необходимых для анализа осей измерения;
- табличное представление анализируемой информации;
- графическое представление.

В ходе дипломного проектирования будут рассмотрены следующие вопросы:

- обзор и сравнительный анализ подходов, средств и систем по обработке инцидентов ИБ и визуального представления информации об инцидентах ИБ;
- описание ЕСМИБ и места подсистемы визуального анализа в ЕСМИБ;
- разработка архитектуры подсистемы визуального анализа с использованием UML-диаграмм и методологии функционального моделирования IDEF;
- разработка программного обеспечения подсистемы визуального анализа;
- тестирование разработанного программного обеспечения;
- вредные психофизиологические факторы, влияющие на опорно-двигательный аппарат специалиста по защите информации, методы и средства физической культуры, снижающие их воздействие.

1 Обзор и сравнительный анализ подходов, средств и систем по обработке инцидентов информационной безопасности и визуального представления информации об инцидентах информационной безопасности

1.1 Анализ подходов по обработке инцидентов информационной безопасности

Наиболее широкое распространение получили следующие подходы по обработке данных, в том числе инцидентов информационной безопасности:

- OLAP;
- Data Mining.

OLAP (англ. online analytical processing, интерактивная аналитическая обработка) — технология обработки данных, заключающаяся в подготовке суммарной (агрегированной) информации на основе больших массивов данных, структурированных по многомерному принципу.

На данный момент OLAP является наиболее распространенным подходом по обработке данных, чем Data Mining.

Одним из преимуществ использования OLAP для обработки запросов — это скорость обработки данных. Реляционные БД хранят сущности в отдельных таблицах, которые обычно хорошо нормализованы. Эта структура удобна для операционных БД (системы OLTP), но сложные многотабличные запросы в ней выполняются относительно медленно.

OLAP-структура, созданная из рабочих данных, называется OLAP-куб. Куб создаётся из соединения таблиц с применением схемы звезды или схемы снежинки. В центре схемы звезды находится таблица фактов, которая содержит ключевые факты, по которым делаются запросы. Множественные таблицы с измерениями присоединены к таблице фактов. Эти таблицы показывают, как могут анализироваться агрегированные реляционные данные. Количество возможных агрегирований определяется количеством способов, которыми первоначальные данные могут быть иерархически отображены.

Например, все клиенты могут быть сгруппированы по городам или регионам страны (Запад, Восток, Север и т. д.), таким образом, 50 городов, восемь регионов и две страны составят три уровня иерархии с 60-ю членами. Также клиенты могут быть объединены по отношению к продукции; если существуют 250 продуктов по 20-ти категориям, три группы продукции и три производственных подразделения, то количество агрегатов составит 16 560. При добавлении измерений в схему количество возможных вариантов быстро достигает десятков миллионов и более. [1]

OLAP-куб — многомерный массив данных, как правило, разреженный и долговременно хранимый, используемый в OLAP. Может быть реализован на основе универсальных реляционных СУБД или специализированным программным обеспечением. [2]

OLAP-куб содержит базовые данные и информацию об измерениях (агрегаты). Куб потенциально содержит всю информацию, которая может потребоваться для ответов на любые запросы. Например, базу данных клиентов, время продажи, количество проданного товара. Пример куба приведен на рисунке 1. При огромном количестве агрегатов зачастую полный расчёт происходит только для некоторых измерений, для остальных же производится «по требованию».

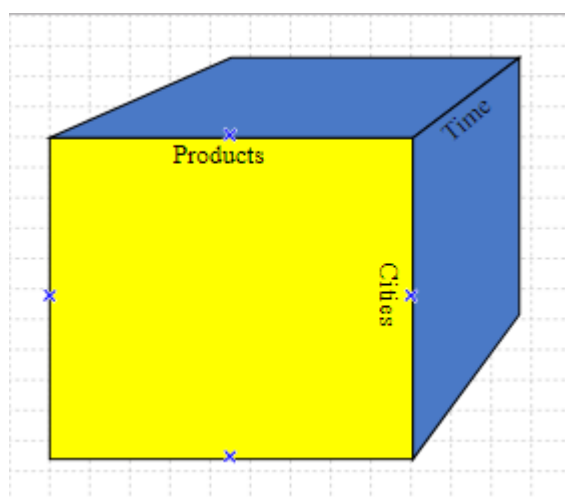


Рисунок 1 – Пример OLAP-куба

Существуют три типа OLAP:

- многомерная OLAP (Multidimensional OLAP — MOLAP);
- реляционная OLAP (Relational OLAP — ROLAP);
- гибридная OLAP (Hybrid OLAP — HOLAP).

MOLAP — классическая форма OLAP, так что её часто называют просто OLAP. Она использует суммирующую БД, специальный вариант процессора пространственных БД и создаёт требуемую пространственную схему данных с сохранением как базовых данных, так и агрегатов.

ROLAP работает напрямую с реляционным хранилищем, факты и таблицы с измерениями хранятся в реляционных таблицах, и для хранения агрегатов создаются дополнительные реляционные таблицы.

HOLAP использует реляционные таблицы для хранения базовых данных и многомерные таблицы для агрегатов.

Особым случаем ROLAP является «ROLAP реального времени» (Real-time ROLAP — R-ROLAP). В отличие от ROLAP в R-ROLAP для хранения агрегатов не создаются дополнительные реляционные таблицы, а агрегаты рассчитываются в момент запроса. При этом многомерный запрос к OLAP-системе автоматически преобразуется в SQL-запрос к реляционным данным.

Каждый тип хранения имеет определённые преимущества, хотя есть разногласия в их оценке у разных производителей. MOLAP лучше всего подходит для небольших наборов данных, он быстро рассчитывает агрегаты и возвращает ответы, но при этом генерируются огромные объёмы данных. ROLAP оценивается как более масштабируемое решение, использующее к тому же наименьшее возможное пространство. При этом скорость обработки значительно снижается. HOLAP находится посреди этих двух подходов, он достаточно хорошо масштабируется и быстро обрабатывается. Архитектура R-ROLAP позволяет производить многомерный анализ OLTP-данных в режиме реального времени.

Однако, сложность в применении OLAP состоит в создании запросов, выборе базовых данных и разработке схемы, в результате чего большинство

современных продуктов OLAP поставляются вместе с огромным количеством предварительно настроенных запросов. Другая проблема — в базовых данных. Они должны быть полными и непротиворечивыми. [1]

Data mining (рус. добыча данных, интеллектуальный анализ данных, глубинный анализ данных) — собирательное название, используемое для обозначения совокупности методов обнаружения в данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности. [3]

Задачи, решаемые методами Data Mining:

- классификация – это отнесение объектов (наблюдений, событий) к одному из заранее известных классов;
- регрессия, в том числе задачи прогнозирования — установление зависимости непрерывных выходных от входных переменных;
- кластеризация – это группировка объектов (наблюдений, событий) на основе данных (свойств), описывающих сущность этих объектов, объекты внутри кластера должны быть "похожими" друг на друга и отличаться от объектов, вошедших в другие кластеры. Чем больше похожи объекты внутри кластера и чем больше отличий между кластерами, тем точнее кластеризация;
- ассоциация – выявление закономерностей между связанными событиями;
- последовательные шаблоны – установление закономерностей между связанными во времени событиями, т.е. обнаружение зависимости, что если произойдет событие X, то спустя заданное время произойдет событие Y;
- анализ отклонений – выявление наиболее нехарактерных шаблонов.

[4]

Ниже будут рассмотрены такие задачи, как классификация и кластеризация, так как для анализа инцидентов ИБ являются наиболее подходящим.

Классификация является наиболее простой и одновременно наиболее часто решаемой задачей Data Mining.

Различают:

- вспомогательную (искусственную) классификацию, которая производится по внешнему признаку и служит для придания множеству предметов (процессов, явлений) нужного порядка;

- естественную классификацию, которая производится по существенным признакам, характеризующим внутреннюю общность предметов и явлений. Она является результатом и важным средством научного исследования, т.к. предполагает и закрепляет результаты изучения закономерностей классифицируемых объектов.

В зависимости от выбранных признаков, их сочетания и процедуры деления понятий классификация может быть:

- простой – деление родового понятия только по признаку и только один раз до раскрытия всех видов. Примером такой классификации является дихотомия, при которой членами деления бывают только два понятия, каждое из которых является противоречащим другому (т.е. соблюдается принцип: "А и не А");

- сложной – применяется для деления одного понятия по разным основаниям и синтеза таких простых делений в единое целое. Примером такой классификации является периодическая система химических элементов.

Классификация — это закономерность, позволяющая делать вывод относительно определения характеристик конкретной группы. Таким образом, для проведения классификации должны присутствовать признаки, характеризующие группу, к которой принадлежит то или иное событие или

объект (обычно при этом на основании анализа уже классифицированных событий формулируются некие правила).

Классификация относится к стратегии обучения с учителем (supervised learning), которое также именуют контролируемым или управляемым обучением.

Задачей классификации часто называют предсказание категориальной зависимой переменной (т.е. зависимой переменной, являющейся категорией) на основе выборки непрерывных и/или категориальных переменных.

Процесс классификации состоит из двух этапов: конструирования модели и ее использования.

Конструирование модели – это описание множества предопределенных классов. Этот этап включает в себя следующие действия:

- каждый пример набора данных относится к одному предопределенному классу;
- на этапе конструирования модели используется обучающее множество, на нем происходит конструирование модели;
- полученная модель представлена классификационными правилами, деревом решений или математической формулой.

Использование модели – это классификация новых или неизвестных значений. Этот этап состоит из следующих действий:

- оценка правильности (точности) модели;
- использование модели для классификации новых примеров, класс которых неизвестен.

При оценке правильности (точности) модели известные значения из тестового примера сравниваются с результатами использования полученной модели. Также при оценке точности модели используется такой показатель, как уровень точности модели. Уровень точности — процент правильно классифицированных примеров в тестовом множестве. Тестовое множество – это множество, на котором тестируется построенная модель. Оно не должно

зависеть от обучающего множества. Если точность модели допустима, то возможно использование модели для классификации новых примеров, класс которых неизвестен.

Для классификации используются различные методы. Основные из них:

- классификация с помощью деревьев решений;
- байесовская (наивная) классификация;
- классификация при помощи искусственных нейронных сетей;
- классификация методом опорных векторов;
- статистические методы, в частности, линейная регрессия;
- классификация при помощи метода ближайшего соседа;
- классификация CBR-методом;
- классификация при помощи генетических алгоритмов.

Пример применения метода классификации.

Необходимо определить, к какому классу принадлежит новый клиент и какой из двух видов рекламных материалов ему стоит отсылать.

Для наглядности база данных представлена в двухмерном измерении (возраст и доход), в виде множества объектов, принадлежащих классам 1 (оранжевая метка) и 2 (серая метка). На рисунке 2 приведены объекты из двух классов.

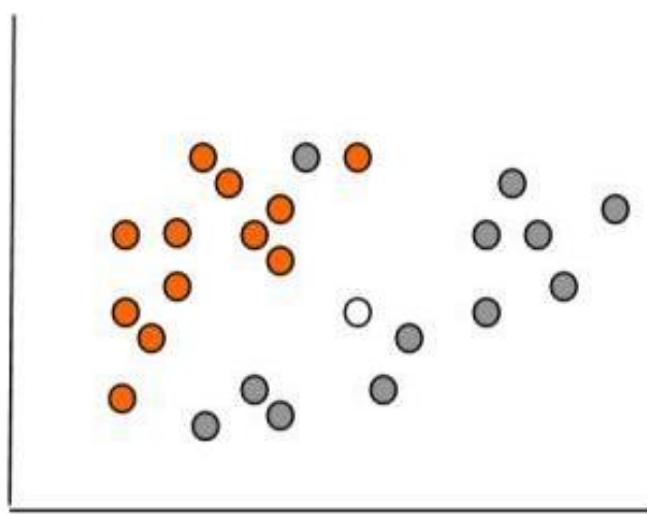


Рисунок 2 – Множество объектов базы данных в двухмерном измерении

Пример классификации с помощью деревьев решений:

```
if X > 5 then grey
else if Y > 3 then orange
else if X > 2 then grey
else orange
```

Пример классификации при помощи искусственных нейронных сетей приведен на рисунке 3.

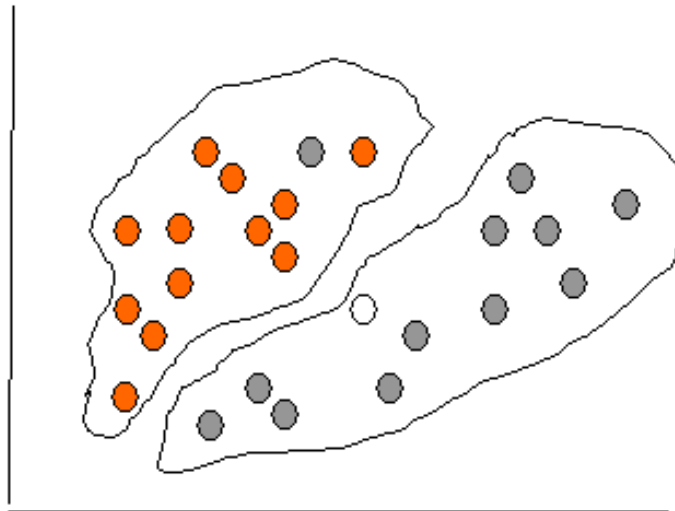


Рисунок 3 – Результат классификации при помощи искусственных нейронных сетей

Задача кластеризации сходна с задачей классификации, является ее логическим продолжением, но ее отличие в том, что классы изучаемого набора данных заранее не predetermined.

Синонимами термина «кластеризация» являются «автоматическая классификация», «обучение без учителя» и «таксономия».

Кластеризация предназначена для разбиения совокупности объектов на однородные группы (кластеры или классы). Если данные выборки представить, как точки в признаковом пространстве, то задача кластеризации сводится к определению "сгущений точек".

Цель кластеризации (кластерного анализа) — поиск существующих структур.

Кластеризация предназначена для разбиения совокупности объектов на однородные группы (кластеры или классы). Если данные выборки представить

как точки в признаковом пространстве, то задача кластеризации сводится к определению "сгущений точек".

Цель кластеризации — поиск существующих структур.

Кластеризация является описательной процедурой, она не делает никаких статистических выводов, но дает возможность провести разведочный анализ и изучить "структуру данных".

Само понятие "кластер" определено неоднозначно: в каждом исследовании свои "кластеры". Переводится понятие кластер (cluster) как "скопление", "гроздь".

Кластер можно охарактеризовать как группу объектов, имеющих общие свойства.

Характеристиками кластера можно назвать два признака:

- внутренняя однородность;
- внешняя изолированность.

Вопрос, задаваемый аналитиками при решении многих задач, состоит в том, как организовать данные в наглядные структуры, т.е. развернуть таксономию.

Следует отметить, что в результате применения различных методов кластерного анализа могут быть получены кластеры различной формы. Например, возможны кластеры "цепочного" типа, когда кластеры представлены длинными "цепочками", кластеры удлиненной формы и т.д., а некоторые методы могут создавать кластеры произвольной формы.

Различные методы могут стремиться создавать кластеры определенных размеров (например, малых или крупных) либо предполагать в наборе данных наличие кластеров различного размера.

Некоторые методы кластерного анализа особенно чувствительны к шумам или выбросам, другие — менее.

В результате применения различных методов кластеризации могут быть получены неодинаковые результаты, это нормально и является особенностью работы того или иного алгоритма. [5]

Типами входных данных для задачи кластеризации могут быть:

- признаковое описание объектов, то есть каждый объект описывается набором своих характеристик, называемых признаками, которые могут быть числовыми или нечисловыми;
- матрица расстояний между объектами, то есть каждый объект описывается расстояниями до всех остальных объектов метрического пространства;
- матрица сходства между объектами, то есть учитывается степень сходства объекта с другими объектами выборки в метрическом пространстве; сходство здесь дополняет расстояние (различие) между объектами до 1.

Общепринятой классификации методов кластеризации не существует, но можно выделить ряд групп подходов (некоторые методы можно отнести сразу к нескольким группам и потому предлагается рассматривать данную типизацию как некоторое приближение к реальной классификации методов кластеризации):

а) вероятностный подход, к таким относятся:

- 1) К-средних;
- 2) К-медиан;
- 3) ЕМ-алгоритм;
- 4) алгоритмы семейства FOREL;
- 5) дискриминантный анализ;

б) подходы на основе систем искусственного интеллекта — весьма условная группа, так как методов очень много и методически они весьма различны, и к ним относятся:

- 1) метод нечеткой кластеризации С-средних (C-means);
- 2) нейронная сеть Кохонена;

- 3) генетический алгоритм;
- в) логический подход (построение дендрограммы осуществляется с помощью дерева решений);
- г) теоретико-графовый подход, к такому подходу относятся графовые алгоритмы кластеризации;
- д) иерархический подход;
- е) другие методы, не вошедшие в предыдущие группы:
 - 1) статистические алгоритмы кластеризации;
 - 2) ансамбль кластеризаторов;
 - 3) алгоритмы семейства KRAV;
 - 4) алгоритм, основанный на методе просеивания;
 - 5) DBSCAN и другие.

Предполагается, что при вероятностных подходах каждый рассматриваемый объект относится к одному из k классов. Некоторые авторы (например, А. И. Орлов) считают, что данная группа вовсе не относится к кластеризации и противопоставляют её под названием «дискриминация», то есть выбор отнесения объектов к одной из известных групп (обучающих выборок). При иерархическом подходе предполагается наличие вложенных групп (кластеров различного порядка). Алгоритмы в свою очередь подразделяются на агломеративные (объединительные) и дивизивные (разделяющие). По количеству признаков иногда выделяют монотетические и политетические методы классификации. Примером иерархического подхода является иерархическая дивизивная кластеризация или таксономия. В этом подходе задачи кластеризации рассматриваются в количественной таксономии. Подходы г) и д) иногда объединяют под названием структурного или геометрического подхода, обладающего большей формализованностью понятия близости. Несмотря на значительные различия между перечисленными методами все они опираются на исходную «гипотезу компактности»: в пространстве объектов все близкие объекты должны относиться к одному

кластеру, а все различные объекты соответственно должны находиться в различных кластерах. [6]

1.2 Обзор и сравнительный анализ средств и систем по обработке инцидентов

ИБ и визуального представления информации об инцидентах ИБ

На российском рынке представлены следующие средства и системы обработки инцидентов ИБ:

- InfoWatch Vision;
- HP ArcSight;
- Splunk Enterprise;
- IBM I2;
- IBM QRadar SIEM;
- Neurodat SIEM.

Все системы обработки относятся к технологии SIEM. SIEM (Security information and event management) — объединение двух терминов, обозначающих область применения ПО: SIM (Security information management) — управление информационной безопасностью, и SEM (Security event management) — управление событиями безопасности. Технология SIEM обеспечивает анализ в реальном времени событий (тревог) безопасности, исходящих от сетевых устройств и приложений. SIEM представлено приложениями, приборами или услугами, и используется также для журналирования данных и генерации отчетов в целях совместимости с прочими бизнес-данными. [7]

InfoWatch Vision — это инструмент визуальной аналитики информационных потоков в режиме реального времени.

Разработчик: компания InfoWatch.

Данное средство разработано для совместного использования с системой InfoWatch Traffic Monitor.

InfoWatch Vision состоит из четырех интерактивных листов: граф связей, сводка, досье, детали, которые отражают основные функциональные возможности продукта.

Граф связей позволяет визуально проанализировать связи сотрудников, внешних контактов и перемещение информации на USB-накопители, принтеры, веб-ресурсы. Граф используется офицером безопасности для визуального анализа связи сотрудников и внешних контактов. Граф можно строить сразу для нескольких сотрудников или целого отдела. Граф мгновенно перестраивается в зависимости от выбранных параметров: сотрудники, временные периоды, политики, пересылаемые файлы, объекты защиты и пр. Узлы и связи на графе кликабельны – можно провалиться в события любой связи.

Досье позволяет заглянуть внутрь информационных потоков, контролировать движение чувствительной информации и выявлять аномалии. Досье используется офицером безопасности для контроля конкретных сотрудников и полного анализа их активности в компании. Досье на внешний контакт позволяет оценить, с кем из компании он связан, какие политики назначены, какие файлы он получил и отправил. Досье на внешний контакт формируется автоматически без участия офицера безопасности.

Инструмент «Сводка» позволяет заглянуть внутрь информационных потоков, контролировать движение чувствительной информации и выявлять аномалии. Сводка используется офицером безопасности для ежедневного мониторинга информационных потоков компании, а также для проведения расследования при отсутствии прямых входящих данных (инцидент, дата, сотрудник и т.д.).

