Система сбора и анализа информации из различных источников в условиях Big Data

Д. В. Смирнов, А. А. Грушо, М. И. Забежайло, Е. Е. Тимонина

***Аннотация*—Исследована задача построения архитектуры и методов поиска признаков инсайдерской деятельности в условиях процессно-реального времени. Задача решена в следующих условиях. Источником «сырых» данных является Big Data, из которых выбираются данные, релевантные признакам инсайдерской активности, в соответствии с текущим перечнем угроз. Поиск ведется по большому множеству пользователей. В этих условиях построен алгоритм, преодолевающий барьер сложности в поиске релевантных данных.**

Важным осложнением задачи являются условия «открытости» данных. Условие «открытости» данных предполагает постоянное обновление данных. В понятие «открытости» также входит изменение признаков враждебной деятельности инсайдеров. При этом могут динамически изменяться и условия поиска.

Построенная архитектура является двухуровневой. Первый уровень содержит данные, собранные из различных баз «сырых» данных, и релевантные текущему перечню угроз. Второй уровень связан с максимальной доступностью организованных на первом уровне данных для проведения анализа с участием экспертов – оперативных работников. Приведено научное обоснование корректности и эффективности задействованных при реализации этого программного комплекса математических моделей и алгоритмов интеллектуального анализа больших данных.

Построенные решения показали свою работоспособность в промышленном варианте решения задачи.

***Ключевые слова*—Информационная безопасность, поиск признаков враждебного инсайдера в Big Data, интеллектуальный анализ данных.**

I. Введение

Цель представляемой работы – разработка архитектуры системы выявления признаков инсайдерской активности в условиях Big Data (BD). Разработка такой системы необходима для поддержки профильной деятельности оперативных работников служб безопасности.

Статья получена 29 марта 2021.

Работа поддержана РФФИ (проект № 18-29-03081-мк).

Д. В. Смирнов, ПАО Сбербанк России, Москва, Россия (e-mail: dvlsmirnov@sberbank.ru).

А. А. Грушо, Федеральный исследовательский центр «Информатика и управление» Российской академии наук, Москва, Россия (e-mail: (e-mail: grusho@yandex.ru).

М. И. Забежайло, Федеральный исследовательский центр «Информатика и управление» Российской академии наук, Москва, Россия (e-mail: m.zabezhailo@yandex.ru).

Е. Е. Тимонина, Федеральный исследовательский центр «Информатика и управление» Российской академии наук, Москва, Россия (e-mail: eltimon@yandex.ru).

Актуальность обсуждаемой тематики обусловлена тем, что на рынке (причем – не только отечественном) отсутствуют готовые коммерческие продукты, обеспечивающие выявление признаков внутренних нарушителей в условиях BD.

Новизна подхода, предлагаемого для решения обсуждаемой проблемы, определяется в первую очередь созданной оригинальной архитектурой системы, с помощью которой можно обрабатывать BD в близком к реальному времени. Работоспособность и практическая значимость методики, а также обеспечивающих ее применение программных инструментов анализа данных и поддержки принятия решений подтверждены внедрением и использованием разработанного инструментария в текущую деятельности крупного отечественного коммерческого банка.

Обычно поисковые сервисы предоставляются их пользователям в рамках трех «архитектурных» моделей:

- облачные услуги (SaaS);
- готовое ПО, включающее в себя библиотеки, пользовательские интерфейсы и т.д.;
- библиотеки «базовых» программных инструментов, реализующие основные функции поисковых систем.

Сервисы доступны как в виде соответствующих корпоративных, так и web-поисковых систем. При этом корпоративные поисковые системы отличаются от web-поисковиков, в первую очередь, тем, что имеют собственные:

- адаптеры для подключения источников различного формата (базы данных, CMS системы, файловые хранилища и т.д.);
- функции контроля доступа;
- пользовательские интерфейсы.

Основными функциями корпоративных поисковых систем являются:

- собственно поиск и навигация с учетом контекста и типа запроса, возможные опечатки, синонимы и словоформы, ввод запроса на разных языках и многое другое;
- голосовой поисковый интерфейс (преобразование в режиме реального времени голосового запроса в текстовый, и далее поиск и выдача лучших результатов);
- геопоиск, позволяющий настроить стратегию ранжирования результатов поиска на основе гео-данных, например, ограничив результаты поиска улицей, городом, континентом и т.п.;
- мобильный поиск (возможности визуализации поисковых ответов для конечных пользовательских устройств iOS, Android и мобильного web-доступа).

