МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ АВТОМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

Кафедра Автоматизированных систем управления

ОТЧЕТ ПО ПРАКТИКЕ

Учебная практика: технологическая (проектно-технологическая) практика

(наименование практики в соответствии с учебным планом)

Направление подготовки: 09.04.03 Прикладная информатика
профиль: Интеллектуальный анализ и управление в социально-экономических системах

Выполнил:		**Проверил:**	
Студент	*Нерянов П.А.*	Преподаватель	*Муртазина М.Ш., Доцент кафедры АСУ, к.т.н.*
Факультет	*АВТФ*		
Направление (специальность) подготовки	*09.04.03– Прикладная информатика*	Балл: _____	
Группа	*АПМ2-20*	Оценка _____	
Шифр	013455911		
_____ подпись		_____ подпись	
Дата сдачи:	«__»_____20__г.	Дата защиты:	«__»_____20__г.

Новосибирск 2021

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Управления рисками информационной безопасности и выполнение мероприятий по снижению, выявленных рисков до требуемого уровня, является обязательным требованием при проектировании информационной сети компании и обеспечению её безопасности. Во многих кредитных организациях уже давно сложилась практика управления рисками информационной безопасности, основу которой составляют определение актуальных угроз ИБ и планирование мероприятий по их нейтрализации.

Корпоративная сеть является узким звеном предприятия, учитывая глобальную тенденцию к росту числа информационных атак, приводящих к значительным финансовым и материальным потерям. Согласно данным предоставленным главой МВД, только за январь-июнь 2020г. рост IT-преступности в России составил 91,7% по сравнению с аналогичным периодом прошлого года [1].

Объектом исследования является система менеджмента информационной безопасности. Предметом исследования система управления рисками информационной безопасности. В настоящее время интенсивного развития цифровизации, невозможно, представить компанию не использующую в своей деятельности работу с информацией и коммуникационными технологиями. При создании системы менеджмента необходимо провести наблюдения за сетью исследуемой компании, эксперимент с данными, а также сравнить полученные результаты и на их основе сделать выводы.

Цель исследования - создание онтологической модели представления знаний по информационной безопасности в рамках прецендентного подхода к описанию угроз. Для её достижения работа были выделены следующие задачи:

1) Систематизировать знания из научных публикаций и классических датасетов об атаках и их влиянии на активы;

2) Обучить нейронную сеть для определения нормальных и аномальных соединений, представленных в датасете KDD CUP 1999;

3) Построить онтологическую модель представления знаний, которая обеспечить поддержку процесса анализа рисков информационной безопасности корпоративной сети компании.

Полученная, в рамках исследования, онтологическая модель будет объединять знания об атаках, распостранённых в компьютерных сетях, и влиянии данных атак на активы, доступ к которым можно получить через корпоративную сеть.

1 Теоретико-методологические основы управление рисками информационной безопасности в корпоративной сети компании

1.1 Управление рисками информационной безопасности как область знаний

Риски информационной безопасности в корпоративной сети — это предпосылки к возникновению прямого или косвенного ущерба для компании. Данный ущерб может отразится на работе компании и привести к потере выгоды, что вследствие может привести к банкротству. Всеми процессами при ведении бизнеса необходимо управлять и выделять на это необходимые ресурсы-активы.

Актив - основные ценные ресурсы предприятия. Данные ресурсы являются доходом предприятия. На данный момент разделяют материальные, финансовые, человеческие, информационный активы и процессы.

Материальные активы — это предметы, которые имеют физическое содержание и могут быть использованы для производства или поставки продуктов и услуг. С точки зрения бухгалтерского учета, материальные активы также включают договоры аренды или акции компании. По сути, они являются основными активами организации и могут отличаться от нематериальных активов, таких как деловая репутация, товарные знаки или патенты, которые не имеют физического содержания, но по-прежнему являются ценными концепциями и активами [2].

Финансовые активы — это денежные средства, право требовать по договору денежные средства или другой финансовый актив, право обмена на другой финансовый инструмент, долевой инструмент [3].

Процесс (актив) — это главный актив, он используется для достижения необходимых целей компании объединяя в себе остальные активы организации.

Еще один актив для компании - её репутация. Информационная безопасность неотъемлемая часть имиджа компании. При утечке

конфиденциальных данных пользователи перестают доверять компании, и её репутация на рынке стремительно падает. Для примера обычный сбой в работе сервера компании, предоставляющей свои товары онлайн может повлиять на его доступность для потенциальных покупателей. Вследствие этого потенциальные потребители приобретут товар у конкурентов, что повлечет к финансовым потерям. Так же для восстановления правильной работы сервера необходимо привлекать человеческие активы переводя их с другого бизнес-процесса, вызывая дефицит ресурсов уже на нём. Всё это влияет на репутацию компании.