Конструктор позволяет формировать отчеты по инцидентам информационной безопасности, представлять специфические разрезы статистики и графики в одном окне, добавлять комментарии, изображения и выгружать результаты в формате pdf и ppt.

Инструмент «Детали» позволяет провалиться в конкретное событие, с возможностью просматривать текст письма или сообщения, выгрузить вложение и список событий по выборке и провести расследование инцидента.

Архитектура системы представлена на рисунке 4. Интерфейс системы представлен на рисунке 5. [8]



Рисунок 4 – Архитектура системы InfoWatch Vision

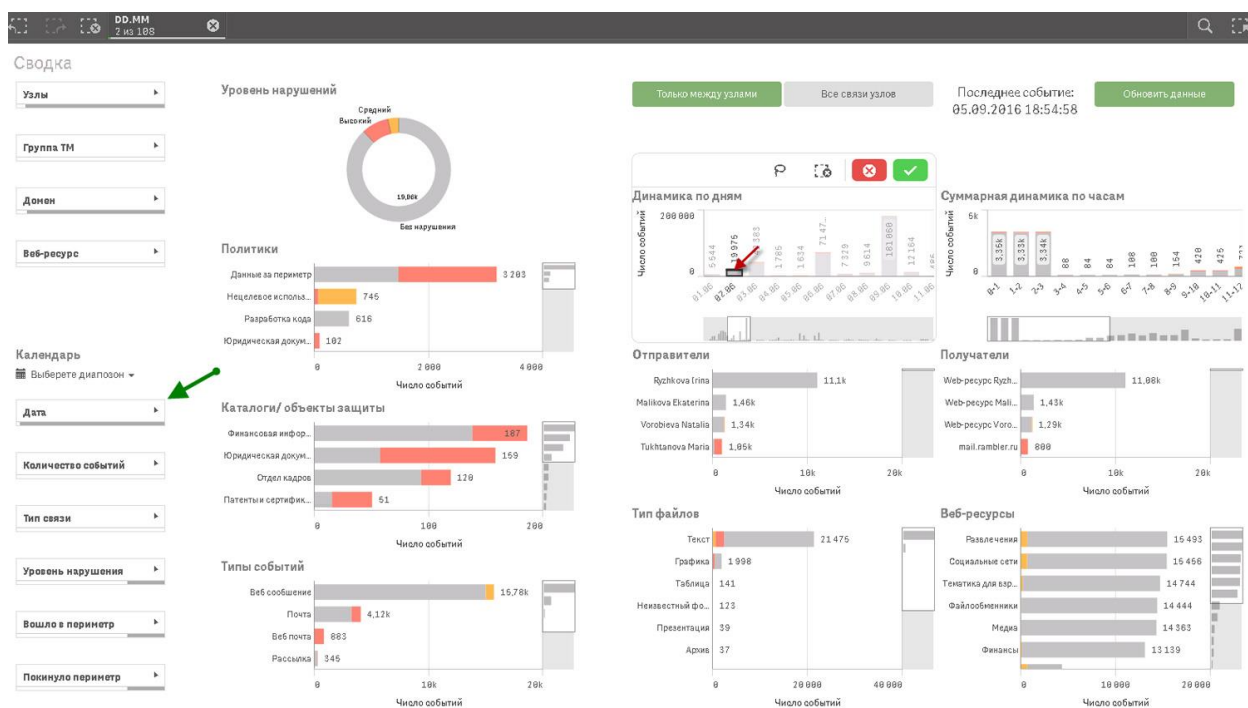


Рисунок 5 – Интерфейс системы InfoWatch Vision

Helwett Packard представил на российском рынке свою систему обработки — ArcSight.

Основой продуктовой линейки HP ArcSight является комплексное решение HP ArcSight Security Intelligence, ядром которого служит продукт HP ArcSight ESM (Enterprise Security Manager). Данный продукт обеспечивает сбор, обработку и хранение событий безопасности, которые могут поступать от различных источников. HP ArcSight ESM поддерживает интеграцию с большим количеством прикладных систем и устройств (более трех сотен) и поставляется с несколькими сотнями предустановленных правил корреляции. В состав поставки также может входить уникальный агент FlexConnector, позволяющий осуществлять интеграцию с любым типом приложения.

HP ArcSight ESM является лидером на рынке по техническим возможностям и возможностям интеграции с бизнес-приложениями. Архитектура HP ArcSight ESM позволяет развернуть решение даже в территориально-распределенной информационной системе с низкоскоростными каналами связи. HP ArcSight ESM поставляется как в программном, так и программно-аппаратном виде, что выгодно отличает его от других систем корреляции.

Для упрощения задачи по сбору, хранению и анализу журналов аудита может использоваться продукт HP ArcSight Logger – готовый программный или программно-аппаратный комплекс, который может собирать и анализировать все данные журналов аудита организации, предоставляя сжатый и экономичный репозиторий для хранения логов.

Для более простого и эффективного сбора информации о событиях безопасности в составе HP ArcSight ESM и HP ArcSight Logger могут использоваться программные комплексы HP ArcSight Connectors. Необходимо отметить, что HP ArcSight Connectors также могут поставляться в виде программно-аппаратных комплексов (HP ArcSight Connector Appliance).

Решения HP ArcSight Security Intelligence включают в себя следующие продукты:

- HP ArcSight Logger — обеспечивает сбор и фильтрацию событий;
- HP ArcSight Threat Response — обеспечивает моментальную реакцию на инциденты путем анализа информации от HP ArcSight ESM, локализацию проблемы и применение ответных мер реагирования;
- HP ArcSight Configuration Management — позволяет провести конфигурацию сетевого оборудования и настроек безопасности;
- HP ArcSight Fraud Detection — уникальное решение для выявления и предотвращения мошенничества в области интернет-банкинга и банковских (пластиковых) карт.

Внедрение системы мониторинга на основе продуктов HP ArcSight дает возможность автоматизировать процесс принятия решений по реагированию на события, связанные с нарушением политик информационной безопасности Компании. При этом применение систем мониторинга также позволяет значительно повысить эффективность уже установленных в организации средств защиты.

На сегодняшний день решения HP ArcSight активно используют во всем мире крупнейшие операторы связи, финансовые организации и государственные структуры. Интерфейс системы HP ArcSight приведен на рисунке 6. [9]

Splunk Enterprise — платформа для операционного анализа. Она упрощает сбор, анализ и работу с большими наборами данных, которые генерируются технологической инфраструктурой, системами безопасности и бизнес-приложениями.

Осуществляя мониторинг и анализ всех действий, от посещений веб-сайтов и выполнения транзакций до сетевых операций и событий в системе безопасности, Splunk Enterprise помогает получить ценную аналитическую

информацию из машинных данных. Многочисленные встроенные функции поиска и визуализации и готовые материалы для разных сценариев использования помогут пользователю быстро получить полезные сведения. Архитектура системы Splunk Enterprise приведен на рисунке 7. Интерфейс представлен на рисунке 8.

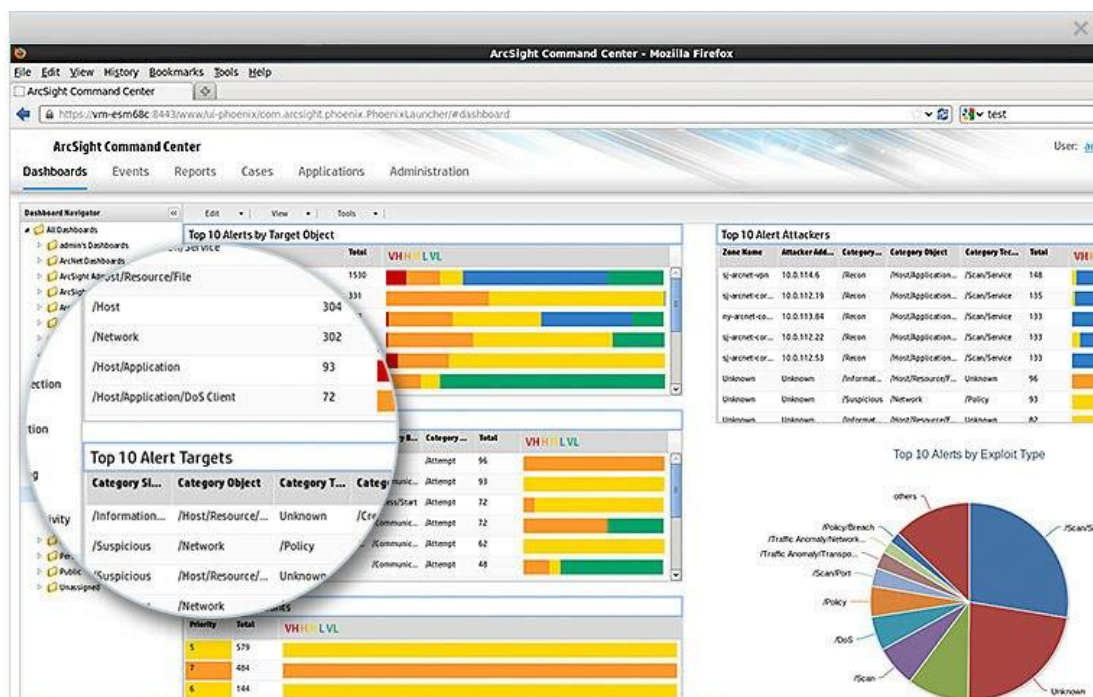


Рисунок 6 – Интерфейс системы HP ArcSight

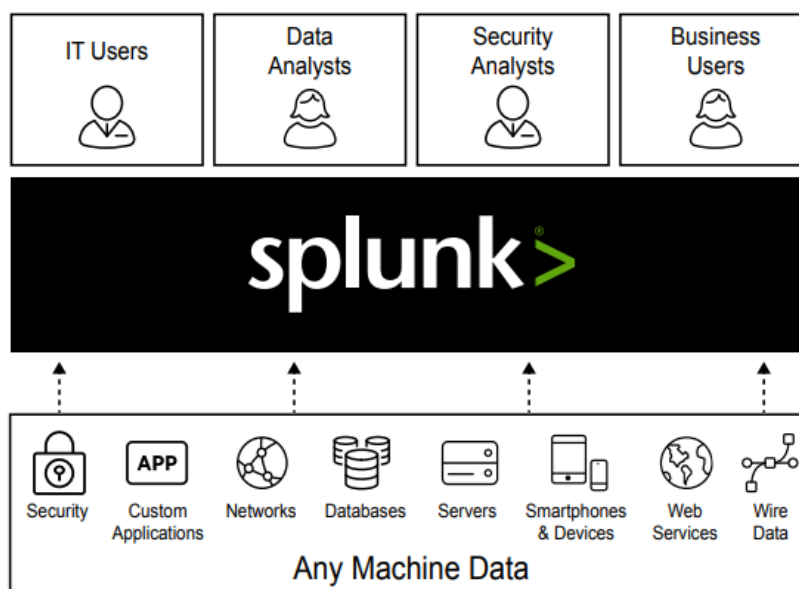


Рисунок 7 – Архитектура системы Splunk Enterprise

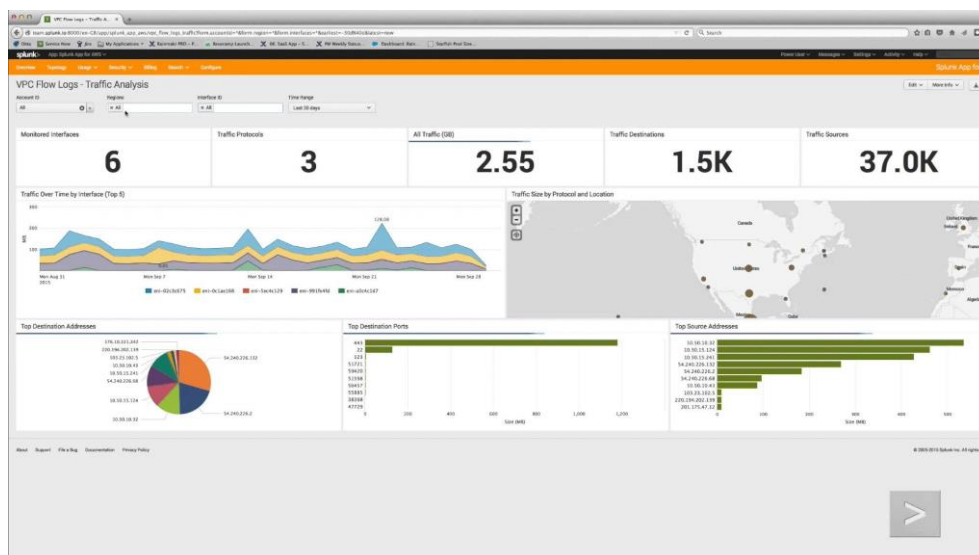


Рисунок 8 – Интерфейс системы Splunk Enterprise

Возможности:

- собирает и индексирует журналы и машинные данные из любого источника;
- мощные возможности поиска, анализа и визуализации для пользователей всех типов;
- существуют приложения для таких областей, как безопасность, ИТ, бизнес-аналитика и т. д.;
- прозрачность в локальных, облачных и гибридных средах;
- масштабируемость, защита и доступность для любой организации;
- доступно как отдельное приложение или как решение SaaS. [10]

Система IBM i2 позволяет производить анализ связей и закономерностей в большом объеме данных. Решение данной задачи реализуется с помощью визуализации связей и статистических закономерностей между записями.

Система IBM i2 состоит из различных компонентов, которые, в зависимости от решаемых задач, могут быть применены в различных вариантах архитектуры.

iBase позволяет собирать и анализировать данные из различных источников. iBase является центральным узлом системы, с помощью которого

осуществляется хранение и анализ информации. Также с помощью этого компонента осуществляется управление учетными записями и правами доступа для многопользовательских систем. iBase позволяет:

- спроектировать базу данных, в которой будет впоследствии храниться собираемая в системе информация;
- формировать правила импорта данных из сторонних источников;
- осуществлять контроль доступа к данным и аудит событий такого доступа;
- централизованно хранить результаты анализа.

iBridge – модуль системы, позволяющий подключать Analyst's Notebook к уже существующим базам данных в режиме реального времени. iBridge позволяет:

- подключаться к внешним базам данных напрямую;
- осуществлять поиск элементов в базе данных;
- формировать визуальные запросы на поиск информации.

Analyst's Notebook – компонент i2, являющийся основным интерфейсом взаимодействия пользователя с системой. Позволяет анализировать информацию, содержащуюся в подключенных к нему файлах и базах данных, представляет результат анализа в виде схем взаимосвязей и временных диаграмм. На схему также можно добавлять произвольные объекты, не содержащиеся в базе данных, например, текстовые комментарии или фотографии. Имеющиеся в Analyst's Notebook аналитические инструменты позволяют выполнять поиск элементов на схеме, взаимосвязей между ними и другие задачи. Analyst's Notebook позволяет:

- выполнять поиск элементов в базе данных;
- графически представлять информацию, содержащуюся в базе данных;

— подключать данные из различных локальных источников (текстовые файлы, электронные таблицы).

Используя различные наборы продуктов семейства IBM i2 можно создать систему анализа и визуализации данных.

К примеру, если информация, подлежащая анализу и визуализации хранится в виде текстовых файлов или таблиц Excel, имея при этом небольшой объем (порядка 50 тысяч записей), то для построения визуальной схемы взаимосвязей или временной шкалы будет достаточно использовать i2 Analyst's Notebook. С помощью мастера импорта можно загрузить в i2 Analyst's Notebook данные хранящиеся практически в любом текстовом формате. При этом мастер импорта имеет интуитивно понятный графический интерфейс, что позволяет пользователю организовать чтение данных из различных источников не прибегая к помощи системного администратора. Данный вариант приведен на рисунке 9.

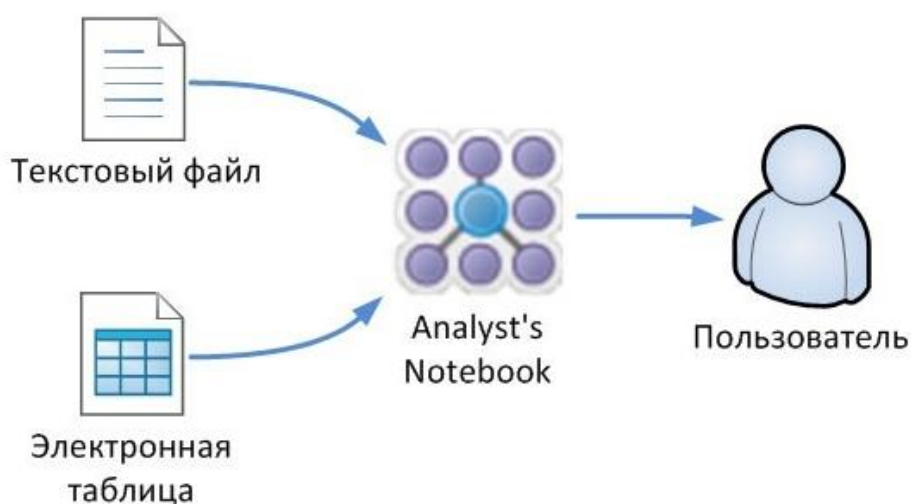


Рисунок 9 – Вариант архитектурного решения №1

В случае необходимости выполнения анализа информации, хранящейся в базе данных и имеющей большие объемы, для ее визуального отображения целесообразно использовать связку из двух продуктов. i2 Analyst's Notebook и i2 iBase. iBase позволяет выгрузить необходимую информацию из баз данных в

собственное хранилище, из которого i2 Analyst's Notebook будет брать данные для построения визуальной схемы. Данный вариант приведен на рисунке 10.

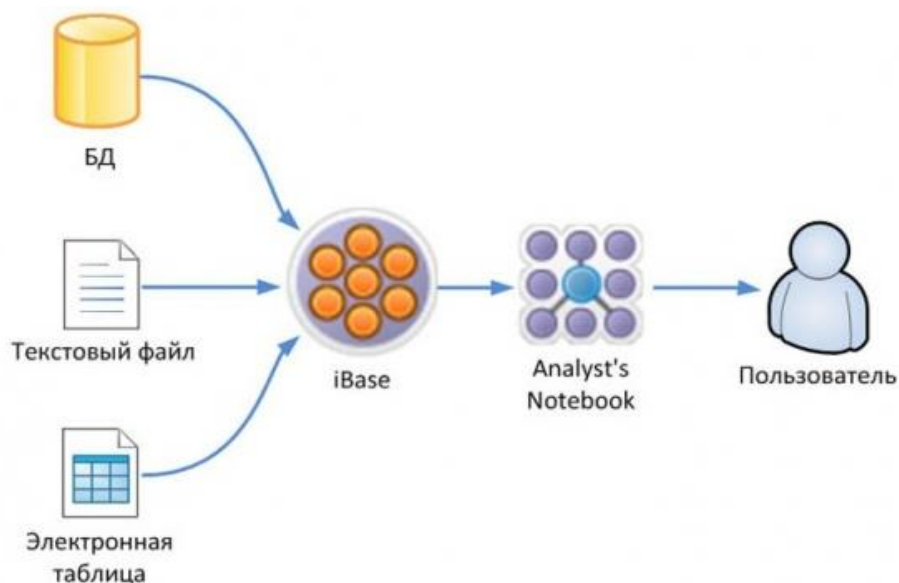


Рисунок 10 – Вариант архитектурного решения №2

Для того чтобы получить возможность визуализировать информацию базы данных, которая изменяется в режиме реального времени, и повысить оперативность построения схем и диаграмм, существует возможность использования связки продуктов i2 Analyst's Notebook и iBridge. Такая архитектура позволит интерпретировать запросы пользователя в Analyst's Notebook в запросы базы данных, минуя шаги импорта. Данный пример приведен на рисунке 11. [11]

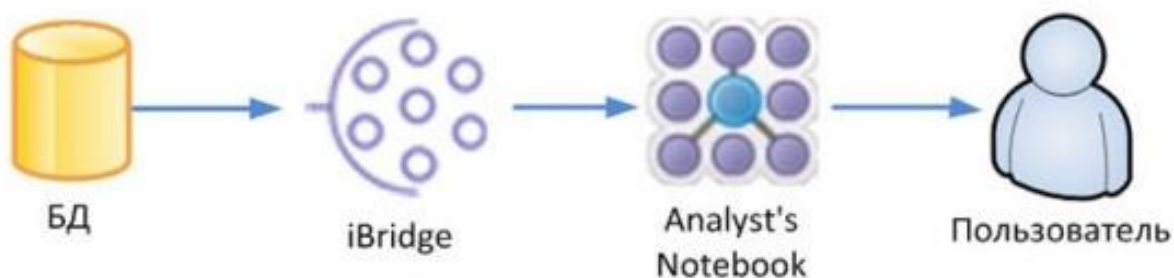


Рисунок 11 – Вариант архитектурного решения №3

Интерфейс системы IBM I2 приведен на рисунке 12.