Среди наиболее известных библиотек, реализующих поисковые технологии, можно отметить следующие:

- Apache Lucene (ElasticSearch, Solr, MongoDB Atlas Search, Datafari, CrateDB) [1];
- Apache Lucy [2];
- FTS, Tsearch2, RUM, GIN, OpenFTS, GIST (Postgres) [3];
- Sphinx/Manticore [4];
- Indri (Lemur) [5];
- Fulltext (MySQL) [6];
- Terrier [7];
- Manatee [8];
- iSearch Library (ArangoSearch) [9];
- Lunar [10];
- Xapian [11].

Широко распространена ситуация, когда производитель поисковой библиотеки может одновременно выпускать также и поисковое ПО.

Инструменты поддержки поиска позволяют отслеживать и анализировать действия и намерения пользователя (которые, например, можно сопоставлять с различными сведениями управленческого характера: текущими целями и задачами его производственной деятельности; должностными полномочиями по доступу к тем или иным информационным ресурсам; и т.п.).

II. ИТ-среда анализа данных и поддержки принятия решений

Задача поиска ориентирована на комплекс накапливания и обработки BD, охватывающий, примерно, 2000 серверов. Данный комплекс BD в силу специфики решаемых на его базе бизнес-задач является динамической структурой, конфигурация которой варьируется (в части увеличения или же, наоборот, уменьшения) примерно на 10 серверов ежесуточно. В этом комплексе одновременно работают несколько тысяч специализированных сотрудников-аналитиков данных, в задачи которых входит обеспечение полного цикла анализа данных и принятия соответствующих решений: подготовка данных, разработка моделей и т.д. Для их работы требуется порядка десятка петабайт так называемых полезных данных при общем объеме накапливаемой и обрабатываемой информации в несколько раз больше. Полезные данные представлены в виде более сотни баз данных и нескольких сотен аналитических продуктов (витрин данных).

Разработанные методика и программный инструментарий выявления признаков инсайдера - это комплекс меньшего размера: примерно 6% от общего парка серверов BD. Данный инструментальный комплекс защиты порождает несколько десятков терабайт так называемых «сырых» данных в сутки, которые необходимо обрабатывать, т.е. фильтровать, приводить к нормализованному виду, индексировать, и т.п., чтобы постоянно обеспечивать требуемые ограничения процессно-реального времени для обновления\актуализации параметров системы мониторинга.

Основные этапы жизненного цикла анализа данных в рассматриваемой системе:

- *построение и актуализация профиля угроз;*
- *поиск релевантных данных в «сырых» источниках;*
- *нормализация данных;*
- *представление знаний (разметка, создание поисковых индексов);*
- *интеллектуальный анализ данных;*
- *взаимодействие с экспертом (оперативным работником).*

Представляемый комплекс средств выявления признаков инсайдера имеет следующую архитектуру. Базовые компоненты Комплекса – это единый слой хранения отобранных из различных источников релевантных данных: BD, и аналитическое ядро: программно-технические инструменты анализа данных и поддержки принятия решений. Используются типовые виды программных инструментов:

- *базы данных,*
- *серверы приложений,*
- *среды исполнения программного кода (Python, Java).*

Созданный инструментарий имеет два различных типа интерфейсов, ориентированных на задачи формализованного представления знаний для так называемых первичного и вторичного поиска.

Первичный поиск характеризуется классом задач, где аналитик:

- *выявляет* в исходных «сырых» данных те данные, которые релевантны отдельно накапливаемым знаниям из текущего профиля угроз, аккумулирующего в себе уже накопленный опыт экспертов служб безопасности об особенностях наблюдавшихся ранее инсайдерских активностей и др.;
- *определяет* необходимые характеристики в «сырых» данных: поля, идентификаторы и т.п.;
- с использованием специальных средств машинного обучения, опираясь на прецеденты ранее идентифицированных инсайдерских угроз, *выделяет* из текущих «сырых» данных всю ту информацию, которая далее будет использована для поддержки текущей работы служб безопасности:
  - мониторинга основного комплекса BD на предмет идентификации в его текущей операционной работе тех или иных аномальных активностей,
  - организации противодействия противоправным действиям инсайдерского характера,
  - обеспечения оперативной отчетности, например, по запросам руководства в ситуациях, когда требуется санкция на те или иные специальные действия.

Вторичный поиск обеспечивает оперативное отображение и ответы на запросы, релевантные целям мониторинга безопасности (это своего рода «локальный Яндекс»), в которых оперативный сотрудник может:

- ввести необходимые идентификационные данные, такие как ФИО, табельный номер или источник данных и др.,
- посмотреть детальный профиль соответствующего сотрудника или подразделения, например, штатный профиль доступов данного сотрудника к ресурсам защищаемого комплекса BD в соотнесении с

текущими характеристиками, полученными в результате работы алгоритмов машинного обучения.