Из приведенного примера видно, что бизнес-процессы в организации подвершены большому количеству факторов риска (внутренние и внешние).

У каждой компании существуют более важные активы (основные) и активы, которые менее важны (вспомогательные). Например, если организация занимается управлением финансов (кредитная деятельность, инвестиции и т.д.), то для неё основные активы это финансовые. Для предприятия, специализирующегося на продаже недвижимости, оборудования, ресурсов основные это материальные активы. Если бизнес основан на наукоемкой отрасли или разработки ПО и информационной собственности, то для него важнейшим являются информационные активы.

В каждой компании существуют различные по значимости активы. Например, если в компании произошел риск при использовании основных активов, то это чревато потерей бизнеса. Однако если риск произошел среди вспомогательных активов, то данный ущерб возможно возместить. Именно поэтому ликвидация рисков основных активов является приоритетной задачей глав компании и её сотрудников. В то время как решение рисков второстепенных активов может быть вообще отправлено на аутсорсинг, т.к. от них зависит в большинстве случаев лишь эффективность управления в организации.

Т.к. не для всех компаний риски информационной безопасности являются основными, то возможны различные способы их управления. Данные способы различаются глубиной и рассчитаны на различное участие информационных активов в производственном процессе компании.

Первый способ рассчитан на компании, в которых информационные активы – вспомогательные активы, а уровень информатизации низок, т.е. оценка информационных рисков для такой компании не является приоритетной задачей. Такой организации при своей работе рекомендуется базовое обеспечение информационной безопасности согласно существующим нормам и стандартам ISO/IEC. Данные нормы и стандарты помогут владельцам компании или лицам, которые занимаются ликвидацией данных рисков на аутсорсинге, т.к. актив является вспомогательным, определить механизмы, которые применимы для данной компании.

Второй способ рассчитан на компании, в которых информационные активы по-прежнему являются вспомогательными активами, однако уровень информатизации высокий, т.е. при их возникновении возможен ущерб основным бизнес-процессам. В таком случае рекомендуется так же использовать базовое обеспечение информационной безопасности согласно существующим нормам и стандартам ISO/IEC и уделить особое внимание наиболее важным участкам.

Третий способ рекомендуется использовать IT-компаниям и рассчитан на глубокий уровень управления. Используется формальный подход и качественные методы.

На практике в компаниях не редки случаи одновременного использования нескольких основных активов с равной степенью важности. Для примера возьмем компанию, в которой два важных основных актива – человеческие и информационные ресурсы. Таким образом, в данной компании наиболее эффективно будет провести высокоуровневый подход оценки рисков для определения систем и их рисков с критической важностью

для функционирования бизнес-процессов компании, и с последующим выявлением риском в системах, определенных на предыдущем этапе. А для остальных рисков воспользоваться базовым обеспечением информационной безопасности согласно существующим нормам и стандартам ISO/IEC.

Понятие риска

Существуют различные точки зрения на понятие риска.

“Риск как возможность - имеет в своем основании концепцию существования взаимосвязи между риском и доходностью [4]. Выше риск, следовательно, выше и доход, однако в то же время и вероятность потерпеть убытки. Так как риск является вероятностной величиной, то он имеет два исхода: наличие риска или его отсутствие. Если риск произошел, то существуют три его вариации:

1) отрицательный (убытки),

2) нулевой (безубыточность, бесприбыльность),

3) положительный (прибыль).

Под экономическим риском понимается стоимостное выражение события, ведущего к потерям. Поскольку возможный убыток измеряется в денежном выражении, величина (уровень) риска также выражен в денежных единицах. Можно выделить следующие особенности экономического риска:

- Случайная возможность возникновения убытка;
- Денежное измерение убытка,
- Нежелательность возникновения убытка,
- Наличие возможностей предотвращения убытка или самого события,
- Количественная оценка риска.

Нулевой риск обеспечивает наименьший доход, а наивысшем риске прибыль может иметь наибольшее значение, т.е. высокий риск связан с вероятностью извлечения большего дохода.

Риск как опасность или угроза - возможность наступления событий с негативными последствиями, т.е. возможность реализации предполагаемой

опасности [4]. Нет уверенности в том, что это событие произойдет, но его возникновение повлечет неблагоприятный исход: возникновение потерь (недополучение доходов), помешает получить заданный результат, снизит качество или надежность этого результата и т.п.