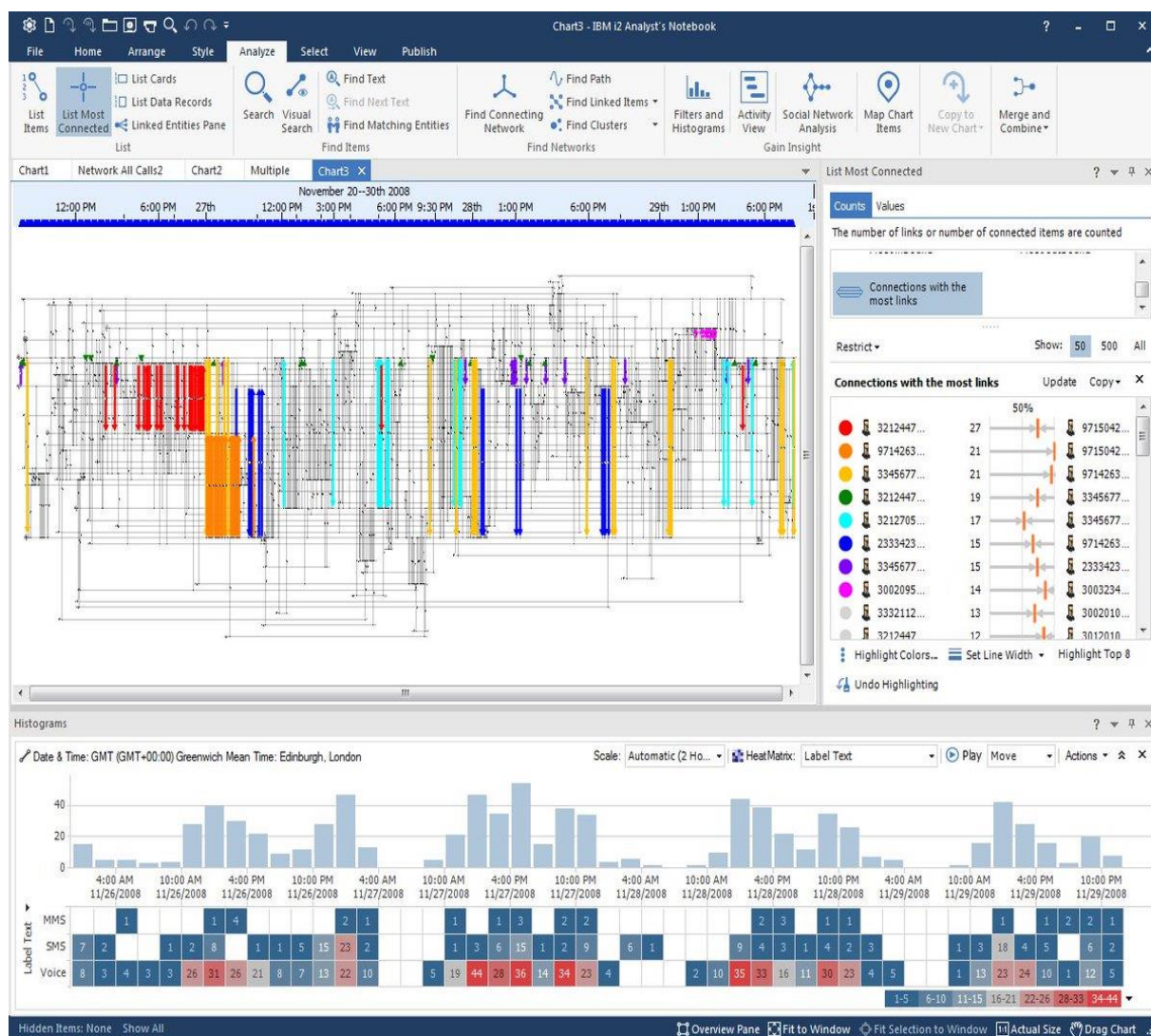


Рисунок 12 – Интерфейс системы IBM I2

QRadar SIEM – средство управления событиями и инцидентами ИБ, являющееся одним из трех модулей, входящих в решение QRadар, которое представляет собой современную интеллектуальную платформу для комплексного мониторинга ИБ предприятия.

QRadar включает в себя три совместно функционирующих модуля:

- модуль управления логами (Log Management);
- модуль управления событиями и инцидентами информационной безопасности (SIEM);
- модуль управления рисками (Risk Management).

Интерфейс системы приведен на рисунке 13.

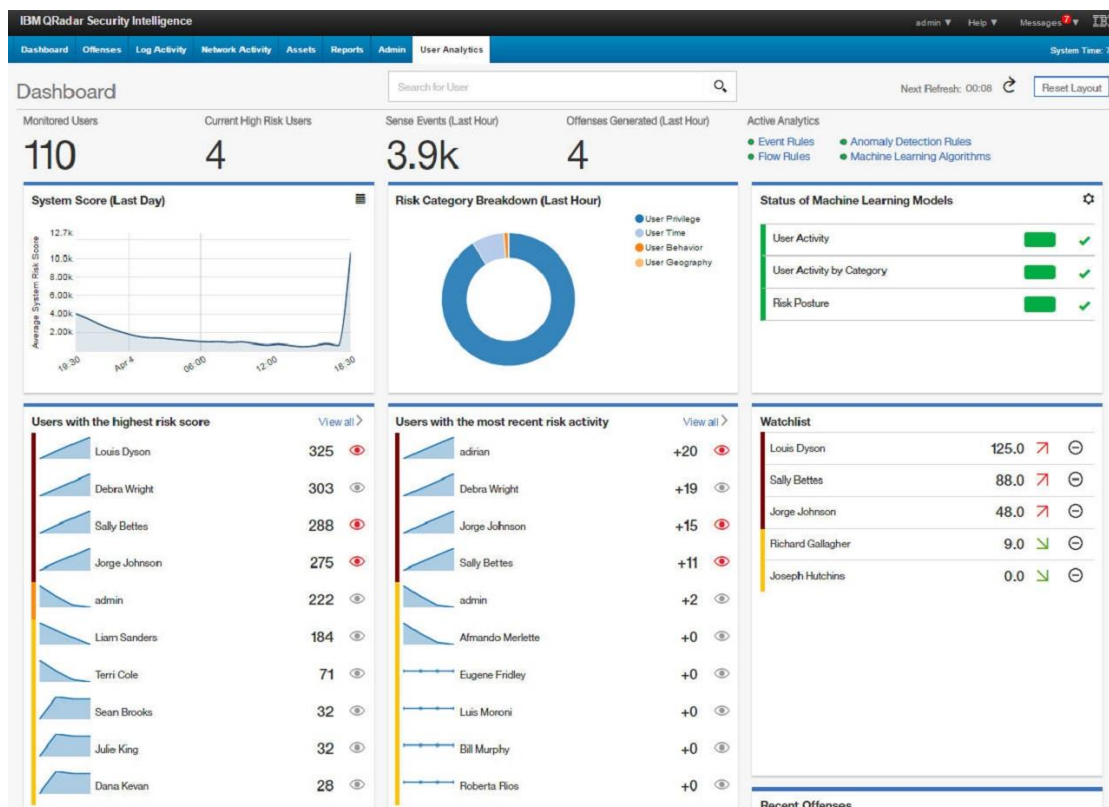


Рисунок 13 – Интерфейс системы QRadar SIEM

QRadar SIEM обеспечивает тотальную видимость внутри сети, осуществляя сбор и анализ данных, которые позволяют оперативно получать всю необходимую информацию о событиях безопасности и работе сетевых устройств, независимо от сложности сетевой конфигурации. Максимальная прозрачность сети позволяет с наибольшей эффективностью управлять ее безопасностью и обнаруживать все существующие и потенциальные угрозы задолго до их реализации. QRadar SIEM – это интегрированное средство, прекрасно справляющееся с задачами управления политиками по соответствию требованиям и стандартам, сбора и анализа логов и предоставляющее самый современный инструментарий для выявления угроз. Решение основано на гибкой платформе QRadar Security Intelligence Platform, которая может развиваться параллельно с предприятием и подстраиваться под его расширяющуюся инфраструктуру, легко и оперативно обеспечивая мониторинг корпоративной безопасности.

QRadar SIEM собирает информацию из следующих источников:

- события системы безопасности – события от брандмауэров, VPNs, IDS/IPS, и т.д.;
- сетевые события – события от свитчей, роутеров, серверов, хостов и т.д.;
- монитор активности сети – контекстные идентификаторы протоколов 7-го уровня от сетевого трафика и приложений;
- монитор активности пользователей – данные продуктов типа IAM (Identity and Access Management) и сканеров уязвимостей;
- журналы событий приложений – ERP, документооборот, базы данных приложений, административные платформы и т.д.

Объединяя разрозненную информацию, QRadar SIEM делает более эффективным обнаружение всех современных угроз. Данные нормируются и коррелируются для своевременного выявления, уведомления и реагирования на угрозы, которые не в состоянии определить другие средства защиты с ограниченной видимостью. Мониторинг, осуществляемый QRadar SIEM, позволит предприятиям обнаружить сложные угрозы, среди которых инсайдерское мошенничество, нецелевое использование приложений и многие другие.

Особенно эффективно применение QRadar SIEM для предприятий с крупномасштабными сетями, в которых регистрируются миллионы и более событий в день. QRadar SIEM осуществляет сбор, анализ и хранение данных и предоставляет корреляцию событий в режиме реального времени. Это позволяет среди огромного количества данных распознать те, которые могут привести к инцидентам безопасности. Миллиарды сетевых событий и потоков могут быть упрощены, что, соответственно, упростит процессы выявления угроз, аудита и создания отчетности, соответствующей требованиям и стандартам. В целях аудита и защиты сетевой инфраструктуры QRadar SIEM обеспечит долгосрочный сбор событий и прикладных данных, их архивирование, поиск необходимых данных и отчетность.

QRadar SIEM контролирует все серьезные инциденты и угрозы, предоставляя хронологию обслуживания и всю необходимую сопутствующую информацию. Благодаря этому решению службы безопасности смогут всегда узнать ответы на такие вопросы, как: кто нарушает безопасность, какой объект подвергается нападению, в каком месте проводить расследование, каковы последствия для бизнеса? QRadар SIEM предоставит полную информацию о факторах, нарушающих нормальный режим работы, пользователях, моделях нарушителей, важности ресурсов, характеристиках уязвимостей, уровне активности угроз и отчетах о предыдущих нарушениях и т.д. Таким образом, служба безопасности сможет получить все необходимые сведения для своевременного реагирования на любые инциденты безопасности.

При помощи QRadар SIEM можно обнаружить любые отклонения и изменения в работе приложений, серверов, хостов и сегментов сети. Возможность идентификации трафика на прикладном уровне позволяет QRadар SIEM достаточно точно анализировать и понимать политики и угрозы корпоративной сети предприятия, а также проводить общий мониторинг сетевой активности. Функция контроля работы с таким приложением, как Skype, и социальными сетями (включая Twitter, LinkedIn и т.д.), также позволяет улучшить видимость сети. Поддерживая обнаружение большого количества отклонений и поведенческих правил, QRadар SIEM может детально ответить на вопрос о том, какой пользователь и что использует. Контентный анализ и оповещения при передаче контента, корреляция с прочей сетевой активностью и журналами событий позволяют выявить нецелевую передачу данных. Возможности фильтрации и выбор любого временного промежутка для анализа позволяют пользователю настроить вариант вывода результатов по собственному усмотрению.

Теперь IT-специалистам доступна улучшенная видимость активности широкого спектра бизнес-приложений в виртуальных сетях. Виртуальные сервера, как и физические, имеют уязвимости в системе безопасности, поэтому

прозрачность виртуальной среды обработки данных требуется для точного определения необходимых мероприятий по защите приложений и данных.

QRadar SIEM определяет наиболее уязвимые участки сети и немедленно оповещает об этом, если данные системы становятся активны и несут за собой потенциальную опасность. К примеру, компании могут сканировать свою сеть на наличие необновленных приложений, устройств и систем, определять, что из этого имеет выход в Интернет и назначать приоритеты на исправление, основываясь на профилях рисков для каждого приложения.

QRadar SIEM помогает придерживаться нормативных требований стандартов и регуляторов, предоставляя отчетность по соответствию. Уникальная корреляция и интеграция всех каналов наблюдения предоставляет:

- полную отчетность по информационным рискам для аудиторов;
- тысячи шаблонов отчетов и правил для соответствия нормативным требованиям IT-отрасли.

Компании могут оперативно реагировать на соответствие требованиям отраслевых стандартов за счет возможности расширения шаблонов, добавляя в них новые характеристики и требования, используя передовой опыт посредством автоматических обновлений. Кроме того, профили всех сетевых ресурсов могут быть сгруппированы по бизнес-функциям. QRadar SIEM предлагает встроенные панели и шаблоны отчетов и правил для соответствия нормативным требованиям следующих стандартов: PCI-DS, CobiT, SOX, GLBA, NERC/FERC и т.д.

Централизованная интуитивно понятная консоль управления обеспечивает ролевой доступ, предоставляя глобальный обзор управления инцидентами и отчетности. Панели управления QRadar SIEM предлагаются как функционал, и пользователи могут сами создать и настроить свое рабочее пространство в соответствии с решаемыми задачами. Такая детализация предоставит возможность гораздо проще выявлять и выбирать всплески событий и сетевые потоки, связанные с нарушениями. QRadar SIEM предлагает

около 3 500 шаблонов отчетов, связанных с конкретными устройствами, ролями и требованиями регуляторов.

Решение QRadar SIEM в первую очередь предназначено для малого и среднего бизнеса, но может быть с успехом развернуто в компании любого масштаба за счет своей легкой масштабируемости. QRadar SIEM делает возможным автоматический переход с другого решения и полную синхронизацию между системами. Внедрять дополнительные решения сторонних производителей нет необходимости, так как QRadar SIEM обеспечивает высокий уровень анализа и хранения данных благодаря plug-and-play устройствам, входящим в семейство продуктов QRadar.

QRadar SIEM осуществляет поддержку свыше 200 продуктов практически всех ведущих вендоров, обеспечивая сбор, анализ и корреляцию данных через широкий спектр систем (включая сетевые решения, средства безопасности, серверы, хосты, операционные системы и приложения). Помимо этого, QRadar SIEM может быть легко настроен для поддержки любых проприетарных приложений и новых систем. В настоящее время поддерживаются устройства следующих компаний-разработчиков: Cisco, Checkpoint, Juniper, Nortel, Oracle, Symantec, ISS/IBM, McAfee, RSA, Sun, Enterasys, Sourcefire и многие другие. [12]

Neurodat SIEM – решение, представленное российской компанией ООО «Центр безопасности информации».

Основные преимущества продукта:

- масштабирование — возможность любого масштабирования и организации иерархии системы, наличие единого центра управления;
- автономность решения — технология работы не предусматривает выдачу информации во внешние системы разработчиков средств;
- отчуждаемость решения — реализован полный отчуждаемый инструментарий;

- возможность использования нестандартных источников событий информации — подключение нестандартных источников событий через механизм разработки специализированного коннектора;
- управление инцидентами — встроенная полнофункциональная подсистема регистрации и отработки инцидентов. Автоматизирован регламент расследования инцидентов;
- контроль работоспособности внешних систем — поставщиков событий;
- импортозамещение — обеспечивает существенную независимость от импортных технологий и колебания курсов мировых валют, возможность доработки и адаптации ПО в любое время в соответствии с потребностями потенциального заказчика.

Дополнительные преимущества:

- интуитивный русскоязычный интерфейс;
- расследование инцидентов по конкретным пользователям и хостам;
- мониторинг действий пользователей в масштабе времени, близком к реальному;
- фильтрация событий от различных источников на входе в систему (избавление от массового «мусора»);
- техническая поддержка системы с возможностью ускоренной разработки дополнительной функциональности;
- развитый механизм управления отработкой инцидентов;
- широкий спектр различных метрик и показателей эффективности процессов отработки инцидентов;
- предустановленные правила корреляции событий (сопоставления).

Функциональные возможности:

- сбор и централизованная обработка информации;

- интеграция и хранение в нормализованном виде в едином хранилище данных;
- автоматизированное формирование различных типов инцидентов ИБ на основе анализа и корреляции (сопоставления) событий, поступающих от различных источников;
- сбор всех событий о запущенных процессах с АРМ и серверов;
- проведение инвентаризации сети;
- обеспечение механизмов контроля работоспособности систем — поставщиков информации для Neurodat SIEM;
- визуализация компьютерных атак на масштабируемой геоинформационной основе;
- проведение периодического анализа защищенности сети;
- оценка выполнения требований безопасности (compliance);
- встроенные механизмы автоматизации выполнения регламентов реагирования на различные типы инцидентов. [13]

Архитектура системы представлена на рисунке 14.

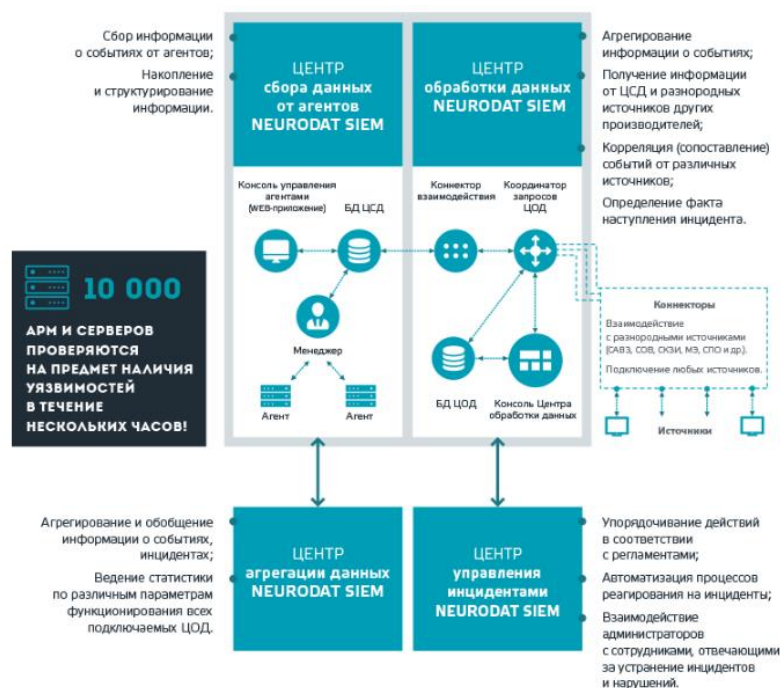


Рисунок 14 – Архитектура Neurodat SIEM

1.3 Результаты обзора и сравнительного анализа

Проведенный обзор и сравнительный анализ подходов по обработке данных показал, что наиболее предпочтительным подходом по обработке инцидентов ИБ является OLAP. Его преимуществами являются:

- скорость на стороне клиента, так как отсутствует необходимость постоянного обновления БД, как при использовании методов классификации и кластеризации Data Mining;
- данный метод наиболее проработан для визуального анализа данных.

Проведенный сравнительный анализ средств и систем по обработке инцидентов ИБ и визуального представления информации об инцидентах ИБ дал следующие результаты:

- проблеме визуализации данных современные системы мониторинга ИБ уделяют большое внимание, так как при помощи визуального анализа можно определить закономерности появления событий и инцидентов ИБ, а также их взаимосвязь, что затруднительно сделать при помощи только одного статистического анализа данных;
- результаты визуального анализа могут быть использованы при создании правил анализа в сигнатурном анализе, а также они могут послужить в качестве критериев для перестройки эталонов при статистическом анализе данных о событиях и инцидентах ИБ;
- практически во всех современных системах мониторинга ИБ существует проблема одновременного отображения большого объема информации, поэтому наиболее оптимальным решением данной проблемы является применение подхода OLAP.

2 Описание Единой системы мониторинга информационной безопасности и места подсистемы визуального анализа в Единой системе мониторинга информационной безопасности

2.1 Общее описание Единой системы мониторинга информационной безопасности

ЕСМИБ ТУ предназначена для автоматизации контроля выполнения требований нормативных и иных актов Банка России по ИБ на объекте автоматизации пользователями АС, в том числе в рамках проведения процедур внутреннего контроля уровня территориального учреждения Банка России (далее по тексту – ТУ Банка России).

Объектом автоматизации является совокупность автоматизированных банковских систем (АБС) и средств защиты информации (СЗИ), функционирующих в платежном сегменте локальной вычислительной сети (ЛВС) ТУ Банка России.