III. Проблемы, потребовавшие решения при разработке системы

При разработке архитектуры и комплекса реализующих ее программно-технических инструментов был идентифицирован ряд типичных для работы с BD барьеров, для преодоления которых пришлось разрабатывать проблемно-ориентированные результативные решения. Такие барьеры и связанные с ними задачи можно объединить в следующие четыре однородные группы.

A. Процессно-реальное время, эффекты Big и Open

Требующие анализа BD необходимо обрабатывать в режиме так называемого процессно-реального времени, обеспечивая «упаковку» всех необходимых стадий обработки в жесткие ограничения по времени. При этом следовало учитывать постоянные динамические изменения в объекте мониторинга, т.е. регулярное поступление новых данных. При организации мониторинга текущего состояния объекта необходимо учитывать, как собственно эффект Big, так и эффект Open – «открытость» (возможности пополнения новой информацией) исследуемого комплекса BD.

B. Интеграция данных, извлекаемых из различных источников

Интеграция данных, отбираемых из различных источников «сырых» первичных данных, представляет собою нетривиальную задачу. Необходимо в режиме процессно-реального времени отбирать релевантную целям мониторинга информацию из огромного перечня объектов (ресурсов), характеризуемых своими собственными типами представления данных: именами полей и доменов, именами и значениями атрибутов и т.п. Для преодоления таких барьеров был предложен и реализован в виде результативных программных инструментов ряд проблемно-ориентированных эвристик, отражающих зарекомендовавшую себя на практике «логику» оперирования с разнородными данными: «склеивание» согласуемых данных, используемых профильными экспертами службы безопасности при поиске инсайдерских активностей.

C. Нормализация обрабатываемых данных и, как следствие, сокращение объемов перечней объектов-примитивов за счет элиминации объектов-дубликатов

Так, например, пользователи сервисов вторичного поиска при работе со средствами диалогового интерфейса допускают различного рода неточности и\или ошибки в именовании искомого объекта. Именно это обстоятельство потребовало разработки соответствующих средств автоматической идентификации и коррекции ошибок: клавиатурных ошибок, опечаток, «ослышек» и т.п.

D. Ресурсные ограничения

Операциональные характеристики разрабатываемого комплекса должны быть оптимизированы. Стоимость инструментального комплекса защиты от инсайдерских активностей не должна превышать 10% процентов от стоимости собственно объекта защиты.

В порядке иллюстрации приведем несколько примеров реальных технических проблем, решение которых пришлось разработать при создании обсуждаемой системы защиты от инсайдерских активностей.

1) При обработке исходных «сырых» данных на первом этапе их фильтрации несколько десятков терабайт «стартовых» характеристик анализируемых событий удалось «ужать» до 600 Гб (1,5 млрд. записей об анализируемых активностях).
2) На втором этапе фильтрации данных эти 600 Гб «ужали» до 2 гигабайт (3 млн. записей об активностях).
3) При этом удалось добиться того, что обеспечивающие вторичный поиск индексы обновлялись в режиме имеющихся ограничений процессно-реального времени, а время отклика на запрос не превышало 10 сек. на выделенном для этого программно-техническом комплексе.

Одной из критически значимых целей такой фильтрации данных было сокращение количества актуальных для поисковой обработки записей. Так, например, известно, что используемое в целом ряде задач поиска промышленное ПО Elastic Search перестает в штатном режиме отвечать на запросы при размерах индекса более 50 млн. записей.

IV. Методика анализа данных и поддержки принятия решений

Разработка методики идентификации признаков инсайдерской активности начинается с формирования актуальной модели угроз. Модель угроз формализуется в виде профиля угроз (ПУ), представляющего собой постоянно поддерживаемый в актуальном состоянии перечень так называемых типовых сценариев (ТС). Каждый из ТС порождается обобщением опыта оперативных сотрудников, вовлеченных в расследования конкретных случаев мошенничества или инсайдерских активностей. Опыт оперативных сотрудников сперва фиксируется в виде текстового описания, которое далее преобразуется в машиночитаемый формализованный вид. При этом задействовано промежуточное представление знаний о каждом из ТС в виде фрейма. Для описания данных в слотах подобных фреймов предусмотрены иерархии типов данных: от булевых значений признаков (Да\Не до графов параметров и отношений между такими параметрами с пометками на вершинах и ребрах, а также текстовых комментариев, например, в виде Binary Large Objects – BLOB.