Под риском понимается возможная опасность потерь, вытекающая из специфики тех или иных явлений природы и видов деятельности человека.

Риски информационной безопасности рассматриваются именно в рамках данной формулировки. Таким образом цель управления будет заключаться в распределении ресурсов для минимизации вероятности возникновения неблагоприятных событий.

Риск как неопределенность - наличие неопределенных факторов, при которых возникает возможность возникновения непредусмотренных событий, приводящих к отличию действительного исхода от планируемого [4].

Неопределенность носит объективный характер, обусловленный наличием факторов недетерминированной природы, благодаря которым конечный результат принимаемого решения не может быть однозначно определен. В этом случае результат фактически может быть определен как вероятностный выбор из пространства исходов.

Любая целенаправленная деятельность человека, направлена в будущее, т.е. всегда существует временной разрыв между начальными усилиями, сопровождающимися затратами ресурсов и конечным результатом. Отсюда следует, что конечный результат заведомо не детерминирован.

Неопределенность - атрибут и фактор принятия решений [5]. Неопределенность показывает факторы недетерминированной природы, их наличие и влияние, которое они оказывают на результат принятия решения;

Существуют следующие причины неопределенности:

1. Незнание − неполнота знаний и представлений об ситуации,

2. Случайность. Спланировать каждый случай невозможно,

3. Противодействие − технические аварии и сбои, действия нарушителей др.

Наиболее рациональным подходом к определению риска является понимание его как совокупности вероятности события и тяжести его возможных последствий/размера ущерба. Именно такой подход принят в стандартах управления рисками информационной безопасности [6].

Факторы риска

Факторы риска – потенциально опасные участки системы: ценные ресурсы предприятия или факторы и события, при которых возможен ущерб системе.

Подходы, описанные выше при своем исполнении, оперируют факторами риска, на основе которых принимается решение какую применить контрмеру для ликвидации риска. Для рисков информационной безопасности выделяют 7 факторов риска:

- Основные ценные ресурсы предприятия (Активы),

- Ущерб - затраты на восстановление системы в работоспособное состояние после возможного инцидента информационной безопасности, а также на восстановление искаженной, утерянной информации или же нейтрализации последствий утечки конфиденциальной информации [7],

- Угроза - потенциально возможное событие, действие, процесс или явление, которое может привести к нанесению ущерба [8],

- Уязвимость - присущие объекту информатизации свойства, приводящие к нарушению безопасности информации на конкретном объекте и обусловленные особенностями процесса функционирования объекта информатизации, свойствами архитектуры автоматизированной системы, протоколами обмена и интерфейсами, применяемыми программным 10

обеспечением и аппаратной платформой, а также условиями эксплуатации [9],

- Механизм контроля – эффективное управление участниками бизнес-процесса управлении рисками,

- Материальные и финансовые потери,

- Банковские операции.

Оценка рисками информационной безопасности и методы анализа применяются согласно выбранной в компании методологии. Данный процесс состоит из двух этапов: анализ рисков и оценка рисков.

При анализе рисков следуют следующей последовательности:

Для начала члены компании определяют основной(-ые) активы компании и определяют их ценность. Далее выявляют потенциальные угрозы для данных активов, вероятность их возникновения и убытки. Заключающей фазой этапа является определения вероятности уязвимости организации.

При оценке рисков члены компании под руководством начальства отвечают на следующий вопрос: Какой уровень риска (материальные и финансовые потери) является приемлемым для организации.

По результатам ответа на данный вопрос (оценки рисков), получают риски, из-за которых текущий уровень риска не соответствует (соответствует) требованиям компании.

Процесс оценки рисков заключается в следующем:

На начальном этапе выбирается способ обработки риска и механизмы контроля. Далее происходит оценка эффективности выбранных способов и методов. В конце составляется план обработки рисков, в котором учитывается стоимость контрмер и сроки их реализации.

Оценка рисков

При оценке рисков необходимо продумать последовательность всех шагов и необходимую для них детализацию выполнения. В классических методиках предусмотрено определение и оценка активов, определение

уязвимостей, разработка модели угроз и модели нападения. Формализованные результаты необходимо оценить, классифицировать и графически отобразить.

При оценке рисков определяется его вероятность для нахождения величины ущерба. Есть два метода для определения вероятности риска: прогнозированием (на основе обнаруженных уязвимостей и потенциальных нападающих) или на основе существующей статистики (классификация уязвимостей).