ЕСМИБ ТУ обеспечивает следующие виды деятельности по регистрации, обработке, анализу и хранению данных о событиях ИБ и коррелированных событий ИБ (КСИБ):

- процесс мониторинга ИБ объекта автоматизации;
- контроль действий пользователей АБС;
- взаимоувязанный анализ событий ИБ от разнотипных объектов из состава АБС и СЗИ;
- формирование сообщений о событиях ИБ и сообщений о КСИБ;
- регистрация и анализ событий ИБ и КСИБ;
- хранение событий ИБ и КСИБ;
- обработка и доведение до персонала объекта автоматизации информации о КСИБ, своевременный учет и выработка рекомендаций для принятия мер по уменьшению негативного влияния КСИБ;
- формирование отчетов и их графическое представление.

Подконтрольными объектами ЕСМИБ ТУ являются технические средства, системное и прикладное программное обеспечение объекта автоматизации, функционирующие в платежном сегменте ЛВС ТУ Банка России.

Перечень подконтрольных объектов ЕСМИБ ТУ:

- региональная автоматизированная банковская информационная система (РАБИС-НП);
- типовая автоматизированная система эмиссионных и кассовых работ в трехзвенной архитектуре (АС ЭКР 2.0);
- активное сетевое оборудование и средства обеспечения сетевой безопасности фирмы Cisco Systems и других производителей;
- операционная система (ОС) Microsoft Windows Workstation/Server (2000/XP/2003/ Vista/2008/7);
- система управления базами данных (СУБД) Microsoft SQL Server (2000/2005/2008);
- СУБД Oracle (8/9/10);
- СЗИ от несанкционированного доступа (НСД) Secret Net (5/6);
- СЗИ от НСД «Аккорд-NT/2000», «Аккорд-Win32», «Аккорд-Win64»;
- антивирус Dr.Web (6 и выше);
- антивирус Касперского (6/8 и выше), Kaspersky Endpoint Security (8 и выше), Kaspersky Administration Kit (6/8 и выше), Kaspersky Security Center (9 и выше);
- Symantec Endpoint Protection (11/12 и выше);
- программный комплекс «Паспорт АРМ УОС» («Технос-К»);
- программное средство (ПС) «Паспорта ПО» («Прогноз»);
- региональная система управления правами доступа пользователей к информационным системам (РСУД) в части предоставления сведений о пользователях, с учетом данных полученных РСУД из автоматизированной

системы «Внутрихозяйственная деятельность» (АС ВХД) и Единой службы каталогов Банка России (ЕСК);

- интегрированная система управления телекоммуникационными и информационными ресурсами (ИСУ ТИР) Банка России.

Первичные события мониторинга, зафиксированные на подконтрольных объектах, являются входными данными для ЕСМИБ ТУ.

Основными целями создания ЕСМИБ ТУ являются:

- повышение эффективности управленческой деятельности и внутреннего контроля уровня ТУ Банка России;

- сокращение временных затрат персонала при выполнении мониторинга ИБ на объекте автоматизации;

- контроль действий пользователей АС на объекте автоматизации;

- контроль выполнения требований нормативных и иных актов Банка России по ИБ на объекте автоматизации;

- регистрация первичных событий мониторинга, формирование сообщений о событиях ИБ, фиксация коррелированных событий ИБ на основе правил анализа событий ИБ от разнотипных источников данных мониторинга объекта автоматизации, обеспечение возможности создания и редактирования, а также проверки описаний коррелированных событий ИБ на основе информации о событиях ИБ или последовательностей событий ИБ;

- обеспечение централизованного хранения зафиксированной информации о событиях ИБ и коррелированных событиях ИБ, автоматизация извещения персонала о зафиксированных событиях ИБ и коррелированных событиях ИБ;

- документационное обеспечение процедур внутреннего контроля действий администраторов и пользователей на объекте автоматизации: обеспечение возможности создания настраиваемых отчетов о событиях ИБ и коррелированных событиях ИБ;

— создание типовых проектных решений, предназначенных для использования в ТУ Банка России.

В состав функций, выполнение которых автоматизируется с помощью специализированного ПО ЕСМИБ ТУ, входят:

— сбор первичных событий мониторинга из журналов штатных подсистем регистрации событий (ШПРС) подконтрольных объектов, перечисленных ранее, а также их регистрация;

— фиксация первичных событий мониторинга о работе персонала объекта автоматизации со съемными периферийными устройствами на подконтрольных объектах, работающих под управлением ОС Microsoft Windows;

— анализ первичных событий мониторинга (входных данных) из журналов ШПРС подконтрольных объектов и формирование сообщений о событиях ИБ по заданным правилам с целью оценки степени их влияния на ИБ объекта автоматизации;

— накопление результатов работы подсистемы анализа в базе данных (БД) событий ИБ ЕСМИБ ТУ (далее – БД событий ИБ);

— проведение отложенного анализа и генерация отчетов о событиях ИБ, содержащихся в подсистеме хранения;

— оперативное отображение состояния подконтрольных объектов;

— создание, редактирование, проверка и графическое представление правил анализа событий ИБ (сценариев);

— выявление КСИБ на основе правил анализа событий ИБ от разнотипных источников данных мониторинга объекта автоматизации с указанием нарушений требований нормативных и иных актов Банка России по ИБ, а также передача коррелированных событий ИБ в подсистему обработки;

- перенос и использование в целях мониторинга ИБ информации из баз данных подсистемы хранения событий ИБ и доверенных систем: РСУД, АС ЭКР 2.0, РАБИС-НП;

- оперативное информирование персонала системы о КСИБ путем передачи сообщений по электронной почте;

- регистрация, информирование, классификация по заданным критериям, а также указание и фиксирование мер по устранению последствий КСИБ в соответствии с ролевыми функциями персонала ЕСМИБ ТУ;

- хранение и резервирование КСИБ;

- подготовка отчетов по КСИБ с учетом задаваемых параметров (временной интервал, оценка КСИБ, его тип и т.п.), вывод на печать и экспорт в файл отчетов.

ЕСМИБ ТУ является трехуровневой системой и включает следующие уровни мониторинга ИБ:

- первый уровень мониторинга ИБ – уровень сбора первичных событий мониторинга и контроля отчуждения информации;

- второй уровень мониторинга ИБ – уровень формирования сообщений о событиях ИБ;

- третий уровень мониторинга ИБ – уровень формирования и обработки КСИБ.

2.2 Описание места подсистемы визуального анализа в Единой системе мониторинга информационной безопасности

Подсистема визуального анализа КСИБ предназначена для визуального анализа информации о зафиксированных инцидентах ИБ за определенный период времени. Структура ЕСМИБ ТУ с подсистемой визуального анализа приведена на рисунке 15.

Структура Единой системы мониторинга информационной безопасности и место подсистемы визуального анализа данных об инцидентах информационной безопасности

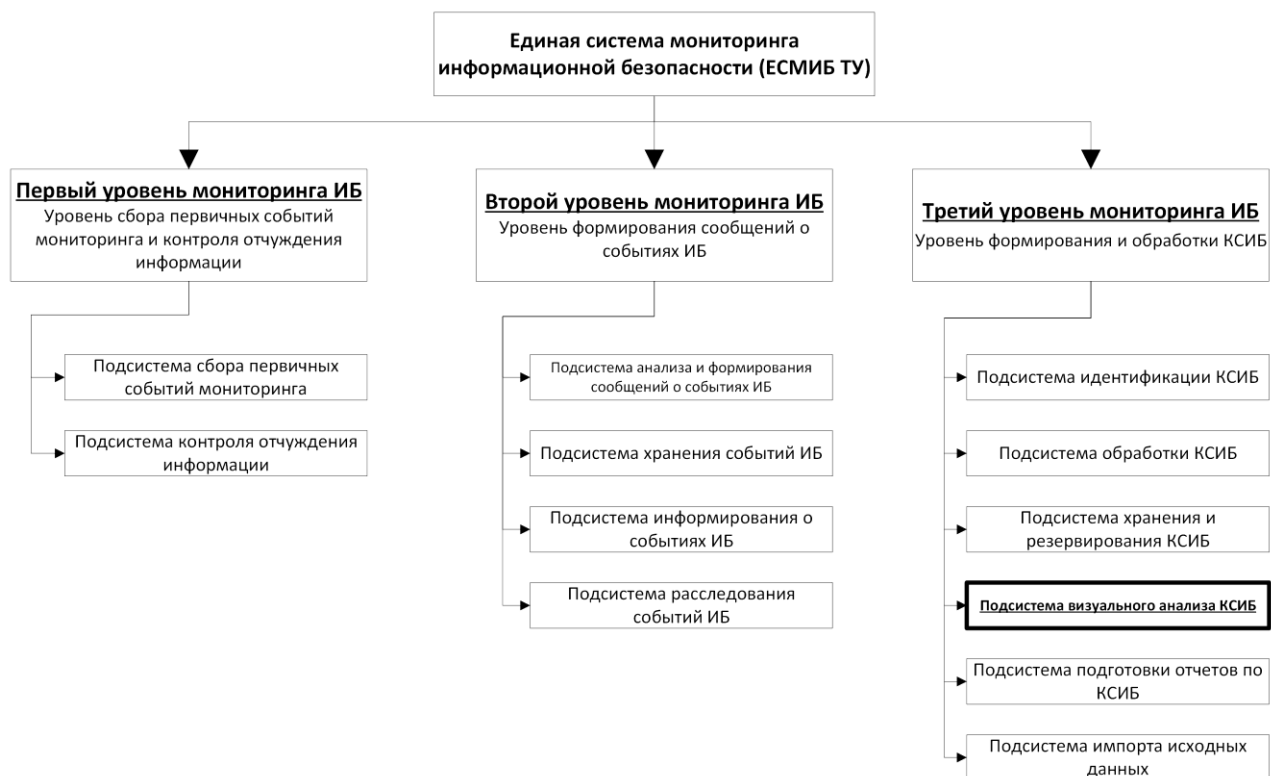


Рисунок 15 – Структура ЕСМИБ ТУ

Разрабатываемая подсистема относится к третьему уровню мониторинга ИБ. Подсистема выполняет следующие функции:

- запрос информации об инцидентах ИБ из БД об инцидентах;
- агрегация запрошенной информации за определенный период времени;
- задание необходимых для анализа осей измерения OLAP;
- табличное представление анализируемой информации;
- графическое представление.

Анализ информации об инцидентах ИБ в подсистеме осуществляется по технологии OLAP. В качестве информации, анализируемой с использованием разрабатываемых OLAP-компонентов, например, может выступать база данных о зарегистрированных инцидентах ИБ, включающая несколько таблиц:

- TIncident;
- TUser;

- THIncident;
- TUserFilters;
- TProcessing;
- TEscalation.

Фрагмент структуры БД о зарегистрированных инцидентах приведена на рисунке 16.

Структура БД об анализируемых инцидентах ИБ (фрагмент)

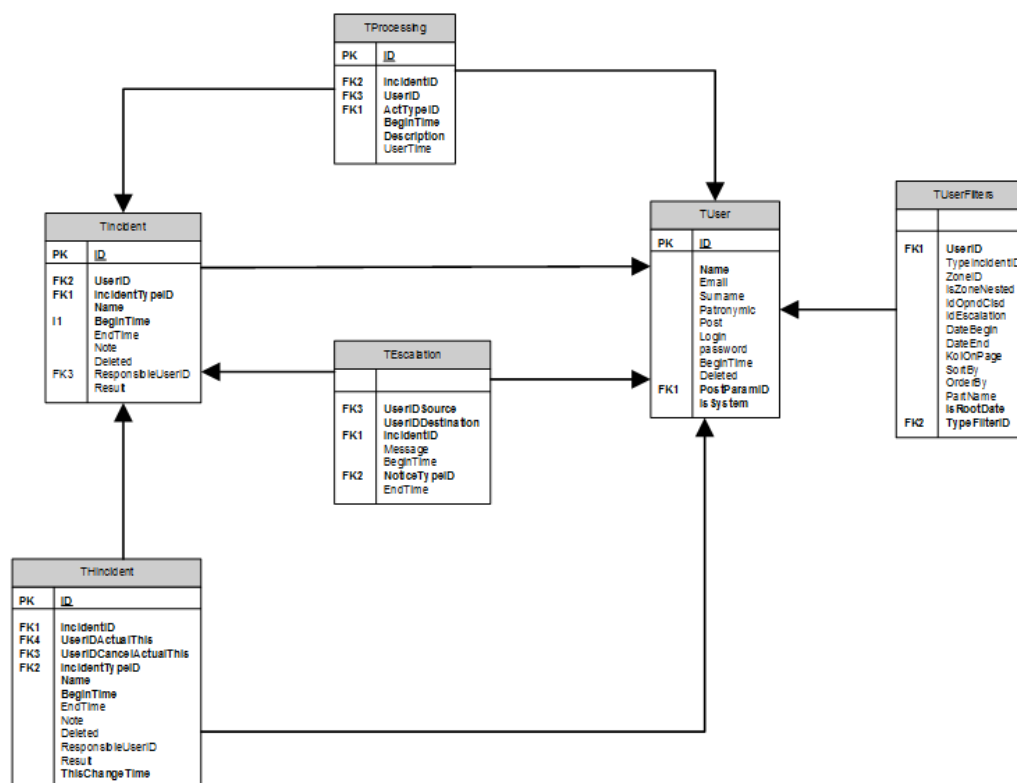


Рисунок 16 – Фрагмент БД о зарегистрированных инцидентах ИБ

Таблица TIncident в БД содержит описание КСИБ. Ее столбцами являются:

- идентификатор записи – ID;
- идентификатор пользователя-источника КСИБ – UserID;
- идентификатор типа КСИБ – IncidentTypeID;
- имя КСИБ – Name;
- дата и время регистрации КСИБ – BeginTime;

- дата и время закрытия КСИБ – EndTime;
- примечания – Note;
- признак закрытия КСИБ – Deleted;
- идентификатор пользователя, который закрыл КСИБ – ResponsibleUserID;
- вывод по закрытию – Result.

Таблица TUser содержит список пользователей. Ее столбцами являются:

- идентификатор записи – ID;
- имя пользователя – Name;
- почтовый адрес – Email;
- фамилия пользователя – Surname;
- отчество – Patronymic;
- должность – Post;
- имя пользователя в системе – Login;
- пароль пользователя в системе – password;
- дата регистрации пользователя – BeginTime;
- признак удаления – Deleted;
- идентификатор параметров почтового сервера – PostParamID;
- признак системного пользователя – IsSystem.

Таблица TIncident содержит список изменений КСИБ. Ее столбцами являются:

- идентификатор КСИБ – IncidentID;
- идентификатор пользователя внесшего изменения – UserIDActualThis;
- идентификатор пользователя отменившего изменения – CancelActualThis;
- идентификатор типа КСИБ – IncidentTypeID;

- имя КСИБ – Name;
- дата и время внесения изменений – BeginTime;
- дата и время отмены изменений – EndTime;
- описание – Note;
- признак закрытия – Deleted;
- идентификатор пользователя внесшего последние изменения – ResponsibleUserID;
- вывод – Result;
- дата и время внесения последних изменений – ThisChangeTime;
- идентификатор записи – ID.

Таблица TUserFilters содержит фильтры КСИБ. Ее столбцами являются:

- идентификатор пользователя – UserID;
- идентификатор типа инцидента – TypeIncidentID;
- идентификатор зоны – ZoneID;
- признак вложенности зоны – IsZoneNested;
- признак закрытого КСИБ – IdOpndClsd;
- признак эскалации – IdEscalation;
- начало диапазона дат регистрации – DateBegin;
- конец диапазона дат регистрации – DateEnd;
- количество на странице – KolOnPage;
- поле сортировки – SortBy;
- порядок сортировки – OrderBy;
- название КСИБ – PartName;
- признак текущей даты – IsRootDate;
- идентификатор типа фильтра – TypeFilterID.

Таблица TProcessing содержит список принятых мер. Ее столбцами являются:

- идентификатор записи – ID;
- идентификатор КСИБ – IncidentID;
- идентификатор пользователя, который принимает меру – UserID;
- идентификатор меры – ActTypeID;
- дата и время занесения записи в БД – BeginTime;
- описание принимаемой меры – Description;
- дата и время принятия меры – UserTime.

Таблица TEscalation содержит список уведомлений пользователя. Ее столбцами являются:

- идентификатор пользователя-источника уведомления – UserIDSource;
- идентификатор пользователя-получателя уведомления – UserIDDestination;
- идентификатор КСИБ – IncidentID;
- сообщение пользователю-получателю – Message;
- дата и время отправки сообщения – BeginTime;
- идентификатор типа сообщения – NoticeTypeID;
- дата и время просмотра уведомления пользователем-получателем – EndTime.

Результаты анализа данных в подсистеме могут быть представлены в табличном и графическом виде.

2.3 Необходимость использования подсистемы визуального анализа

Изучив цели, задачи и состав ЕСМИБ ТУ можно сделать следующие выводы:

- в ЕСМИБ ТУ необходимо использование подсистемы визуального анализа, основанного на использовании OLAP-компонентов, так как в ЕСМИБ ТУ обрабатывается большой объем данных, касающихся инцидентов ИБ;

- в ЕСМИБ ТУ существуют правила анализа, при создании которых могут быть использованы результаты работы с разработанной подсистемой визуального анализа;
- разработанная подсистема визуального анализа является одним из источников для составления отчета о состоянии ИБ в организации для ее руководства, так как разработанная подсистема позволяет представить информацию за определенный период (год, квартал или месяц) о произошедших инцидентах ИБ в сжатом, но информативном виде.

3 Разработка архитектуры подсистемы визуального анализа с использованием UML-диаграмм и методологии функционального моделирования IDEF

Первоначальным этапом в разработке подсистемы визуального анализа является разработка ее архитектуры. Архитектура подсистемы будет построена с использованием диаграмм, составленные на языке графического описания для объектного моделирования UML и диаграмм, построенные с использованием методологии функционального моделирования IDEF.

Методологии IDEF — методологии для решения задач моделирования сложных систем, позволяет отображать и анализировать модели деятельности широкого спектра сложных систем в различных разрезах. Для построения архитектуры подсистемы визуального анализа использовалась методология IDEF0. С помощью наглядного графического языка IDEF0 система предстает в виде набора взаимосвязанных функций (функциональных блоков). Как правило, моделирование средствами IDEF0 является первым этапом изучения любой системы. Методологию IDEF0 можно считать следующим этапом развития хорошо известного графического языка описания функциональных систем SADT (Structured Analysis and Design Technique). [13]

Модель визуального анализа представлена на рисунке 17.



Рисунок 17 – Модель визуального анализа

Подсистема визуального анализа инцидентов ИБ, основанная на OLAP-анализе, является частью системы менеджмента ИБ. В основе организации проведения OLAP-анализа инцидентов ИБ лежит циклическая модель PDCA. Анализ основывается на поиске среди данных мониторинга сигнатур инцидентов – характерных признаков, свидетельствующих о наступлении инцидента ИБ. Необходимостью обновления и добавления новых сигнатур обуславливается применение циклической модели в организации анализа. Модель организации анализа инцидентов ИБ с использованием технологии OLAP приведена на рисунке 18.



Рисунок 18 – Модель организации анализа инцидентов ИБ с использованием технологии OLAP

При планировании анализа инцидентов ИБ выполняются следующие мероприятия:

- формирование методики проведения OLAP-анализа;
- составление программы обучения лиц, ответственных за OLAP-анализ.

При осуществлении OLAP-анализа инцидентов ИБ выполняются следующие мероприятия:

- подготовка к осуществлению OLAP-анализа;

- OLAP-анализ инцидентов ИБ;
- составление отчета о произошедших инцидентах.

При проверке эффективности OLAP-анализа инцидентов ИБ выполняется анализ результатов обработки инцидентов ИБ.

При совершенствовании обработки инцидентов ИБ выполняются следующие мероприятия:

- корректировка методики проведения OLAP-анализа;
- составление требований на доработку системы проведения анализа.

На начальном этапе организации OLAP-анализа производится развертывание подсистемы визуального анализа и создание инфраструктуры, необходимой для последующей работы подсистемы. На основании описаний известных инцидентов формируются список сигнатур, на основании которых формируется методика проведения OLAP-анализа. В методике необходимо указывать список наиболее уязвимых компонентов сети для фокусировки внимания аналитика на данных, относящихся именно к этим компонентам. Также в рамках этого процесса необходимо составить программу обучения лиц, которые будут осуществлять OLAP-анализ.

Диаграмма процесса планирования OLAP-анализа инцидентов ИБ отображена на рисунке 19.



Рисунок 19 – Диаграмма процесса планирования OLAP-анализа инцидентов ИБ

На основе разработанной методики проведения визуального анализа осуществляется OLAP-анализ инцидентов ИБ.