Подобные иерархии типов данных могут быть задействованы в случае необходимости получения более тонкой «дифференциации» состояний НОРМА\АНОМАЛИЯ с помощью использования более детального представления знаний об анализируемых

инцидентах. Простейший вариант представления знаний в ТС ПУ – использование булевых значений Да\Нет, позволяющих описать каждый такой фрейм в виде множества признаков, характеризующих именно его. В свою очередь множество всех используемых при описании текущего ПУ признаков определяет множество битовых переменных соответствующими единицами которых кодируется каждый из соответствующих ТС в ПУ. Обработка машиночитаемого описания фреймов, представленных в виде именно битовых векторов, дает возможность получить существенный выигрыш в производительности при анализе текущих данных, т.к. позволяет организовать сравнение текущей ситуации с описаниями ТС средствами одной вычислительной макрооперации.

Рис. 1. Формализация описания ПУ

Текущий, актуальный на данный момент ПУ – динамически изменяемая во времени, пополняемая с учетом постоянно накапливаемого опыта оперативных действий конструкция. В таком ПУ могут находится десятки или сотни ТС. Ниже представлены некоторые примеры текстовых версий ТС.

- Сотрудник X выполнил «точечный» запрос к базе Y, в которой 100 млн. записей.
- Сотрудник Z, работающий в одном подразделении X, имеет 80% доступ к данным другого подразделения Y.
- Сотрудник X, работающий с данными, не посещает офис более 1 дня в неделю.
- Сотрудник X, имеющей те же доступы, что и его коллеги из офиса Y, физически размещается в другом офисе Z.
- Сотрудник X, имеющей одновременно доступ в аналитическую систему Y и транзакционную систему Z.

В формализованном описании текущий ПУ может быть описан (см. Рис. 1) как матрица, строки которой при использовании булевого варианта представления знаний от ТС в соответствующих фреймах соответствуют задействованным при описании угроз параметрам/признакам, а каждый из столбцов этой матрицы представляет соответствующий ТС.

Сравнивая элементы (ячейки) этой матрицы с характеристиками (профилем значений признаков) доступа к защищаемым ресурсам комплекса BD, актуальными в данный момент для конкретного мониторимого сотрудника, можно оценить весомость угроз несанкционированных активностей этого сотрудника, т.е. релевантность его текущего поведения каким-либо известным угрозам из текущего ПУ. Однако, проведение таких сравнений методом «грубой силы» оказывается чрезвычайно ресурсоемким (см. Раздел 1). Таким образом, востребованными оказываются любые результативные приемы, подходы и методы сокращения объемов перебора при формировании «диагностических» заключений по каждому из мониторимых сотрудников.

Наряду с уже представленными выше инструментами нормализации и фильтрации исходных «сырых» данных существенный выигрыш в объемах необходимых вычислений позволяет получить процедурное уточнение идеи (эвристики) учета сходств в описаниях ТС. Действительно, при оценке релевантности текущего профиля доступов конкретного сотрудника к защищаемым информационным ресурсам представляется вполне естественным начать такие проверки с наиболее общих для всех актуальных ТС множеств признаков, переходя далее ко все менее и менее общим, завершая весь процесс сравнением с собственно каждым из имеющихся ТС.

Говоря формально (см. [12]), определив бинарную алгебраическую операцию сходства описаний ТС [12-14] можно построить диаграмму взаимной вложимости множеств признаков, задействованных в описаниях ТС. Далее, один раз сформировав такую диаграмму, проверять релевантность текущего анализируемого профиля доступов конкретного сотрудника имеющемуся ПУ, начиная со сравнения его элементов с элементами нижнего «этажа», т.е. максимальных по вложению подмножеств признаков, одновременно актуальных для нескольких ТС, и далее двигаясь лишь по релевантным цепочкам частичного порядка этой диаграммы к ее верхнему «этажу» (подмножеств минимальных по числу актуальных общих признаков), а от него – к релевантным данной ситуации описаниям ТС (см. Рис. 2).

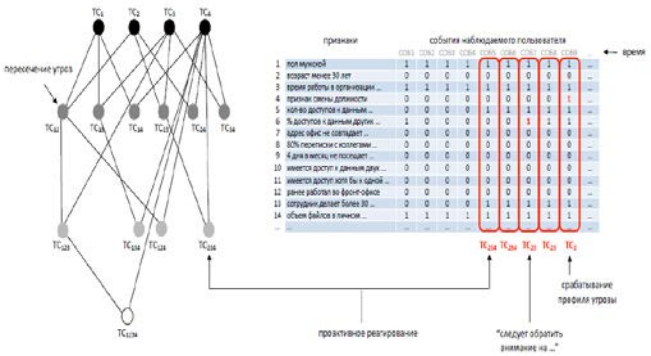

Рис. 2. Диаграмма сходств ТС из ПУ

Подобная тактика первоочередного использования наиболее общих для имеющихся описаний ТС множеств признаков и последующего движения лишь вдоль актуальных цепочек частичного порядка в один раз построенной диаграмме позволяет не только

существенным образом сократить необходимые объемы вычислений при проверке релевантности текущей профиля доступов конкретного сотрудника и актуального ПУ, но и в проактивном режиме подсказать офицеру безопасности в текущем конкретном случае наиболее опасные варианты дальнейшего развития событий, «подсвечивая» соответствующие цепочки частичного порядка на диаграмме сходств описаний ТС, двигаясь с ее нижнего «этажа» вверх к релевантным этому конкретному профилю доступов описаниям ТС.