CVSS – одна из систем классификации уязвимостей, которая позволяет определить основные характеристики уязвимости и получить числовую оценку, и преобразовать её в качественное представление (например, низкое, среднее, высокое и критическое), чтобы помочь организациям правильно оценить и определить приоритеты своих процессов управления уязвимостями [10].

Ущерб определяется для каждого актива с обнаруженной уязвимостью. Величина ущерба не во всех случаях является денежной или процентной величиной. Так как в большинстве случаев полученные результаты в таком виде представить невозможно из-за методов оценки рисков.

Процесс анализа рисков является не тривиальной задачей. Для успешности данного процесса необходимо учесть множество факторов, поэтому использование одних лишь методических рекомендаций будет недостаточно для получения требуемого результата. Таким образом помимо методических рекомендаций необходимо выбрать инструментарий с поддержкой необходимых методов и стандартов, а также соблюсти регламент управления рисками. Данный процесс помимо информационных ресурсов неразрывно связан и с человеческими активами, т.е. различными структурными подразделениями компании и привлечением в процесс менеджеров компании, которые обеспечат управление структурных подразделений.

1.2 Вторжения в корпоративную сеть компании как проблема управления рисками

Корпоративная сеть является узким звеном предприятия, учитывая глобальную тенденцию к росту числа информационных атак, приводящих к значительным финансовым и материальным потерям.

Злоумышленники находят, как и новые уязвимости в программной коде всемирно-популярного ПО, так и эксплуатируют старые. Например, в декабре 2021-го года была раскрыта уязвимость Log4Shell библиотеки Log4j (используется во множестве приложений, написанных на языке Java). Данная уязвимость позволяет запускать вредоносный код на серверах потенциальных жертв и использовать его для управления сетью и кражи данных. По результатам анализа более 200 корпоративных облачных сред известными аудиторско-консалтинговыми компаниями (WIZ и YE) уязвимости были подвержены более 93% всех облачных сред. Apache Foundation выпустила патч для исправления, но спустя 10 дней после публичного объявления организации исправили только 45% своих ресурсов. Учитывая тот факт, что данная уязвимость легка в использовании и её оценка серьёзности CSVV имеет наивысший балл, сложно представить общий нанесенный ущерб компаниям.

Таким образом, любое вторжение в корпоративную сеть влияет не только на её корректное функционировании, но и данные, хранящиеся на серверах компании. Потеря контроля над их распространением пагубно влияет на имидж компании из чего следует риск в потере потенциальных клиентов.

1.3 Стратегии обнаружения вторжений

Системы обнаружения вторжений отличаются от традиционных средств защиты. В то время как антивирус проверяет файлы, firewall анализирует соединения, а спам-фильт письма - система обнаружения вторжений обеспечивают более высокий уровень защиты. Хоть система

обнаружение вторжений и выявляет угрозы на основании анализа трафика, дальнейшие действие возлагаются на системного администратора предприятия.

При планировании стратегии обнаружения вторжений необходимо определиться с требованиями, которые мы возлагаем на систему обнаружения вторжений.

Так при установке системы перед firewall необходимо использовать такую её разновидность, как NIDS. Данная система поддерживает глубокий анализ пакетов и анализирует все пакеты с канального уровня до прикладного уровня (уровень приложений). Недостаток данной системы заключается в том, что она значительно повышает нагрузку на сеть.

Можно установить систему обнаружения вторжений за firewall. При таком раскладе, необходимо использовать такой тип системы как PIDS. Данная система будет анализировать трафик с транспортного уровня до прикладного уровня. Тем самым снижается нагрузка на сеть, в отличии от системы NIDS.

Так же существуют узконаправленные системы, для защиты одного хоста или димилитаризированной зоны сети.

1.4 Применение системы обнаружения вторжений для мониторинга рисков

Использование систем обнаружения вторжений является одним из шагов для использования множества моделей и методов мониторинга информационной безопасности при управлении рисками. Например, стандарт BS 7799-3[11] и наследуемый от него ISO 27001[12] определяют существующие 4-х фазные процессные модели управления рисками. Определяют 4 фазы процессной модели: планирование, реализация, проверка, действия.

1 фаза – Планирование. Во время исполнения данной фазы происходит выбор методики управления рисками и производится оценка рисков и требуемого уровня рисков.

2 фаза – Реализация. Выполняется обработка рисков, обнаруженная на предыдущей фазе, и применяются необходимые механизмы устранения для обнаруженных рисков (игнорирование, минимизация, ликвидация, экстрадирование).

3 фаза – Проверка. Производится аудирование существующих механизмов контроля, который проходит в 3 этапа: разработка регламента проведения аудита, сбор информации, составление отчета уровня безопасности.