В рамках данного процесса осуществляются:

- обучение персонала, ответственного за анализ инцидентов ИБ;
- процесс подготовки к проведению OLAP-анализа инцидентов ИБ;
- процесс анализа многомерных данных;
- процесс составления отчета о произошедших инцидентах ИБ.

Диаграмма процесса осуществления OLAP-анализа инцидентов ИБ отображена на рисунке 20.



Рисунок 20 – Диаграмма процесса проведения OLAP-анализа инцидентов ИБ

Перед проведением OLAP-анализа инцидентов ИБ, необходимо провести обучение персонала, ответственного за анализ инцидентов ИБ. Подготовка производится согласно программе обучения персонала. После прохождения подготовки необходимо произвести контроль полученных навыков по методике оценки уровня квалификации персонала. По результатам проведения обучения составляются отчеты.

Диаграмма процесса обучения персонала представлена на рисунке 21.

На начальном этапе процесса подготовки к проведению OLAP-анализа инцидентов ИБ производится инициализация подсистемы визуального анализа, осуществляется:

- проверка подлинности пользователя подсистемы;

- ввод параметров подключения к базе данных;
- ввод запроса к серверу сбора данных мониторинга;
- проверка корректности введенных параметров;

На основании этой информации производится запрос к БД мониторинга (БД инцидентов ИБ). На основе данных по этому запросу строится многомерное представление, на основе которого производится OLAP-анализ.

Диаграмма процесса подготовки к проведению OLAP-анализа инцидентов ИБ показана на рисунке 22.

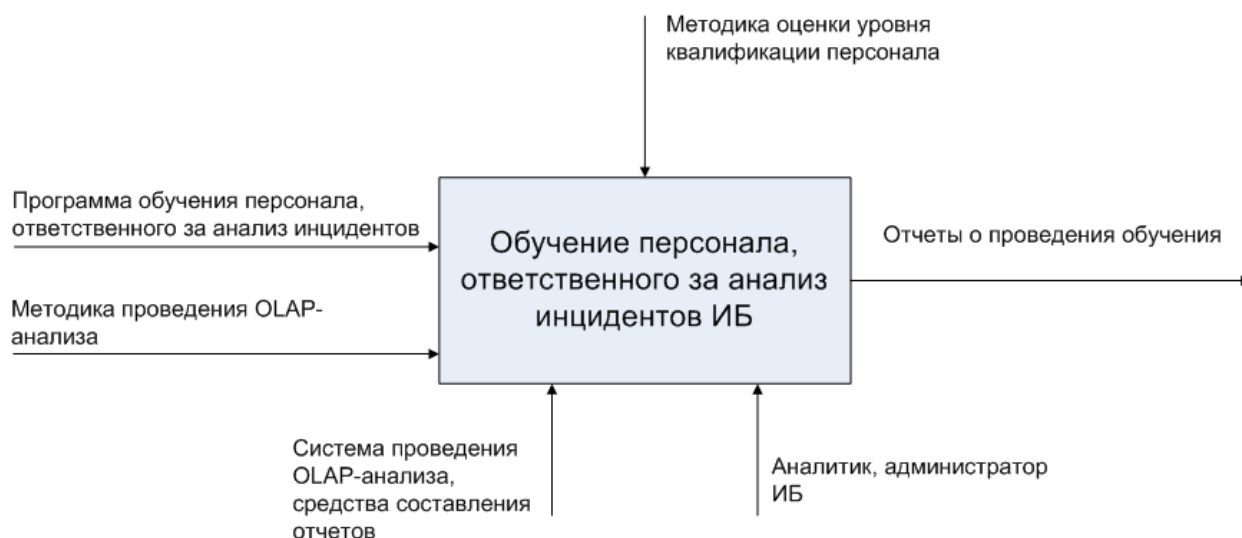


Рисунок 21 – Диаграмма процесса обучения персонала, ответственного за анализ инцидентов ИБ

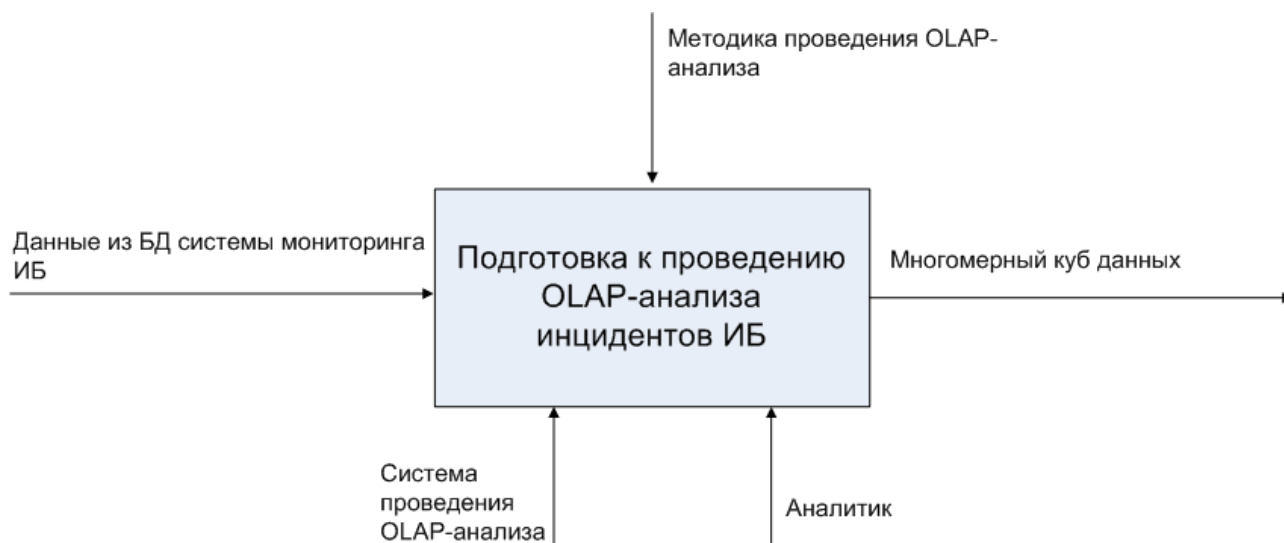


Рисунок 22 – Диаграмма процесса подготовки к проведению OLAP-анализа инцидентов ИБ

На этапе процесса анализа многомерных данных производится OLAP-анализ инцидентов. Аналитик на основании методики проведения анализа с помощью средств подсистемы осуществляет поиск сигнатур и признаков инцидентов в многомерном представлении данных мониторинга. Результатом данного процесса является информация о выявленных инцидентах и нарушениях.

Диаграмма процесса анализа многомерных данных, отображена на рисунке 23.



Рисунок 23 – Диаграмма процесса анализа многомерных данных

В завершении OLAP-анализа производится составление отчета о найденных инцидентах ИБ. Отчеты в дальнейшем используются для совершенствования подсистемы анализа инцидентов и для использования в системе менеджмента ИБ.

Диаграмма процесса составления отчетов о произошедших инцидентах ИБ представлена на рисунке 24.

Для осуществления проверки эффективности подсистемы OLAP-анализа необходимо произвести объективную и субъективную оценку подсистемы анализа.

Диаграмма процесса проверки эффективности OLAP-анализа инцидентов ИБ приведена на рисунке 25.

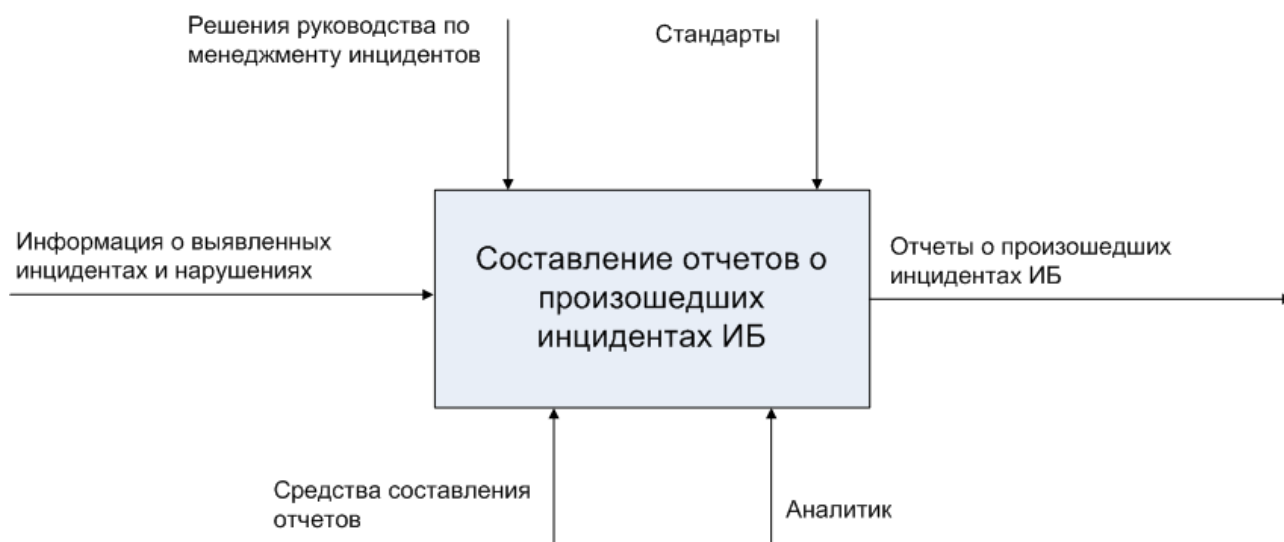


Рисунок 24 – Диаграмма процесса составления отчетов о произошедших инцидентах ИБ

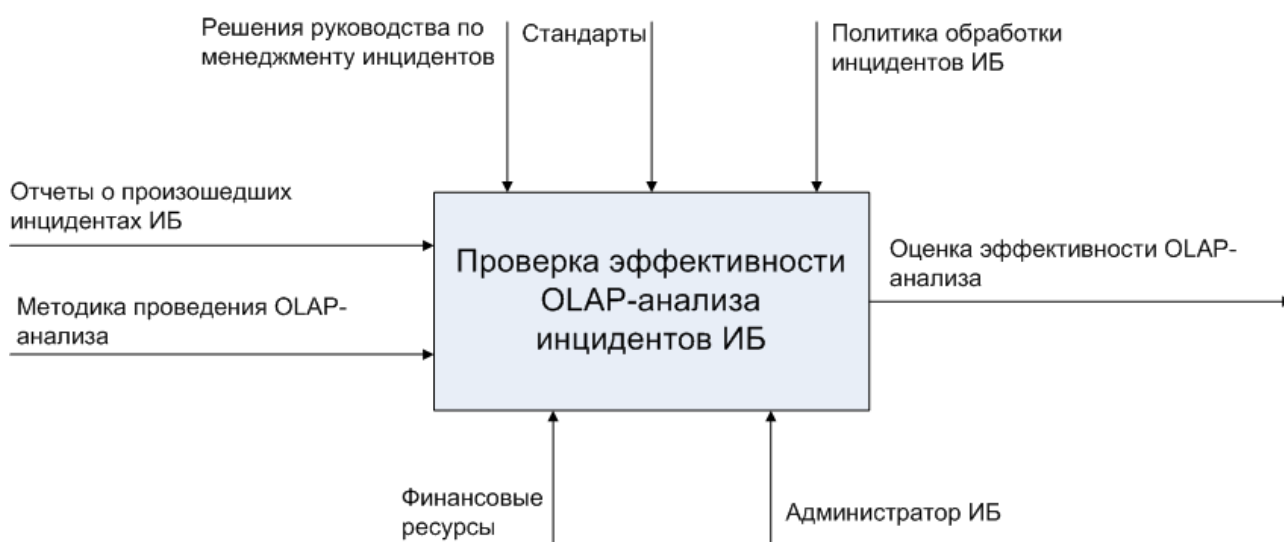


Рисунок 25 – Диаграмма процесса проверки эффективности OLAP-анализа инцидентов ИБ

На основании оценки эффективности подсистемы OLAP-анализа и отчетов о произошедших инцидентах ИБ производится совершенствование подсистемы OLAP-анализа. На основе этих данных производится корректировка методики проведения OLAP-анализа, производится дополнение и обновления сигнатур инцидентов, корректируется программа обучения лиц, ответственных за анализ инцидентов ИБ, формируются требования улучшения подсистемы OLAP-анализа.

Диаграмма совершенствования OLAP-анализа инцидентов ИБ показана на рисунке 26.

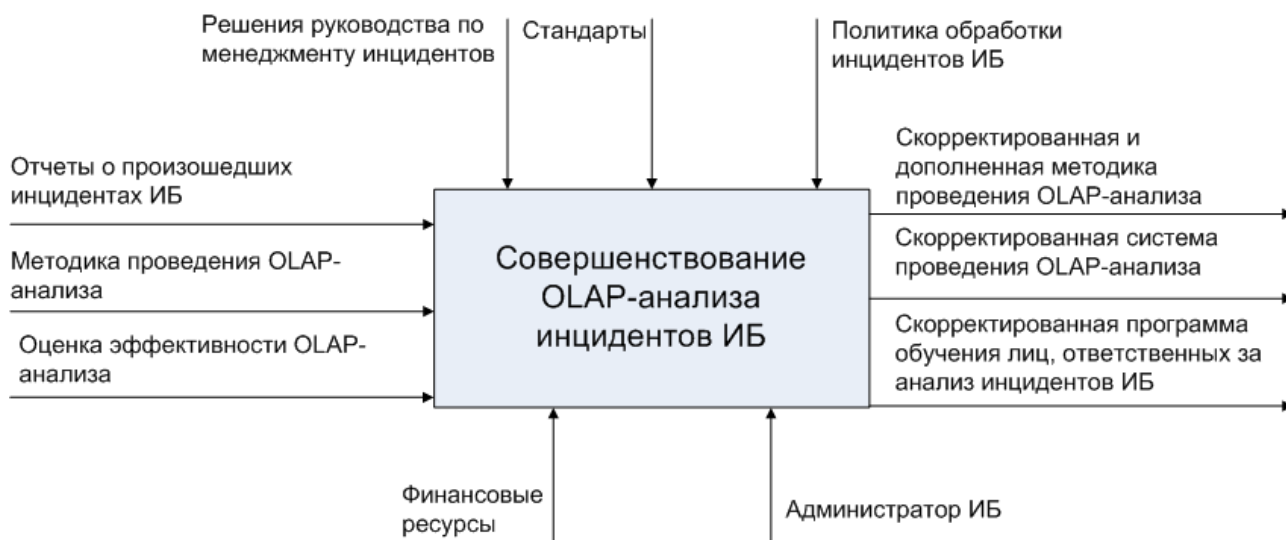


Рисунок 26 – Диаграмма процесса совершенствования OLAP-анализа инцидентов ИБ

UML (англ. Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения, моделирования бизнес-процессов, системного проектирования и отображения организационных структур.

Существуют следующие типы диаграмм:

- диаграмма классов;
- диаграмма компонентов;
- диаграмма композитной/составной структуры;
- диаграмма развёртывания;
- диаграмма объектов;
- диаграмма пакетов;
- диаграмма деятельности;
- диаграмма автомата;
- диаграмма сценариев использования;
- диаграммы коммуникации и последовательности;

- диаграмма сотрудничества;
- диаграмма обзора взаимодействия;
- диаграмма синхронизации.

При разработке архитектуры подсистемы визуального анализа были построены диаграммы развертывания и сценариев использования.

Диаграмма развёртывания (диаграмма размещения) служит для моделирования работающих узлов (аппаратных средств) и артефактов, развёрнутых на них. Между артефактом и логическим элементом (компонентом), который он реализует, устанавливается зависимость манифестации. [14]

Диаграмма развертывания приведена на рисунке 27.

Диаграмма развертывания подсистемы визуального анализа данных об инцидентах информационной безопасности

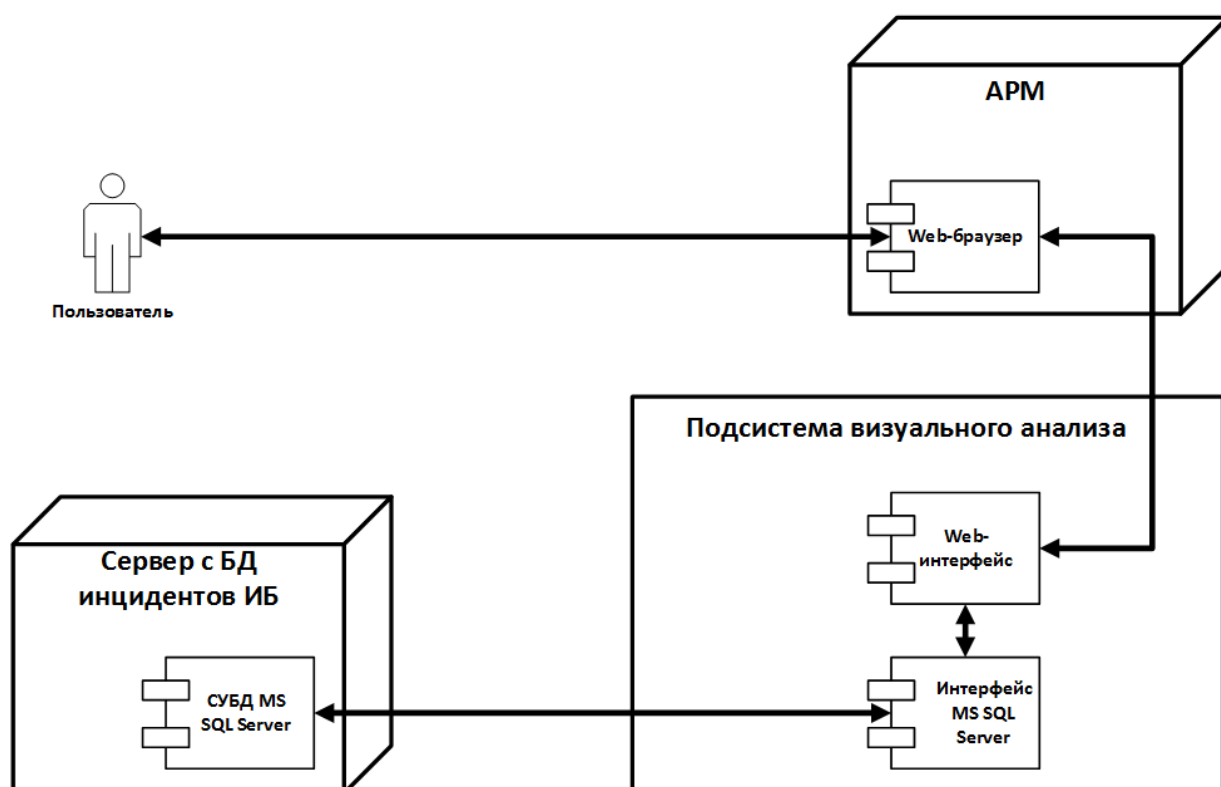


Рисунок 27 – Диаграмма развертывания подсистемы визуального анализа

Как видно из диаграммы развертывания, для работы подсистемы визуального анализа необходимы АРМ пользователя с Web-браузером, сервер с

БД инцидентов ИБ с СУБД Microsoft SQL Server. При работе подсистемы используются Web-интерфейсы и интерфейс СУБД Microsoft SQL Server.

Диаграмма сценариев использования (вариантов использования, прецедентов) – диаграмма, отражающая отношения между актёрами (множество логически связанных ролей в UML, исполняемых при взаимодействии с прецедентами или сущностями (система, подсистема или класс)) и прецедентами (спецификация последовательностей действий (варианты последовательностей и ошибочные последовательности) в UML, которые может осуществлять система, подсистема или класс, взаимодействуя с внешними действующими лицами) и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне. [14]

Диаграмма сценариев использования подсистемы визуального анализа приведена на рисунке 28 соответственно.

Диаграмма вариантов использования подсистемы визуального анализа данных об инцидентах информационной безопасности



Рисунок 28 – Диаграмма сценариев использования подсистемы визуального анализа

При работе с подсистемой визуального анализа авторизованный в ЕСМИБ ТУ пользователь выполняет следующую последовательность действий:

- выполняет запрос к БД инцидентов ИБ;
- задает необходимые для анализа оси измерения OLAP;
- получает результат анализа и выбирает табличное представление анализируемой информации или графическое представление.

Таким образом, в результате разработки архитектуры подсистемы визуального анализа была построена диаграмма, отражающая модель визуального анализа данных об инцидентах ИБ, по методологии IDEF0, диаграммы развертывания и сценариев использования подсистемы визуального анализа данных об инцидентах ИБ на языке графического описания UML.