В случае булевого представления данных об имеющихся ТС каждый такой ТС характеризуется битовым вектором. Таким образом получаем возможность вычислять сходства описаний ТС, используя стандартные для многих современных системных программных сред макро-операции с битовыми векторами. Это позволяет работать с имеющимися BD достаточно быстро и эффективно, формируя результат сходства описаний ТС средствами соответствующей машинной макро-операции.

Идею оценки релевантности текущего профиля доступов конкретного сотрудника ТС актуального ПУ иллюстрирует алгоритм Рис. 3. Для выполнения такой оценки достаточно выявления общих частей описаний объекта мониторинга и ТС, в том числе с учетом ранее рассчитанных риск-индикаторов идентификации аномалий на наличие признаков инсайдерской активности (см. [15-19]).

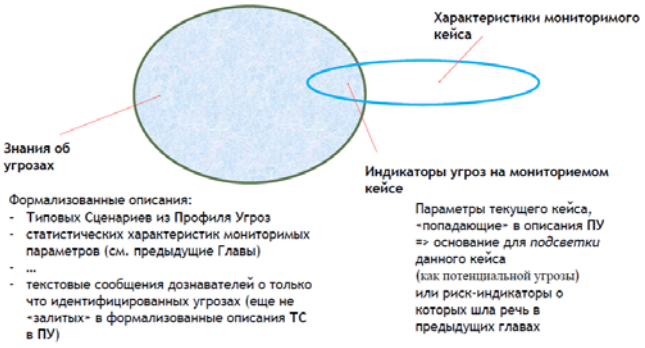

Рис. 3. Отношение релевантности текущая ситуация ~ ПУ

Особого внимания требует учет того обстоятельства, что ПУ – это динамически изменяемая конструкция, которая может быть в любое время модифицирована аналитиками, обрабатывающими накапливаемый опыт идентификации и противодействия вредоносным активностям. При этом следует учитывать, что управление перебором, т.е. уход от сравнения «всего» в описании анализируемого профиля доступов со «всем» в ПУ при обсуждаемой оценке их релевантности, в рассматриваемом контексте BD оказывается неприемлемо ресурсоемкой тактикой анализа данных и поддержки принятия решений. При этом необходимо провести исчерпывающий анализ совпадений фрагментов текущего профиля доступов каждого конкретного сотрудника по всем ТС актуального Профиля Угроз, что требует обработки данных о нескольких миллионах пользователей в сутки.

Итак, в предлагаемом подходе при использовании сходств ТС из актуального ПУ при оценке опасности действий конкретного сотрудника при его доступе к защищаемым от инсайдерских активностей информационным ресурсам можно существенным образом оптимизировать по сравнению с тактикой «грубой силы», предусматривающей сравнения «всего» со «всем» объемы необходимых вычислений. Для этого следует один раз сформировать диаграмму сходств и последовательно вести проверки ее пересечений ее фрагментов с теми «релевантными» анализируемому профилю доступов конкретного сотрудника элементами диаграммы сходств ТС, которые размещены на ее цепочках частичного порядка. В ситуации, когда речь идет о миллиардах событий и о сотнях ТС, таким способом можно сформировать значительный выигрыш в скорости принятия финальных решений. Необходимость подобной оптимизации мотивируется достаточно естественным образом: угроз защищаемому комплексу BD со временем становится все больше, и это требует эффективного управления имеющимися вычислительными ресурсами.

Дополнительный аргумент в пользу предлагаемого подхода – возможность организовать проактивный мониторинг негативного развития «аномальных» ситуаций, подсказывая конкретному сотруднику безопасности наиболее опасные варианты изменения отслеживаемой им конкретной ситуации вдоль релевантных ей цепочек частичного порядка на диаграмме сходств ТС.