4 Разработка программного обеспечения подсистемы визуального анализа

4.1 Общие принципы разработки

В соответствии с техническим заданием разработка программного обеспечения подсистемы визуального анализа выполняется с использованием интегрируемой среды разработки Microsoft Visual Studio 2013 на языке программирования C#, а также с использованием программной платформы Microsoft .NET Framework 4.5 и платформы для создания веб-приложение ASP.NET.

Microsoft Visual Studio – интегрированная среда разработки программного обеспечения и ряд других инструментальных средств, позволяющая разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight.

Windows .NET Framework (.NET) – это комплекс сетевых сервисов и приложений. Основой платформы является общезыковая среда исполнения Common Language Runtime (CLR), которая подходит для разных языков программирования. Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду.

ASP.NET (Active Server Pages для .NET) — платформа разработки веб-приложений, в состав которой входит: веб-сервисы, программная инфраструктура, модель программирования, от компании Майкрософт. ASP.NET входит в состав платформы .NET Framework и является развитием более старой технологии Microsoft ASP.

ASP.NET внешне во многом сохраняет схожесть с более старой технологией ASP, что позволяет разработчикам относительно легко перейти на ASP.NET. В то же время внутреннее устройство ASP.NET существенно

отличается от ASP, поскольку она основана на платформе .NET и, следовательно, использует все новые возможности, предоставляемые этой платформой.

Существенным преимуществом является тот факт, что для работы Web-приложения, созданного на базе ASP.NET, на клиентской стороне не требуется установка дополнительного программного обеспечения.

Также были использованы компоненты DevExpress версии 18.1, содержащие OLAP-компоненты. DevExpress для платформы ASP.NET представляет собой программный пакет, в состав которого входят элементы управления и библиотек для разработки Web-приложений на платформе ASP.NET. Оптимизированные Web-инструменты данного решения позволяют разрабатывать гибко настраиваемые приложения, отвечающие конкретным поставленным задачам. DevExpress ASP.NET отличается повышенной производительностью и поддерживает работу со всеми основными браузерами.

Особенности DevExpress ASP.NET:

- ASPxGridView предоставляет уникальные функции для сетки данных;
- XtraReports Suite предлагает широкие возможности для генерирования отчетности на корпоративном уровне;
- мощные инструменты для глубокого анализа данных;
- DevExpress ASP.NET Spreadsheet (beta) — удобная работа с привычными инструментами Microsoft Excel;
- инструменты для полной интеграции и упрощенного использования для Microsoft Office и Visual Studio;
- DevExpress ASP.NET HTML Editor — многофункциональный редактор HTML для WebForms, который имеет интерфейс Microsoft Word;
- DevExpress ASP.NET Scheduler — удобное представление элементов расписания Microsoft Outlook: сортировка по дням, неделям, месяцам,

временной шкале и встроенная поддержка печати, что позволяет быстро создать программу-планировщик РИМ.

Разработанное программное обеспечение является Веб-приложением. Описание этапов разработки программного обеспечения подсистемы визуального анализа приведено в параграфе 4.2.

4.2 Описание этапов разработки программного обеспечения подсистемы визуального анализа

Разработка ПО подсистемы визуального анализа состоит из следующих этапов:

- создание сервера с БД инцидентов ИБ на базе Microsoft SQL Server 2008 R2;
- создание проекта веб-приложения ASP.NET с использованием OLAP-компонентов в среде разработки Microsoft Visual Studio 2013.

На первом необходимо развернуть сервер с БД инцидентов. Для этого согласно техническому заданию был установлен Microsoft SQL Server 2008 R2. Далее был создан сервер WIN-HGM47QE3M0O\SQLEXPRESS и на нем развернута БД с инцидентами ИБ IncidentFactor2018, как показано на рисунке 29.

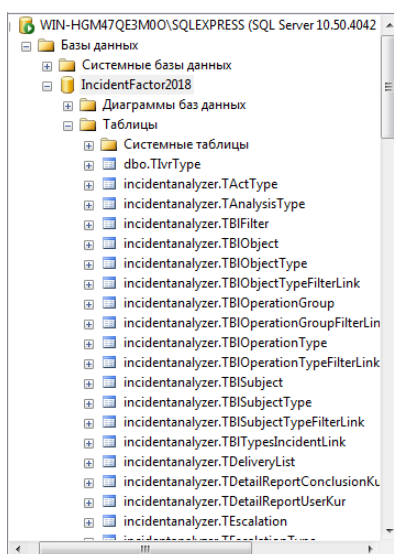


Рисунок 29 – Сервер WIN-HGM47QE3M0O\SQLEXPRESS

Далее был создан пользователь incidentanalyzer, под которым ведется работа в БД с инцидентами ИБ. SQL-инструкция приведена в листинге А.1 приложения А.

Результат создания приведен на рисунке 30.

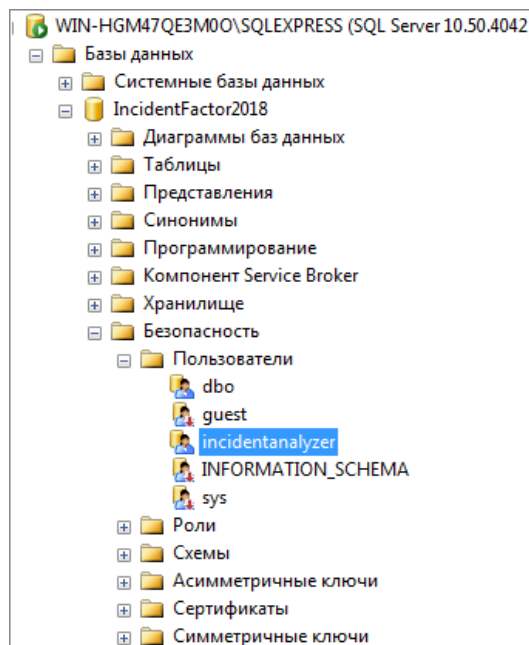


Рисунок 30 – Результат выполнения SQL-инструкции

На втором этапе для разработки программного обеспечения подсистемы визуального анализа в среде разработки Microsoft Visual Studio 2013 был создан ASP.NET проект веб-приложения Подсистема_Визуального_Анализа. В проекте созданного приложения была добавлена веб-форма MainPage. С помощью панели элементов в конструкторе этой форме был добавлен компонент Dev Express ASPxPivotGrid. Этот компонент будет отвечать за табличную форму представления данных. В качестве источника данных для данного компонента выступает база данных IncidentFactor2018 на сервере WIN-HGM47QE3M0O\SQLEXPRESS. Для извлечения данных для обработки был указан специальный SQL-оператор, приведенный в листинге А.2 приложения А.

Далее были настроены измерения OLAP в компоненте Dev Express ASPxPivotGrid. Настройка приведена на рисунке 31.

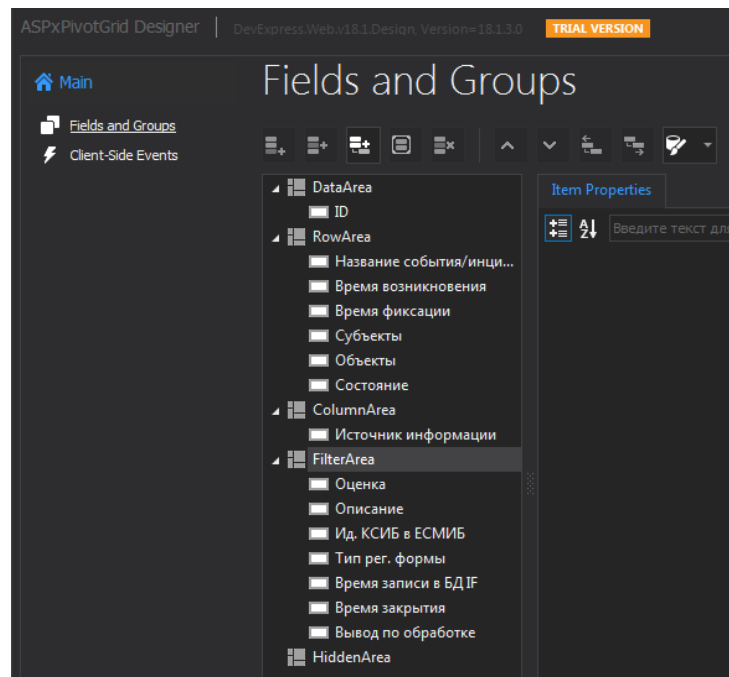


Рисунок 31 – Настройка компонента Dev Express ASPxPivotGrid

В разделе DataArea указана мера, по которой ведется подсчет инцидентов ИБ в таблице. В качестве меры было выбрано поле ID. Также была выбрана функция-счетчик Count в поднастройке SummaryType раздела «Данные» в настройках формата поля ID. Формат поля указан на рисунке 32.

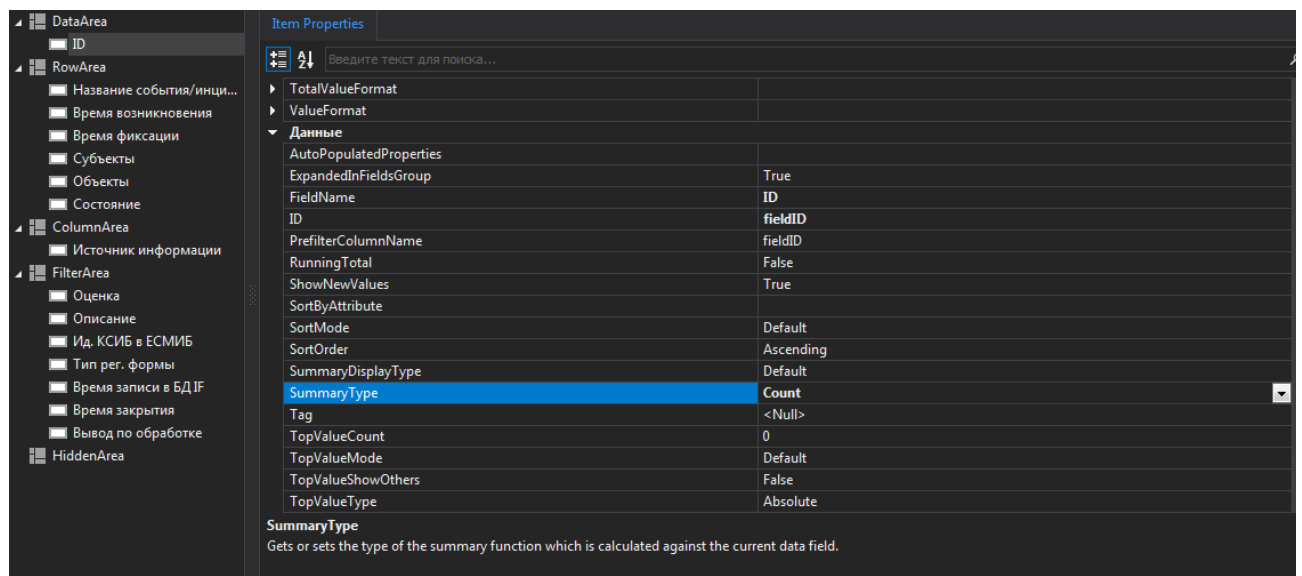


Рисунок 32 – Настройка формата поля ID

В разделах RowArea и ColumnArea указаны поля, выбранные в качестве осей измерения OLAP, которые будут выбраны по умолчанию при старте работы подсистемы визуального анализа.

Для реализации графического представления подсистемы визуального анализа была добавлена веб-форма `GrafficPage`, переход которой будет осуществляться из веб-формы, на которой расположено табличное представление подсистемы.

Для построения диаграмм по выборке из табличного представления был добавлен компонент `Dev Express WebChartControl`. В качестве источника данных для построения был указан компонент `ASPxPivodGrid`, расположенный в веб-форме `MainPage`.

В конструкторе веб-формы `MainPage` для перехода на страницу `GrafficPage` был добавлен компонент `ASP.NET Button` с текстом «Перейти к статистике».

Коды страниц `MainPage.aspx` и `GrafficPage` приведены в листинге А.3 и А.4 приложения А соответственно.

`Web.config` приведен в листинге А.5 приложения А.

Алгоритм работы подсистемы визуального анализа приведен в приложении Б.

4.3 Описание разработанного программного обеспечения подсистемы визуального анализа

Интерфейс разработанной подсистемы визуального анализа приведен на рисунке 33.

Веб-приложение состоит из 2-х форм:

- табличного представления;
- графического представления.

В табличном представлении интерфейс представлен в виде таблицы, в которой пользователь может задать необходимые для него оси измерения OLAP. Для отображения необходимой информации по умолчанию заданы следующие оси:

- источник информации;

- название события/инцидента;
- время возникновения;
- время фиксации;
- субъекты;
- объекты;
- состояние (статус инцидента).

Добро пожаловать в Подсистему Визуального Анализа КСИБ ЕСМИБ ТУ

Страница 1 из 23 (Всего элементов: 221)

Оценка

Описание

Ид. КСИБ в ЕСМИБ

Тип рег. формы

Время записи в БД IF

Время закрытия

Вывод по обработке

ID

Источники информации

Название события/инцидента

Время возникновения

Время фиксации

Субъекты

Объекты

Состояние

1

11

22

33

4

6

7

8

ЕСМИБ 191779

ЕСМИБ 191780

ЕСМИБ 191781

Подсистема идентификации КСИБ – автоматический режим

⊙ ЕСМИБ 191779

⊙ 02.11.2017 19:20:14

⊙ 02.11.2017 19:20:14

⊙ ЕСМИБ 191779

⊙ ЕСМИБ 191779

Закрыт

1

⊙ ЕСМИБ 191780

⊙ 02.11.2017 19:19:46

⊙ 02.11.2017 19:19:46

⊙ ЕСМИБ 191780

⊙ ЕСМИБ 191780

Закрыт

1

⊙ ЕСМИБ 191781

⊙ 02.11.2017 19:18:49

⊙ 02.11.2017 19:18:49

⊙ ЕСМИБ 191781

⊙ ЕСМИБ 191781

Открыт

1

⊙ ЕСМИБ Тест

⊙ 15.11.2014 09:35:15

⊙ 15.11.2014 09:35:15

⊙ 1

⊙ 1

Закрыт

⊙ 15.11.2014 09:46:28

⊙ 15.11.2014 09:46:28

⊙ 1

⊙ 1

Закрыт

⊙ 15.11.2014 10:11:50

⊙ 15.11.2014 10:11:50

⊙ 1

⊙ 1

Закрыт

⊙ 15.11.2014 10:45:26

⊙ 15.11.2014 10:45:26

⊙ 1

⊙ 1

Закрыт

⊙ 15.11.2014 10:51:29

⊙ 15.11.2014 10:51:29

⊙ 1

⊙ 1

Закрыт

ЕСМИБ Тест всего

⊙ Не открыт ОД в ОБУиО

⊙ 04.11.2015 05:00:00

⊙ 04.11.2015 06:16:15

⊙ ;

⊙ ;

Закрыт

Итого

3

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

17

Рисунок 33 – Интерфейс разработанной подсистемы визуального анализа

В верхней части окна приложения приведены поля из БД инцидентов ИБ, которые по своему желанию пользователь может добавить для отображения необходимой информации.

Переход из табличного представления в графическое осуществляется по кнопке «Перейти к статистике».

Интерфейс графического представления представлен в виде построенных диаграмм по заданным ранее табличным представлениям.

Работа пользователя в подсистеме визуального анализа осуществляется в веб-браузере.

5 Тестирование разработанного программного обеспечения подсистемы визуального анализа

5.1 Разработка схемы тестирования разработанного программного обеспечения подсистемы визуального анализа

Целью тестирования разработанного программного обеспечения подсистемы визуального анализа является проверка соответствия техническому заданию, обнаружение и устранение недостатков, ошибок, а также разработка предложений по улучшению работы подсистемы.

Для осуществления анализа результатов функционирования программного обеспечения подсистемы визуального анализа необходимо произвести его системное тестирование. На сегодняшний день процесс тестирования включает в себя: планирование, проектирование, создание, поддержку и выполнение тестов и тестовых окружений. Для получения и анализа результатов функционирования разработанного программного обеспечения будут использоваться следующие виды тестирования:

- функциональное тестирование;
- тестирование производительности.

Функциональное тестирование включает в себя:

- тестирование функциональной пригодности;
- тестирование интерфейса подсистемы;

Тестирование производительности, в свою очередь включает в себя следующие виды тестирования:

- нагрузочное тестирование;
- тестирование стабильности.

Схема тестирования представлена на рисунке 34.

Схема тестирования программного обеспечения подсистемы визуального анализа

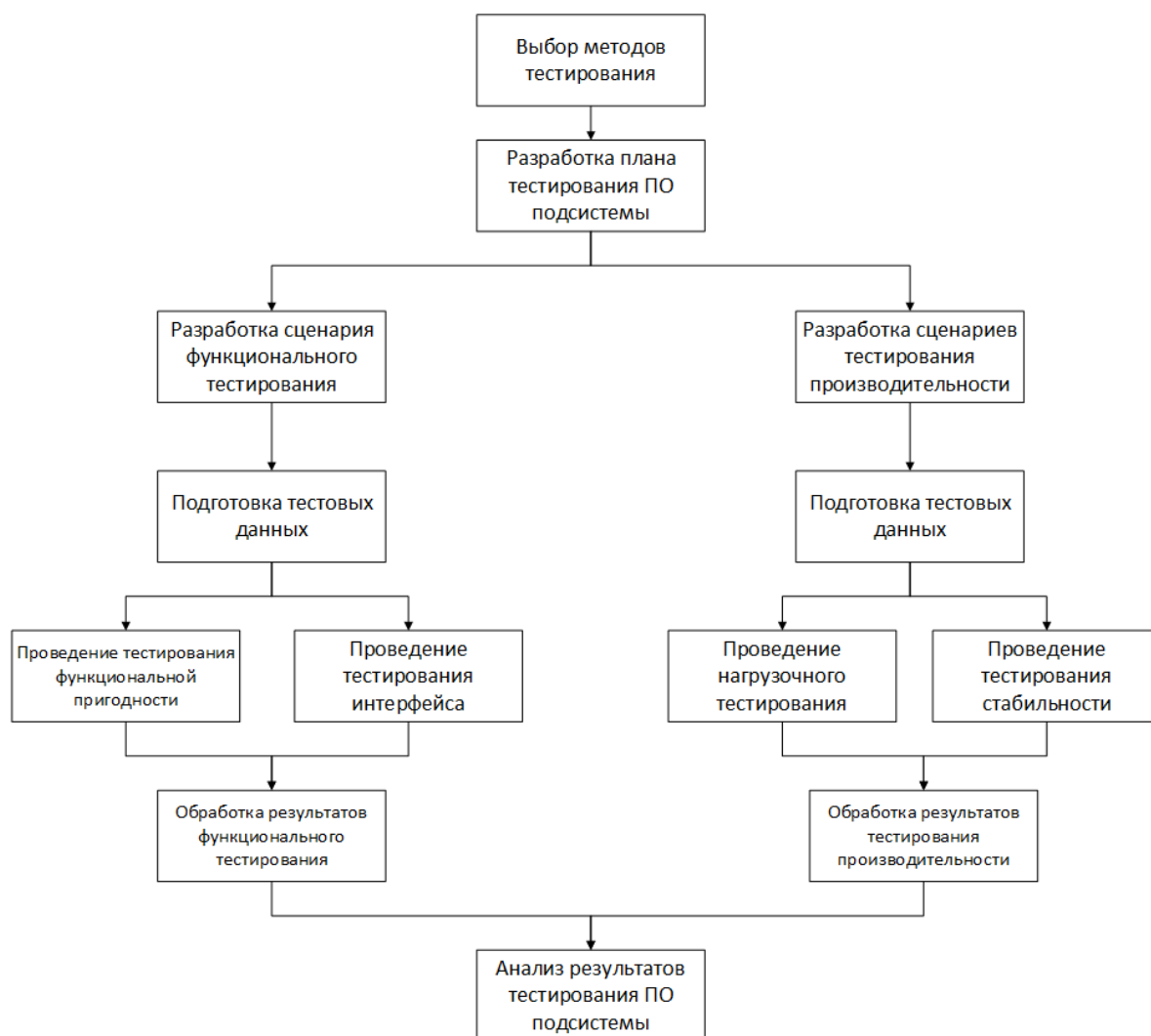


Рисунок 34 – Схема тестирования программного обеспечения подсистемы

5.2 Функциональное тестирование программного обеспечения подсистемы

Функциональное тестирование подсистемы — это тестирование подсистемы в целях проверки реализуемости функциональных требований, то есть способности подсистемы в определённых условиях решать задачи, нужные пользователям. Функциональные требования определяют, что именно делает подсистема, какие задачи оно решает.