V. Программный инструментарий реализации предложенной методики

Как уже отмечалось выше, ПУ – это динамически изменяемая конструкция, предполагающая возможность модификации в соответствии со вновь накапливаемыми эмпирическими данными о поведении объектов мониторинга, а также опытом противодействия, как успешного, так и нерезультативного, идентифицированным вредоносным активностям. Поддержка изменений в «архитектуре» актуального ПУ потребовала разработки соответствующих программных инструментов экономного реинжиниринга структуры диаграммы сходств описаний ТС. Показано [13-14], что при порождении диаграммы сходств ТС в общем случае приходится иметь дело с объектом, размеры которого растут экспоненциально быстро при линейном росте размеров множества ТС. Таким образом, актуальной оказалась задача оптимизации перебора вариантов (локальных сходств описаний ТС) при формировании диаграммы сходств ТС. Для этого был разработан специальный программный инструмент со встроенными алгоритмом экономной организации генерации локальных сходств описаний ТС. Следуя подходу Rapid Application Development, сперва в инструментальной среде Matlab был проведена отладка и проверка корректности этого алгоритма анализа данных и принятия решений, а далее на Python была реализована его промышленная версия, использующая возможности экономной обработки битовых векторов.

Вместе со специально разработанным проблемно-ориентированным графическим редактором, обеспечивающим аналитикам возможности формировать новые ТС и поддерживать ПУ в актуальном состоянии, эта промышленная Python-версия генератора диаграммы сходств описаний ТС образует ядро программного инструментария представления и обработки знаний о признаках вредоносных инсайдерских активностях.

Архитектура разработанной системы мониторинга и анализа признаков инсайдерских угроз предусматривает реализацию двух типов поиска:

- *первичного*, включая выделение релевантной информации из первичных «сырых» данных,
- и *вторичного* – быстрый поиск в уже отобранных релевантных данных для подготовки управленческой отчетности, а также для информационного сопровождения оперативной деятельности сотрудников службы безопасности.

Для поддержки этих двух классов информационных сервисов разработаны соответствующие пользовательские интерфейсы. Так в первичном поиске интерфейс помогает аналитику отбирать все те сведения, что должны быть «подсвечены» в последующей работе, как релевантные знания об угрозах. Исходные данные для машинного обучения, как описания прецедентов, формируемые на базе анализа инцидентов безопасности, проанализированных экспертами, вводятся в систему первичного поиска через соответствующие интерфейсы. Сюда «удобным» образом подключены нормализованные данные из единого слоя хранения и имеются алгоритмы, ориентированные на оптимизацию перебора вариантов, которая необходима для соблюдения ограничений режима процессно-реального времени анализа данных и поддержки принятия соответствующих управленческих решений.

Вторичный поиск обеспечивается отдельной поисковой системой, пользовательский интерфейс которой поддерживает работу с текстовым полем для ввода запросов и кнопкой «искать». Цель вторичного поиска – оперативно предоставлять информацию, включающую результаты работы алгоритмов машинного обучения, в простом и понятном виде для сотрудников, не имеющих продвинутых ИТ-навыков. Важнейший эффект, обеспечиваемый средствами вторичного поиска, – это ускорение работы оперативных сотрудников, занятых мониторингом признаков действий инсайдеров и противодействием вредоносным инсайдерским активностям.

Особого внимания заслуживают возможности специально разработанных лингвистических программных сервисов, которые в автоматическом режиме поддерживают процесс нормализации анализируемых данных. В различных источниках поля и значения таких полей данных, как правило, называются различным образом. В частности, одно и то же наименование места или города в различных базах данных может иметь разные называния, например, город Москва может иметь десятки различных написаний, в т. ч. «MOSCOW», «G. MOSKVA», «MOSKVA», «ГОРОД МОСКВА», «МОСКВА, МОСКОВСКАЯ ОБЛАСТЬ», и т.д. Именно по этой причине анализируемые данные необходимо нормализовать. При этом проводимая нормализация преследует три базовые цели:

1. уменьшить объемы данных, используемых в реальном мониторинге,
2. объединять данные из разных источников,
3. представлять данные пользователю в унифицированной форме.

В инфраструктуру вторичного поиска встроены специально разработанные программные инструменты, реализующие алгоритмы корректировки опечаток и «ослышек». Исправление «ослышек» необходимо, например, в ситуациях, когда оперативный работник узнал фамилию из сообщения в телефоне и не знает, как именно пишется эта фамилия. Он вбивает в строку поиска то, что услышал, и алгоритм корректирует результаты ввода. Для решения таких задач коррекции был разработан собственный фонетический алгоритм, реализующий два этапа: фонетическое редуцирование и механизм (правила) так называемого оглушения. Для устранения опечаток ввода запроса были использованы обыкновенные триграммы – алгоритм измерения дистанции между эталонным названием и опечаткой. Тестирование разработанных программных инструментов эмпирическим путем подтвердило корректность работы алгоритмов на базе в несколько сотен тысяч сотрудников.

Представляемые методика и программно-технический комплекс идентификации признаков вредоносных инсайдерских активностей разработаны для использования в среде BD, одной из критически значимых характеристик, которой является так называемый эффект Open, т.е. открытый (незамкнутый), допускающий пополнение новыми сведениями массив анализируемых данных. Очевидно, что предлагаемые средства анализа данных и поддержки принятия решений должны быть способны учесть, что:

- анализируемые данные постоянно расширяются,
- количество пользователей защищаемой системы постоянно увеличивается,
- актуальный профиль угроз регулярно редактируется и изменяется.

В этой связи были разработаны соответствующие регламенты, а также программные инструменты поддержки изменений и развития обсуждаемой системы выявления признаков вредоносных инсайдерских активностей:

- средства для поддержки реорганизации (расширения и модификации) ПУ с учетом динамически накапливаемого опыта. Инструментальные средства поддержки таких реорганизаций: методики\регламенты, программные инструменты анализа данных и визуализации результатов;
- средства для поддержки реорганизации (расширения и модификации) поискового аппарата (поисковых индексов, классификационных систем и

т.п.) для поддержания эффективности вторичного поиска в динамически изменяемой информационной среде.

VI. Заключение

Основные результаты представленных исследований и разработок можно суммировать следующим образом.

1) Создана оригинальная технология поиска признаков инсайдерской активности, методика анализа данных, обеспечивающая противодействие вредоносным инсайдерским активностям в условиях работы с BD.
2) Для практической реализации этой методики разработан комплекс программных инструментов, который во взаимодействии с рядом уже эксплуатируемых индустриальных программных продуктов продемонстрировал свою работоспособность и результативность в крупном отечественном коммерческом банке.
3) Приведено научное обоснование корректности и эффективности задействованных при реализации этого программного комплекса математических моделей и алгоритмов интеллектуального анализа больших данных.

Библиография

[1] Welcome to Apache Lucene. Available: https://lucene.apache.org.
[2] The Apache Software Foundation. Available: https://lucy.apache.org.
[3] O. Bartunov, “Do you need a Full-Text Search in PostgreSQL?”, in PGConf.eu, Oct 26, Lisbon, 84 p., 2018. Available: https://www.postgresql.eu/events/pgconfeu2018/sessions/session/2116/slides/137/pgconf.eu-2018-fts.pdf.
[4] Open-source database for search applications. Available: https://manticoresearch.com.
[5] INDRI: Language modeling meets inference networks. Available: https://www.lemurproject.org/indri/.
[6] MySQL: Full-Text Search Functions. Available: https://dev.mysql.com/doc/refman/8.0/en/fulltext-search.html.
[7] Welcome to the Terrier IR Platform. Available: http://terrier.org.
[8] NLP-Center: NoSketch Engine. Available: https://nlp.fi.muni.cz/trac/noske.
[9] ArangoDB: Powerful Search Included. Available: https://www.arangodb.com/full-text-search-engine/.
[10] LUNR: Search made simple. Available: https://lunrjs.com.
[11] Xapian: Open Source Search Engine Library. https://xapian.org.
[12] М. И. Забежайло, “О некоторых возможностях управления перебором в ДСМ-методе,” *Искусственный интеллект и принятие решений*, Часть I: № 1, С. 95-110, Часть II: № 3, С. 3-21, 2014
[13] А. А. Грушо, М. И. Забежайло, А. А. Зацаринный, Е. Е. Тимонина, “О некоторых возможностях управления ресурсами при организации проактивного противодействия компьютерным атакам,” *Информатика и ее применения*, Т. 12, № 1, С. 62-70, 2018.
[14] М. И. Забежайло, “О некоторых оценках сложности вычислений в ДСМ-рассуждениях,” *Искусственный интеллект и принятие решений*, Часть I: №1, С. 3-17, Часть II: №2. С. 3-17, 2015.
[15] А. А. Грушо, М. И. Забежайло, Д. В. Смирнов, Е. Е. Тимонина, "О комплексной аутентификации," *Системы и средства информ*., Т. 27, Вып. 3, С.4–11, 2017.
[16] А. А. Грушо, М. И. Забежайло, Д. В. Смирнов, Е. Е. Тимонина, "Модель множества информационных пространств в задаче поиска инсайдера,” *Информатика и ее применения*, Т. 11, № 4, С. 65-69, 2017.
[17] А. А. Грушо, Н. А. Грушо, М. И. Забежайло, Д. В. Смирнов, Е.Е. Тимонина, “Параметризация в прикладных задачах поиска эмпирических причин,” *Информатика и ее применения*, Т. 12, № 3, С. 62-66, 2018.
[18] А. А. Грушо, М. И. Забежайло, Д. В. Смирнов, Е. Е. Тимонина, С. Я. Шоргин,” Методы математической статистики в задаче поиска инсайдера,” *Информатика и ее применения*, Т. 14, Вып. 3, С. 71-75, 2020.
[19] А. А. Грушо, М. И. Забежайло, Д. В. Смирнов, Е. Е. Тимонина, “О вероятностных оценках достоверности эмпирических выводов,” *Информатика и ее применения*, Т. 14, Вып. 4, С. 3-8, 2020.