Функциональное тестирование включает в себя следующие этапы:

- разработка сценариев;

- подготовка тестовых данных;
- проведение тестирования функциональной пригодности;
- проведение тестирования интерфейса подсистемы;
- обработка результатов функционального тестирования.

Тестирование функциональной пригодности включает в себя выполнение подсистемой требований, определенных в техническом задании. Тестирование проводится методом «черного ящика», то есть проведения тестирования осуществляется без доступа к коду программного обеспечения подсистемы визуального анализа. В качестве тестовых данных используются требования к подсистеме визуального анализа, приведенные в техническом задании. Результаты тестирования функциональной пригодности приведены в таблице 1.

Тестирование интерфейса подсистемы выполняется с целью определения удобства использования подсистемы пользователем для предполагаемого использования. Результаты тестирования интерфейса подсистемы также приведены в таблице 1.

Таблица 1 – Результаты функционального тестирования подсистемы визуального анализа

Критерий тестирования	Оценка (+/-)
Подсистема функционирует в операционных системах (ОС) Windows 7/8/8.1/10	+
Анализ информации об инцидентах ИБ осуществляется по технологии OLAP	+
Результаты анализа данных в подсистеме представлены в табличном виде	+
Результаты анализа данных в подсистеме представлены в графическом виде	+
Интерфейс подсистемы удобен пользователю для использования	+

Как видно из результатов функционального тестирования подсистемы визуального анализа удовлетворяет требованиям технического задания.

5.3 Тестирование производительности программного обеспечения подсистемы

Тестирование производительности подсистемы – это тестирование, которое проводится для определения скорости работы подсистемы при заданной нагрузке. Часто используется для проверки и подтверждения различных атрибутов качества системы:

- масштабируемость;
- надёжность;
- потребление ресурсов.

Тестирование производительности ПО подсистемы включает следующие этапы:

- разработка сценариев;
- подготовка тестовых данных;
- проведение нагрузочного тестирования;
- проведение тестирования стабильности;
- обработка результатов функционального тестирования.

Нагрузочное тестирование – это простейшая форма тестирования производительности. Нагрузочное тестирование обычно проводится для того, чтобы оценить поведение ПО подсистемы под заданной ожидаемой нагрузкой. Этой нагрузкой может быть, например, ожидаемое количество одновременно работающих пользователей приложения, совершающих заданное число запросов за интервал времени. Такой тип тестирования обычно позволяет получить время отклика всех запросов.

Результаты нагрузочного тестирования приведены в таблице 2.

Таблица 2 – Результаты нагрузочного тестирования

Количество запросов	10	20	30	40	50	60	70	80	90
Время отклика, мс	30	31	31	32	33	34	36	38	39

Тестирование стабильности проводится с целью убедиться в том, что подсистема выдерживает ожидаемую нагрузку в течение длительного времени. При проведении этого вида тестирования осуществляется наблюдение за потреблением подсистемой памяти, чтобы выявить потенциальные утечки. Также важным, но часто незамеченным, фактором является деградация производительности, смысл которого сводится к тому, чтобы скорость обработки информации и (или) время ответа подсистемы через длительное время работы были такими же или лучше, чем в самом начале теста.

Результаты тестирования стабильности приведены в таблице 3.

Таблица 3 – Результаты тестирования стабильности

Время работы подсистемы, ч	12	24	48	72
Количество отказов	0	0	0	0

Проведенное тестирование ПО подсистемы визуального анализа показало следующие результаты:

- тестирование функциональной пригодности показало, что разработанная подсистема отвечает требованиям, определенным в техническом задании на выпускную квалификационную работу;
- тестирование интерфейса подсистемы показало, что интерфейс ПО подсистемы визуального анализа удобен для предполагаемого использования пользователем ЕСМИБ ТУ;
- нагрузочное тестирование ПО показало, что подсистема визуального анализа работает при любой заданной нагрузке, то есть при любом количестве запросов;
- тестирование стабильности показало, что подсистема визуального выдерживает ожидаемую нагрузку в течение длительного времени.

6 Вредные психофизиологические факторы, влияющие на опорно-двигательный аппарат специалиста по защите информации, методы и средства физической культуры, снижающие их воздействие

Работа специалиста по защите информации, являющимся пользователем ЕСМИБ ТУ, сопровождается напряженностью труда. Наибольшее влияние на напряженность оказывают психофизиологические факторы. Психофизиологические факторы – факторы, обусловленные особенностями физиологии и психологии человека, которые могут нанести вреда при определенных обстоятельствах. Психофизиологические факторы могут влиять на следующие органы и состояние специалиста:

- опорно-двигательный аппарат;
- сосудисто-сердечная система;
- органы зрения;
- нервная система;
- психоэмоциональное состояние.

Психофизиологическими факторами, влияющими на опорно-двигательную систему, могут быть:

- постоянное неподвижное положение;
- интенсивная работа с клавиатурой;
- синдром запястного канала;
- неправильная осанка.

Постоянное неподвижное, сидячее положение специалиста по защите информации, в течение длительного времени прикованного к экрану монитора, приводит к усталости и возникновению болей в позвоночнике, шее, плечевых суставах, а также к развитию слабости мышц и изменениям формы позвоночника.

Интенсивная работа с клавиатурой вызывает болевые ощущения в локтевых суставах, предплечьях, запястьях, в кистях и пальцах рук. Часто присутствуют жалобы на онемение шеи, боль в плечах и пояснице или

покалывание в ногах. Но бывают и более серьезные заболевания. Наиболее распространен синдром запястного канала, при котором нервные окончания руки повреждаются вследствие частой и длительной работы на компьютере. В наиболее тяжелой форме этот синдром проявляется в виде мучительных болей, лишаящих человека трудоспособности.

Синдром запястного канала (СЗК) по существу представляет собой травму запястья. Патологическое состояние, называемое синдромом запястного канала, вызывается ущемлением срединного нерва в запястном канале. Оно возникает при распухании срединного нерва и/или сухожилий кисти. Чаще всего СЗК – это следствие многочасового сидения за компьютером с неправильной осанкой. Накапливающаяся травма вызывает накопление продуктов распада в области запястного канала. Если пользователь не делает регулярных перерывов и не выполняет простые упражнения для кисти, продукты распада вызывают распухание, а затем и развитие СЗК.

Неправильная осанка бывает различных типов и может привести к различным проблемам. К примеру, сгорбленное положение во время работы увеличивает и без того большую нагрузку на позвоночник, приводит к чрезмерному растяжению мышц, поддерживающих осанку. Сгорбленное положение может приводить к заболеваниям: синдром запястного канала, грыжа межпозвоночных дисков поясничного отдела, грыжа межпозвоночных дисков шейного отдела. Когда в результате неправильной осанки голова выдвинута вперед, это усиливает прогиб шейного отдела позвоночника, вытягивание головы вперед вызывает напряжение мышц основания головы и шеи, что ограничивает кровоток в сосудах шеи, кровоснабжение головы и отток крови от нее. Напряженные мышцы у основания головы и шеи при вытягивании головы вперед могут привести к таким нарушениям, как: головные боли, боль в шее, боль в руках и кистях. Причиной неправильной осанки может являться сутулость. Сутулость вызывает чрезмерную нагрузку на плечевые сухожилия,

что приводит к напряжению мышц плеча. Сутулость может привести к развитию: синдрома запястного канала, синдрома ущемления плеча.

Профилактикой проблем с опорно-двигательной системой являются:

- эргономичная организация рабочего места для сведения к минимуму рабочей нагрузки на область запястного канала.
- упражнения на растягивание и укрепление мышц запястья и ладони.
- регулярные перерывы в работе с мышью и клавиатурой.
- правильная осанка.

Заключение

В ходе дипломного проектирования была разработана подсистема визуального анализа, внедренная в ЕСМИБ ТУ. В качестве подхода обработки данных об инцидентах использован OLAP. Его преимуществами являются:

- скорость на стороне клиента, так как отсутствует необходимость постоянного обновления БД, как при использовании методов классификации и кластеризации Data Mining;
- данный метод наиболее проработан для визуального анализа данных.

Проведенный сравнительный анализ средств и систем по обработке инцидентов ИБ и визуального представления информации об инцидентах ИБ дал следующие результаты:

- проблеме визуализации данных современные системы мониторинга ИБ уделяют большое внимание, так как при помощи визуального анализа можно определить закономерности появления событий и инцидентов ИБ, а также их взаимосвязь, что затруднительно сделать при помощи только одного статистического анализа данных;
- результаты визуального анализа могут быть использованы при создании правил анализа в сигнатурном анализе, а также они могут послужить в качестве критериев для перестройки эталонов при статистическом анализе данных о событиях и инцидентах ИБ;
- практически во всех современных системах мониторинга ИБ существует проблема одновременного отображения большого объема информации, поэтому наиболее оптимальным решением данной проблемы является применение подхода OLAP.

Изучив цели, задачи и состав ЕСМИБ ТУ были сделаны следующие выводы по актуальности использования подсистемы визуального анализа в системе:

- в ЕСМИБ ТУ необходимо использование подсистемы визуального анализа, основанного на использовании OLAP-компонентов, так как в ЕСМИБ ТУ обрабатывается большой объем данных, касающихся инцидентов ИБ;

- в ЕСМИБ ТУ существуют правила анализа, при создании которых могут быть использованы результаты работы с разработанной подсистемой визуального анализа;

- разработанная подсистема визуального анализа является одним из источников для составления отчета о состоянии ИБ в организации для ее руководства, так как разработанная подсистема позволяет представить информацию за определенный период (год, квартал или месяц) о произошедших инцидентах ИБ в сжатом, но информативном виде.

Была разработана архитектура подсистемы с использованием методологии IDEF0 и UML-диаграмм.

Было разработано ПО подсистемы визуального анализа и проведено тестирование, которое показало, что подсистема соответствует требованиям технического задания.

А также были изучены вредные психофизиологические факторы, влияющие на опорно-двигательный аппарат специалиста по защите информации, методы и средства физической культуры, снижающие их воздействие.

Таким образом, дипломное проектирование выполнено в полном объеме.

Список используемых источников

- 1 OLAP [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/OLAP>, свободный. – Загл. с экрана
- 2 OLAP-куб [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/OLAP-%D0%BA%D1%83%D0%B1>, свободный. – Загл. с экрана
- 3 Data Mining [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/Data_mining, свободный. – Загл. с экрана
- 4 Data Mining [Электронный ресурс]. Режим доступа: <https://basegroup.ru/community/articles/data-mining>, свободный. – Загл. с экрана
- 5 Data Mining. Классификация и кластеризация [Электронный ресурс]. Режим доступа: <https://studfiles.net/preview/5554364/page:17/>, свободный. – Загл. с экрана
- 6 Кластерный анализ [Электронный ресурс]. Режим доступа: https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D0%B0%D1%81%D1%82%D0%B5%D1%80%D0%BD%D1%8B%D0%B9_%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D0%B7, свободный. – Загл. с экрана
- 7 SIEM [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/SIEM>, свободный. – Загл. с экрана
- 8 Infowatch Vision [Электронный ресурс]. Режим доступа: <https://www.infowatch.ru/products/vision/features>, свободный. – Загл. с экрана
- 9 ArcSight Russia [Электронный ресурс]. Режим доступа: <http://arcsight-russia.ru/products-hp-arcsight/products-hp-arcsight>, свободный. – Загл. с экрана
- 10 Splunk Enterprise [Электронный ресурс]. Режим доступа: https://www.splunk.com/ru_ru/products/splunk-enterprise.html, свободный. – Загл. с экрана
- 11 IBM i2 [Электронный ресурс]. Режим доступа: <http://security-zone.ru/blog/awards/62.html>, свободный. – Загл. с экрана

12 IBM QRadar [Электронный ресурс]. Режим доступа: <https://www.ibm.com/ru-ru/marketplace/ibm-qradar-siem/details#product-header-top>, свободный. – Загл. с экрана

13 Neurodat SIEM [Электронный ресурс]. Режим доступа: <http://neurodat.ru/>, свободный. – Загл. с экрана

14 IDEF [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/IDEF>, свободный. – Загл. с экрана

15 UML [Электронный ресурс]. Режим доступа: <https://ru.wikipedia.org/wiki/UML>, свободный. – Загл. с экрана

Приложение А

(обязательное)

Коды программного обеспечения подсистемы визуального анализа данных

Листинг A.1 – Оператор SQL, используемый для создания пользователя incidentanalyzer

```
if not exists (select * from master.dbo.syslogins where loginname
= N'incidentanalyzer')
BEGIN
    declare @logindb nvarchar(132), @loginlang nvarchar(132)
    select @logindb = N'incidentanalyzer', @loginlang = N'us_english'
    if @logindb is null or not exists (select * from
master.dbo.sysdatabases where name = @logindb)
        select @logindb = N'master'
    if @loginlang is null or (not exists (select * from
master.dbo.syslanguages where name = @loginlang) and @loginlang <>
N'us_english')
        select @loginlang = @@language

    exec sp_addlogin @loginname = N'incidentanalyzer',
        @passwd = N'crystall',
        @defdb = @logindb,
        @deflanguage = @loginlang
END
GO
exec sp_addsrvrolemember @loginname = N'incidentanalyzer' ,
    @rolename = 'dbcreator'
GO
sp_change_users_login @Action = N'Update_One',
    @UserNamePattern =
N'incidentanalyzer',
    @LoginName =
N'incidentanalyzer'
GO
```

Листинг А.2 – Оператор SQL, используемый для извлечения данных с БД ИНЦИДЕНТ

```

SELECT DISTINCT
            incidentanalyzer.TIncident.ID,
incidentanalyzer.returnDefaultNameIncidentIfNull(incidentanalyzer.
TIncident.Name) AS [Название события/инцидента],
            CC0.Val AS [Время возникновения], CC1.Val
AS [Время фиксации], CC2.Val AS [Источник информации], CC3.Val AS
Оценка, CC4.Val AS Описание,
            CC5.Val AS Субъекты, CC6.Val AS Объекты,
CC7.Val AS [Ид. КСИБ в ЕСМИБ], incidentanalyzer.TIncidentType.Name
AS [Тип рег. формы],

incidentanalyzer.returnDateOrValueDefault(incidentanalyzer.TIncide
nt.BeginTime) AS [Время записи в БД IF],

incidentanalyzer.returnOpenedOrClosedOrEscalation(incidentanalyzer
.TIncident.Deleted, AAA.NoticeTypeID) AS Состояние,

incidentanalyzer.returnDateOrValueDefault(incidentanalyzer.TIncide
nt.EndTime) AS [Время закрытия],
            incidentanalyzer.TIncident.Result          AS
[Вывод по обработке]
FROM            incidentanalyzer.TIncident INNER JOIN
            incidentanalyzer.TIncidentType          ON
incidentanalyzer.TIncident.IncidentTypeID          =
incidentanalyzer.TIncidentType.ID LEFT OUTER JOIN
            incidentanalyzer.TUser                  ON
incidentanalyzer.TUser.ID = incidentanalyzer.TIncident.UserID LEFT
OUTER JOIN
            incidentanalyzer.TEscalation AS AAA      ON
AAA.IncidentID      =      incidentanalyzer.TIncident.ID      AND
AAA.NoticeTypeID = 2 INNER JOIN
            incidentanalyzer.TForm                  ON
incidentanalyzer.TForm.IncidentTypeID              =
incidentanalyzer.TIncidentType.ID INNER JOIN
            incidentanalyzer.TFormElementRank        ON
incidentanalyzer.TFormElementRank.FormID           =
incidentanalyzer.TForm.ID LEFT OUTER JOIN
            incidentanalyzer.TFormElementValue AS CC0
ON      incidentanalyzer.TIncident.ID      =      CC0.IncidentID      AND
CC0.FormElementID IN
            (SELECT DISTINCT ID + 1 AS Expr1
            FROM
incidentanalyzer.TFormElement
            WHERE      (ElementTypeID = 3)
AND (Value = 'Время возникновения')) LEFT OUTER JOIN
            incidentanalyzer.TFormElement AS DD0      ON
CC0.FormElementID = DD0.ID LEFT OUTER JOIN

```

```

                                incidentanalyzer.TFormElementValue AS CC1
ON      incidentanalyzer.TIncident.ID      =      CC1.IncidentID      AND
CC1.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE      (ElementTypeID = 3)
AND (Value = 'Время фиксации')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD1 ON
CC1.FormElementID = DD1.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC2
ON      incidentanalyzer.TIncident.ID      =      CC2.IncidentID      AND
CC2.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE      (ElementTypeID = 3)
AND (Value = 'Источник информации')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD2 ON
CC2.FormElementID = DD2.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC3
ON      incidentanalyzer.TIncident.ID      =      CC3.IncidentID      AND
CC3.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE      (ElementTypeID = 3)
AND (Value = 'Оценка')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD3 ON
CC3.FormElementID = DD3.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC4
ON      incidentanalyzer.TIncident.ID      =      CC4.IncidentID      AND
CC4.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE      (ElementTypeID = 3)
AND (Value = 'Описание')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD4 ON
CC4.FormElementID = DD4.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC5
ON      incidentanalyzer.TIncident.ID      =      CC5.IncidentID      AND
CC5.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE      (ElementTypeID = 3)
AND (Value = 'Субъекты')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD5 ON
CC5.FormElementID = DD5.ID LEFT OUTER JOIN

```

```

                                incidentanalyzer.TFormElementValue AS CC6
ON      incidentanalyzer.TIncident.ID      =      CC6.IncidentID      AND
CC6.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE                                (ElementTypeID = 3)
AND (Value = 'Объекты')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD6 ON
CC6.FormElementID = DD6.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC7
ON      incidentanalyzer.TIncident.ID      =      CC7.IncidentID      AND
CC7.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE                                (ElementTypeID = 3)
AND (Value = 'Ид. КСИБ в ЕСМИБ')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD7 ON
CC7.FormElementID = DD7.ID
WHERE      (incidentanalyzer.TIncidentType.ID = 243)
ORDER BY [Время записи в БД IF] DESCов БД ЕСМИБ ТУ

```

Листинг А.3 – Код страницы MainPage.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="MainPage.aspx.cs"
Inherits="Подсистема_Визуального_Анализа.MainPage" %>

<%@ Register assembly="DevExpress.Web.ASPxPivotGrid.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a"
namespace="DevExpress.Web.ASPxPivotGrid" tagprefix="dx" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-
8"/>
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>

<asp:Label ID="Label1" runat="server"
BorderColor="#333333" BorderStyle="None" Font-Names="HP Simplified
Light" Font-Overline="False" Font-Size="X-Large" Text="Добро
пожаловать в Подсистему Визуального Анализа КСИБ ЕСМИБ
ТУ"></asp:Label>

</div>
<dx:ASPxPivotGrid ID="ASPxPivotGrid2" runat="server"
ClientIDMode="AutoID" DataSourceID="BD_IIB_ESMIB_TU"
Theme="Metropolis">
<Fields>
<dx:PivotGridField ID="fieldID" Area="DataArea"
AreaIndex="0" FieldName="ID" Name="fieldID" Options-
ShowInFilter="True" SummaryType="Count">
</dx:PivotGridField>
<dx:PivotGridField ID="field" Area="RowArea"
AreaIndex="0" FieldName="Название события/инцидента" Name="field"
Options-ShowInFilter="True">
</dx:PivotGridField>
<dx:PivotGridField ID="field1" Area="RowArea"
AreaIndex="1" FieldName="Время возникновения" Name="field1"
Options-ShowInFilter="True">
</dx:PivotGridField>
<dx:PivotGridField ID="field2" Area="RowArea"
AreaIndex="2" FieldName="Время фиксации" Name="field2" Options-
ShowInFilter="True">
</dx:PivotGridField>
```

```

        <dx:PivotGridField ID="field3" Area="RowArea"
AreaIndex="3" FieldName="Субъекты" Name="field3" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field4" Area="RowArea"
AreaIndex="4" FieldName="Объекты" Name="field4" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field5" Area="RowArea"
AreaIndex="5" FieldName="Состояние" Name="field5" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field6" Area="ColumnArea"
AreaIndex="0" FieldName="Источник информации" Name="field6"
Options-ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field7" AreaIndex="0"
FieldName="Оценка" Name="field7" Options-ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field8" AreaIndex="1"
FieldName="Описание" Name="field8" Options-ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field9" AreaIndex="2"
FieldName="Ид. КСИБ в ЕСМИБ" Name="field9" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field10" AreaIndex="3"
FieldName="Тип пер. формы" Name="field10" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="fieldIF" AreaIndex="4"
FieldName="Время записи в БД IF" Name="fieldIF" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field11" AreaIndex="5"
FieldName="Время закрытия" Name="field11" Options-
ShowInFilter="True">
        </dx:PivotGridField>
        <dx:PivotGridField ID="field12" AreaIndex="6"
FieldName="Вывод по обработке" Name="field12" Options-
ShowInFilter="True">
        </dx:PivotGridField>
    </Fields>
</dx:ASPxPivotGrid>
<asp:SqlDataSource ID="BD_IIB_ESMIB_TU" runat="server"
ConnectionString="<%$
ConnectionStrings:IncidentFactor2018ConnectionString %>"
SelectCommand="SELECT DISTINCT
            incidentanalyzer.TIncident.ID,
incidentanalyzer.returnDefaultNameIncidentIfNull(incidentanalyzer.
TIncident.Name) AS [Название события/инцидента],

```