System for collecting and analyzing information from various sources in Big Data conditions

D. V. Smirnov, A. A. Grusho, M. I. Zabezhailo, E. E. Timonina

***Abstract*— The problem of constructing an architecture and methods of searching for insider activity signs in process-real-time conditions has been investigated. The problem is solved in the following conditions. The source of "raw" data is Big Data, from which data relevant to insider activity signs is selected in accordance to the current list of threats. Search is conducted on a large number of users. Under these conditions, the algorithm is built that breaks the difficulty barrier in finding relevant data.**

The important complication of the task is the conditions of "openness" of the data. The condition of "openness" of the data involves constant updating of the data. The concept of "openness" also includes changing the signs of hostile activities of insiders. In this case, the search conditions can also dynamically change.

The built architecture is two-level. The first level contains data collected from various "raw" databases and relevant to the current list of threats. The second level relates to the maximum availability of data organized at the first level for analysis with the participation of experts - operational workers. Scientific justification of correctness and efficiency of mathematical models and big data mining algorithms involved in implementation of this software system is given.

The built solutions showed their operability in the industrial version of the solution of the problem.

***Keywords*— Information security, searching signs of hostile insider in Big Data, intelligent data analysis.**

REFERENCES

[1] Welcome to Apache Lucene. Available: https://lucene.apache.org.
[2] The Apache Software Foundation. Available: https://lucy.apache.org.
[3] O. Bartunov, “Do you need a Full-Text Search in PostgreSQL?”, in PGConf.eu, Oct 26, Lisbon, 84 p., 2018. Available: https://www.postgresql.eu/events/pgconfeu2018/sessions/session/2116/slides/137/pgconf.eu-2018-fts.pdf.
[4] Open-source database for search applications. Available: https://manticoresearch.com.
[5] INDRI: Language modeling meets inference networks. Available: https://www.lemurproject.org/indri/.
[6] MySQL: Full-Text Search Functions. Available: https://dev.mysql.com/doc/refman/8.0/en/fulltext-search.html.
[7] Welcome to the Terrier IR Platform. Available: http://terrier.org.
[8] NLP-Center: NoSketch Engine. Available: https://nlp.fi.muni.cz/trac/noske.
[9] ArangoDB: Powerful Search Included. Available: https://www.arangodb.com/full-text-search-engine/.
[10] LUNR: Search made simple. Available: https://lunrjs.com.
[11] Xapian: Open Source Search Engine Library. https://xapian.org.
[12] M. I. Zabezhailo, “To the some new possibilities to control computational complexity of hypotheses,” *Scientific and Technical Information Processing*, Part I: no. 1, pp. 95-110, Part II: no. 3, pp. 3-21, 2014.
[13] A. A. Grusho, M. I. Zabezhailo, A. A. Zatsarinny, E. E. Timonina, “On some possibilities of resource management for organizing active counteraction to computer attacks,” *Informatics and Applications*, vol. 12, no. 1, pp. 62-70, 2018.
[14] M. I. Zabezhailo, “To the computational complexity of hypotheses generation in JSM-method,” *Scientific and Technical Information Processing*, Part I: no. 1, С. 3-17, Часть II: no. 2. С. 3-17, 2015.
[15] A. A. Grusho, N. A. Grusho, M. I. Zabezhailo, D. V. Smirnov, E. E. Timonina, “About complex authentication,” *Systems and Means of Informatics*, vol. 27, no. 3, pp. 3-10, 2017.
[16] A. A. Grusho, M. I. Zabezhailo, D. V. Smirnov, E. E. Timonina, “The model of the set of information spaces in the problem of insider detection,” *Informatics and Applications*, vol. 11, no. 4, pp. 65-69, 2017.
[17] A. A. Grusho, N. A. Grusho, M. I. Zabezhailo, D. V. Smirnov, E. E. Timonina, “Parametrization in Applied Problems of Search of the Empirical Reasons,” *Informatics and Applications*, vol. 12, no. 3, pp. 62-66, 2018.
[18] A. A. Grusho, M. I. Zabezhailo, D. V. Smirnov, E. E. Timonina, S. Ya. Shorgin, “Mathematical statistics in the task of identifying hostile insiders,” *Informatics and Applications*, vol. 14, no. 3, pp. 71-75, 2020.
[19] A. A. Grusho, M. I. Zabezhailo, D. V. Smirnov, E. E. Timonina, “On probabilistic estimates of the validity of empirical conclusions,” *Informatics and Applications*, vol. 14, no. 4, pp. 3-8, 2020.