```

        CC0.Val AS [Время возникновения], CC1.Val
AS [Время фиксации], CC2.Val AS [Источник информации], CC3.Val AS
Оценка, CC4.Val AS Описание,
        CC5.Val AS Субъекты, CC6.Val AS Объекты,
CC7.Val AS [Ид. КСИБ в ЕСМИБ], incidentanalyzer.TIncidentType.Name
AS [Тип рег. формы],

incidentanalyzer.returnDateOrValueDefault(incidentanalyzer.TIncide
nt.BeginTime) AS [Время записи в БД IF],

incidentanalyzer.returnOpenedOrClosedOrEscalation(incidentanalyzer
.TIncident.Deleted, AAA.NoticeTypeID) AS Состояние,

incidentanalyzer.returnDateOrValueDefault(incidentanalyzer.TIncide
nt.EndTime) AS [Время закрытия],
        incidentanalyzer.TIncident.Result AS
[Вывод по обработке]
FROM          incidentanalyzer.TIncident INNER JOIN
                incidentanalyzer.TIncidentType ON
incidentanalyzer.TIncident.IncidentTypeID =
incidentanalyzer.TIncidentType.ID LEFT OUTER JOIN
                incidentanalyzer.TUser ON
incidentanalyzer.TUser.ID = incidentanalyzer.TIncident.UserID LEFT
OUTER JOIN
                incidentanalyzer.TEscalation AS AAA ON
AAA.IncidentID = incidentanalyzer.TIncident.ID AND
AAA.NoticeTypeID = 2 INNER JOIN
                incidentanalyzer.TForm ON
incidentanalyzer.TForm.IncidentTypeID =
incidentanalyzer.TIncidentType.ID INNER JOIN
                incidentanalyzer.TFormElementRank ON
incidentanalyzer.TFormElementRank.FormID =
incidentanalyzer.TForm.ID LEFT OUTER JOIN
                incidentanalyzer.TFormElementValue AS CC0
ON incidentanalyzer.TIncident.ID = CC0.IncidentID AND
CC0.FormElementID IN
                (SELECT DISTINCT ID + 1 AS Expr1
FROM
incidentanalyzer.TFormElement
                WHERE          (ElementTypeID = 3)
AND (Value = 'Время возникновения')) LEFT OUTER JOIN
                incidentanalyzer.TFormElement AS DD0 ON
CC0.FormElementID = DD0.ID LEFT OUTER JOIN
                incidentanalyzer.TFormElementValue AS CC1
ON incidentanalyzer.TIncident.ID = CC1.IncidentID AND
CC1.FormElementID IN
                (SELECT DISTINCT ID + 1 AS Expr1
FROM
incidentanalyzer.TFormElement
                WHERE          (ElementTypeID = 3)
AND (Value = 'Время фиксации')) LEFT OUTER JOIN

```

```

                                incidentanalyzer.TFormElement AS DD1 ON
CC1.FormElementID = DD1.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC2
ON incidentanalyzer.TIncident.ID = CC2.IncidentID AND
CC2.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Источник информации')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD2 ON
CC2.FormElementID = DD2.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC3
ON incidentanalyzer.TIncident.ID = CC3.IncidentID AND
CC3.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Оценка')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD3 ON
CC3.FormElementID = DD3.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC4
ON incidentanalyzer.TIncident.ID = CC4.IncidentID AND
CC4.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Описание')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD4 ON
CC4.FormElementID = DD4.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC5
ON incidentanalyzer.TIncident.ID = CC5.IncidentID AND
CC5.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Субъекты')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD5 ON
CC5.FormElementID = DD5.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC6
ON incidentanalyzer.TIncident.ID = CC6.IncidentID AND
CC6.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Объекты')) LEFT OUTER JOIN

```

```

                                incidentanalyzer.TFormElement AS DD6 ON
CC6.FormElementID = DD6.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC7
ON incidentanalyzer.TIncident.ID = CC7.IncidentID AND
CC7.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE (ElementTypeID = 3)
AND (Value = 'Ид. КСИБ в ЕСМИБ')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD7 ON
CC7.FormElementID = DD7.ID
WHERE (incidentanalyzer.TIncidentType.ID = 243)
ORDER BY [Время записи в БД IF] DESC"></asp:SqlDataSource>
    <asp:Button ID="Button1" runat="server" Font-Names="HP
Simplified Light" Font-Overline="False"
PostBackUrl="~/GrafficPage.aspx" Text="Перейти к статистике"
Width="186px" />
    </form>
</body>
</html>

```

Листинг А.4 – Код страницы GrafficPage.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="GrafficPage.aspx.cs"
Inherits="Подсистема_Визуального_Анализа.GrafficPage" %>

<%@ Register assembly="DevExpress.XtraCharts.v18.1.Web,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a"
namespace="DevExpress.XtraCharts.Web" tagprefix="dx" %>
<%@ Register assembly="DevExpress.XtraCharts.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" namespace="DevExpress.XtraCharts"
tagprefix="dx" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-
8"/>
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <asp:Label ID="Label1" runat="server" Font-Names="HP
Simplified Light" Font-Size="X-Large" Text="Подсистема Визуального
Анализа КСИБ ЕСМИБ ТУ"></asp:Label>
        <dx:WebChartControl ID="WebChartControl1" runat="server"
CrosshairEnabled="True" DataSourceID="BD_IIB_ESMIB_TU"
Height="434px" Width="623px">
<Legend Name="Default Legend"></Legend>
            <SeriesSerializable>
                <dx:Series ArgumentDataMember="Источник
информации" Name="Ряд 1" ValueDataMembersSerializable="ID">
                    <viewserializable>
                        <dx:PieSeriesView>
                        </dx:PieSeriesView>
                    </viewserializable>
                </dx:Series>
            </SeriesSerializable>
        </dx:WebChartControl>
        <asp:SqlDataSource ID="BD_IIB_ESMIB_TU" runat="server"
ConnectionString="<%"$
ConnectionStrings:IncidentFactor2018ConnectionString %>"
SelectCommand="SELECT DISTINCT
                                incidentanalyzer.TIncident.ID,
incidentanalyzer.returnDefaultNameIncidentIfNull(incidentanalyzer.
TIncident.Name) AS [Название события/инцидента],
```

```

        CC0.Val AS [Время возникновения], CC1.Val
AS [Время фиксации], CC2.Val AS [Источник информации], CC3.Val AS
Оценка, CC4.Val AS Описание,
        CC5.Val AS Субъекты, CC6.Val AS Объекты,
CC7.Val AS [Ид. КСИБ в ЕСМИБ], incidentanalyzer.TIncidentType.Name
AS [Тип рег. формы],

incidentanalyzer.returnDateOrValueDefault(incidentanalyzer.TIncide
nt.BeginTime) AS [Время записи в БД IF],

incidentanalyzer.returnOpenedOrClosedOrEscalation(incidentanalyzer
.TIncident.Deleted, AAA.NoticeTypeID) AS Состояние,

incidentanalyzer.returnDateOrValueDefault(incidentanalyzer.TIncide
nt.EndTime) AS [Время закрытия],
        incidentanalyzer.TIncident.Result AS
[Вывод по обработке]
FROM          incidentanalyzer.TIncident INNER JOIN
                incidentanalyzer.TIncidentType ON
incidentanalyzer.TIncident.IncidentTypeID =
incidentanalyzer.TIncidentType.ID LEFT OUTER JOIN
                incidentanalyzer.TUser ON
incidentanalyzer.TUser.ID = incidentanalyzer.TIncident.UserID LEFT
OUTER JOIN
                incidentanalyzer.TEscalation AS AAA ON
AAA.IncidentID = incidentanalyzer.TIncident.ID AND
AAA.NoticeTypeID = 2 INNER JOIN
                incidentanalyzer.TForm ON
incidentanalyzer.TForm.IncidentTypeID =
incidentanalyzer.TIncidentType.ID INNER JOIN
                incidentanalyzer.TFormElementRank ON
incidentanalyzer.TFormElementRank.FormID =
incidentanalyzer.TForm.ID LEFT OUTER JOIN
                incidentanalyzer.TFormElementValue AS CC0
ON incidentanalyzer.TIncident.ID = CC0.IncidentID AND
CC0.FormElementID IN
                (SELECT DISTINCT ID + 1 AS Expr1
FROM
incidentanalyzer.TFormElement
                WHERE          (ElementTypeID = 3)
AND (Value = 'Время возникновения')) LEFT OUTER JOIN
                incidentanalyzer.TFormElement AS DD0 ON
CC0.FormElementID = DD0.ID LEFT OUTER JOIN
                incidentanalyzer.TFormElementValue AS CC1
ON incidentanalyzer.TIncident.ID = CC1.IncidentID AND
CC1.FormElementID IN
                (SELECT DISTINCT ID + 1 AS Expr1
FROM
incidentanalyzer.TFormElement
                WHERE          (ElementTypeID = 3)
AND (Value = 'Время фиксации')) LEFT OUTER JOIN

```

```

                                incidentanalyzer.TFormElement AS DD1 ON
CC1.FormElementID = DD1.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC2
ON incidentanalyzer.TIncident.ID = CC2.IncidentID AND
CC2.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Источник информации')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD2 ON
CC2.FormElementID = DD2.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC3
ON incidentanalyzer.TIncident.ID = CC3.IncidentID AND
CC3.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Оценка')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD3 ON
CC3.FormElementID = DD3.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC4
ON incidentanalyzer.TIncident.ID = CC4.IncidentID AND
CC4.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Описание')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD4 ON
CC4.FormElementID = DD4.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC5
ON incidentanalyzer.TIncident.ID = CC5.IncidentID AND
CC5.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Субъекты')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD5 ON
CC5.FormElementID = DD5.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC6
ON incidentanalyzer.TIncident.ID = CC6.IncidentID AND
CC6.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE
                                (ElementTypeID = 3)
AND (Value = 'Объекты')) LEFT OUTER JOIN

```

```

                                incidentanalyzer.TFormElement AS DD6 ON
CC6.FormElementID = DD6.ID LEFT OUTER JOIN
                                incidentanalyzer.TFormElementValue AS CC7
ON incidentanalyzer.TIncident.ID = CC7.IncidentID AND
CC7.FormElementID IN
                                (SELECT DISTINCT ID + 1 AS Expr1
                                FROM
incidentanalyzer.TFormElement
                                WHERE (ElementTypeID = 3)
AND (Value = 'Ид. КСИБ в ЕСМИБ')) LEFT OUTER JOIN
                                incidentanalyzer.TFormElement AS DD7 ON
CC7.FormElementID = DD7.ID
WHERE (incidentanalyzer.TIncidentType.ID = 243)
ORDER BY [Время записи в БД IF] DESC"></asp:SqlDataSource>
</form>
</body>
</html>

```

Листинг А.5 – Код Web.config

```
<configuration>
  <configSections>
    <!-- For more information on Entity Framework
configuration, visit http://go.microsoft.com/fwlink/?LinkID=237468
-->
    <section name="entityFramework"
type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework, Version=6.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089" requirePermission="false"/>
    <sectionGroup name="devExpress">
      <section name="themes"
type="DevExpress.Web.ThemesConfigurationSection,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" requirePermission="false"/>
      <section name="compression"
type="DevExpress.Web.CompressionConfigurationSection,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" requirePermission="false"/>
      <section name="settings"
type="DevExpress.Web.SettingsConfigurationSection,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" requirePermission="false"/>
      <section name="errors"
type="DevExpress.Web.ErrorsConfigurationSection,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" requirePermission="false"/>
      <section name="resources"
type="DevExpress.Web.ResourcesConfigurationSection,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" requirePermission="false"/>
    </sectionGroup>
  </configSections>
  <connectionStrings>
    <add name="DefaultConnection" connectionString="Data
Source=(LocalDb)\v11.0;AttachDbFilename=|DataDirectory|\aspnet-
Подсистема_Визуального_Анализа-20180606124325.mdf;Initial
Catalog=aspnet-Подсистема_Визуального_Анализа-
20180606124325;Integrated Security=True"
providerName="System.Data.SqlClient"/>
    <add name="IncidentFactor2018ConnectionString"
connectionString="Data Source=WIN-HGM47QE3M00\SQLEXPRESS;Initial
Catalog=IncidentFactor2018;Integrated Security=True"
providerName="System.Data.SqlClient"/>
  </connectionStrings>
  <system.web>
    <authentication mode="None"/>
    <compilation debug="true" targetFramework="4.5">
      <assemblies>
```

```

        <add
assembly="DevExpress.Web.ASPxPivotGrid.v18.1, Version=18.1.3.0,
Culture=neutral, PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.PivotGrid.v18.1.Core,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.Data.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.Web.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.Printing.v18.1.Core,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="System.Windows.Forms,
Version=4.0.0.0, Culture=neutral,
PublicKeyToken=B77A5C561934E089"/>
        <add assembly="System.Design, Version=4.0.0.0,
Culture=neutral, PublicKeyToken=B03F5F7F11D50A3A"/>
        <add assembly="DevExpress.Web.ASPxThemes.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a"/>
        <add assembly="DevExpress.XtraCharts.v18.1.Web,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.XtraCharts.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.Charts.v18.1.Core,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/>
        <add assembly="DevExpress.Web.Resources.v18.1,
Version=18.1.3.0, Culture=neutral,
PublicKeyToken=B88D1754D700E49A"/></assemblies>
    </compilation>
    <httpRuntime targetFramework="4.5"/>
    <pages>
        <namespaces>
            <add namespace="System.Web.Optimization"/>
            <add namespace="Microsoft.AspNet.Identity"/>
        </namespaces>
        <controls>
            <add
assembly="Microsoft.AspNet.Web.Optimization.WebForms"
namespace="Microsoft.AspNet.Web.Optimization.WebForms"
tagPrefix="webopt"/>
        </controls>
    </pages>
    <membership>
        <providers>

```

```

<!--
    В этом шаблоне отключено членство в ASP.NET. Чтобы
    получить дополнительные сведения о поддержке членства в ASP.NET в
    данном шаблоне, перейдите по следующей ссылке:
    http://go.microsoft.com/fwlink/?LinkId=301889
-->

```

```

        <clear/>
    </providers>
</membership>
<profile>
    <providers>
        <!--

```

В этом шаблоне отключен профиль членства в ASP.NET. Чтобы получить дополнительные сведения о поддержке членства в ASP.NET в данном шаблоне, перейдите по следующей ссылке:
<http://go.microsoft.com/fwlink/?LinkId=301889>

```

-->
        <clear/>
    </providers>
</profile>
<roleManager>
    <!--

```

В этом шаблоне отключена роль членства в ASP.NET. Чтобы получить дополнительные сведения о поддержке членства в ASP.NET в данном шаблоне, перейдите по следующей ссылке:
<http://go.microsoft.com/fwlink/?LinkId=301889>

```

-->
    <providers>
        <clear/>
    </providers>
</roleManager>
<!--

```

Если выполняется развертывание в облачной среде с несколькими экземплярами веб-сервера, следует изменить состояние сеанса с "InProc" на "Custom". Также

измените строку подключения "DefaultConnection", чтобы выполнять подключение к экземпляру сервера SQL Server (в том числе SQL Azure и SQL Compact), а не SQL Server Express.

```

-->
    <sessionState mode="InProc"
customProvider="DefaultSessionProvider">
        <providers>
            <add name="DefaultSessionProvider"
type="System.Web.Providers.DefaultSessionStateProvider,
System.Web.Providers, Version=2.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35"
connectionStringName="DefaultConnection"/>
        </providers>
    </sessionState>

```

```

        <httpModules>
            <add type="DevExpress.Web.ASPxHttpHandlerModule,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" name="ASPxHttpHandlerModule"/>
        </httpModules>
        <httpHandlers>
            <add type="DevExpress.Web.ASPxHttpHandlerModule,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" verb="GET" path="DX.ashx"
validate="false"/>
        </httpHandlers>
    </system.web>
    <system.webServer>
        <modules>
            <remove name="FormsAuthentication"/>
            <add type="DevExpress.Web.ASPxHttpHandlerModule,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" name="ASPxHttpHandlerModule"/>
        </modules>
        <validation validateIntegratedModeConfiguration="false"/>
        <handlers>
            <add type="DevExpress.Web.ASPxHttpHandlerModule,
DevExpress.Web.v18.1, Version=18.1.3.0, Culture=neutral,
PublicKeyToken=b88d1754d700e49a" verb="GET" path="DX.ashx"
name="ASPxHttpHandlerModule" preCondition="integratedMode"/>
        </handlers>
    </system.webServer>
    <runtime>
        <assemblyBinding xmlns="urn:schemas-microsoft-
com:asm.v1">
            <dependentAssembly>
                <assemblyIdentity name="Newtonsoft.Json"
culture="neutral" publicKeyToken="30ad4fe6b2a6aeed"/>
                <bindingRedirect oldVersion="0.0.0.0-6.0.0.0"
newVersion="6.0.0.0"/>
            </dependentAssembly>
            <dependentAssembly>
                <assemblyIdentity name="WebGrease"
culture="neutral" publicKeyToken="31bf3856ad364e35"/>
                <bindingRedirect oldVersion="0.0.0.0-
1.5.2.14234" newVersion="1.5.2.14234"/>
            </dependentAssembly>
            <dependentAssembly>
                <assemblyIdentity name="EntityFramework"
publicKeyToken="b77a5c561934e089"/>
                <bindingRedirect oldVersion="0.0.0.0-6.0.0.0"
newVersion="6.0.0.0"/>
            </dependentAssembly>
            <dependentAssembly>
                <assemblyIdentity name="Microsoft.Owin"
culture="neutral" publicKeyToken="31bf3856ad364e35"/>

```

```

        <bindingRedirect oldVersion="0.0.0.0-3.0.1.0"
newVersion="3.0.1.0"/>
    </dependentAssembly>
    <dependentAssembly>
        <assemblyIdentity
name="Microsoft.Owin.Security.OAuth" culture="neutral"
publicKeyToken="31bf3856ad364e35"/>
        <bindingRedirect oldVersion="0.0.0.0-3.0.1.0"
newVersion="3.0.1.0"/>
    </dependentAssembly>
    <dependentAssembly>
        <assemblyIdentity
name="Microsoft.Owin.Security.Cookies" culture="neutral"
publicKeyToken="31bf3856ad364e35"/>
        <bindingRedirect oldVersion="0.0.0.0-3.0.1.0"
newVersion="3.0.1.0"/>
    </dependentAssembly>
    <dependentAssembly>
        <assemblyIdentity
name="Microsoft.Owin.Security" culture="neutral"
publicKeyToken="31bf3856ad364e35"/>
        <bindingRedirect oldVersion="0.0.0.0-3.0.1.0"
newVersion="3.0.1.0"/>
    </dependentAssembly>
</assemblyBinding>
</runtime>
<entityFramework>
    <defaultConnectionFactory
type="System.Data.Entity.Infrastructure.SqlConnectionFactory,
EntityFramework"/>
    <providers>
        <provider invariantName="System.Data.SqlClient"
type="System.Data.Entity.SqlServer.SqlProviderServices,
EntityFramework.SqlServer"/>
    </providers>
</entityFramework>
<devExpress>
    <themes enableThemesAssembly="true" styleSheetTheme=""
theme="" customThemeAssemblies="" baseColor="" font=""/>
    <compression enableHtmlCompression="false"
enableCallbackCompression="true" enableResourceCompression="true"
enableResourceMerging="true"/>
    <settings accessibilityCompliant="false" bootstrapMode=""
doctypeMode="Html5" rightToLeft="false"
checkReferencesToExternalScripts="true" protectControlState="true"
ieCompatibilityVersion="edge"/>
    <errors callbackErrorRedirectUrl=""/>
</devExpress>
</configuration>

```

Приложение Б
(обязательное)

Алгоритм работы подсистемы визуального анализа данных об инцидентах
информационной безопасности

**Алгоритм работы подсистемы визуального анализа данных об инцидентах
информационной безопасности**



Рисунок Б.1 – Алгоритм работы программного обеспечения подсистемы
визуального анализа