4 фаза – Действия. По результатам проведенного аудита, в случае необходимости, производятся действия, связанные с переоценкой рисков, а также изменением выбранных методов оценки и управления рисками.

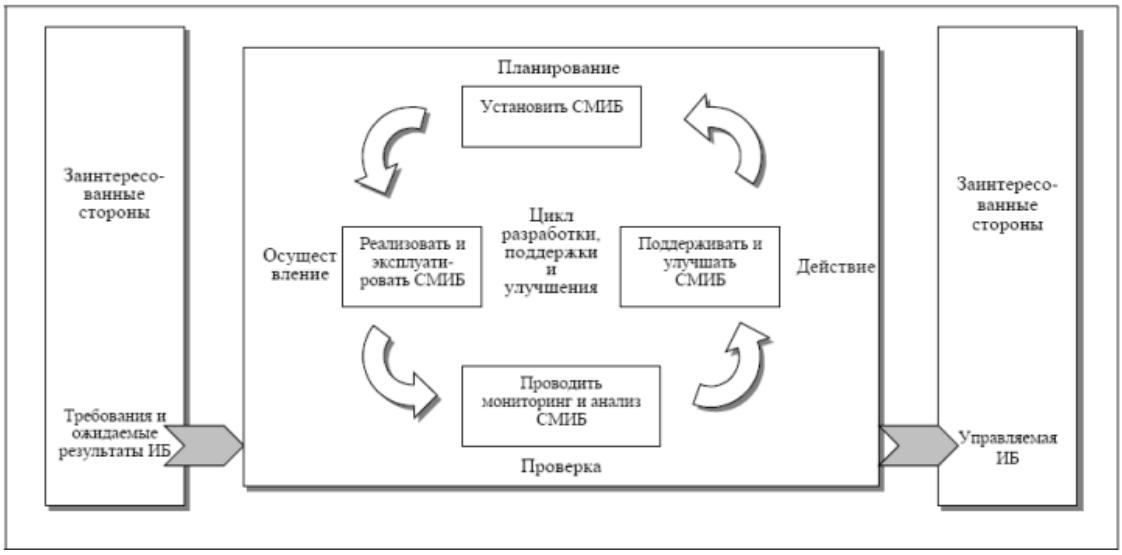

Рисунок 1 – 4-х фазная процессная модель управления рисками

Так же существует большое множество методов управления рисками, как количественные, так и качественные, в которых присутствует шаг по обнаружению уязвимостей в исследуемой системе.

Методы управления рисками можно разделить следующим образом:

1) качественные методы,

2) количественные методы,

3) сочетание качественных и количественных методов (смешанные методы).

На данный момент не существует явных причин, при которых строго регламентирован выбор того или иного метода управления. Приведем некоторые из наиболее известных методов управления рисками, которые хорошо себя зарекомендовали.

OCTAVE – Оценка критичных угроз, активов и уязвимостей (Operationally Critical Threat, Asset, and Vulnerability Evaluation) [13]. Данный метод рассчитан на организации различных сфер и объёмов производства, и состоит из 3-х этапов, не включающих планирование. Во время планирования разрабатывается график семинаров и назначение ролей.

Другой метод оценки рисков - CRAMM (CCTA Risk Analysis and Management Method)[14]. Основа данного метода — это оценка рисков посредством проведения опросов. Данный метод востребован большим количеством компаний из-за удобного программного обеспечения для управления. Данное ПО содержит постоянно обновляющуюся базу знаний рисков и контрмер, а также возможность определить величину риска.

Различие с предыдущим методом заключается в подготовительном этапе и последовательности действий. Подготовительный этап в данном случае заключается в определении целесообразности проведения оценки рисков и уже при необходимости проводится глубокая оценка рисков. В противном же случае используются способы, уже содержащиеся в базе знаний инструментария CRAMM.

Первый этап заключается в оценке активов и определяются связи между различными активами.

Второй этапе заключается в оценке рисков и его вероятности. Риски определяются для связки: Ресурс-Угроза-Уязвимость. Контрмеры к рискам на данном этапе не производятся.

Заканчивается анализ определением контрмер к обнаруженным рискам на основе базы знаний CRAMM. Полученный результат сравнивается с требуемым уровнем рисков.

Метод FRAP (Facilitated Risk Analysis Process) является качественным методом. В методе FRAP используются следующие этапы при оценке рисков:

- Определение защищаемых активов (автоматизированный анализ сети),

- Идентификация угроз,

- На основе идентифицированных угроз производится возможная оценка нанесенного ущерба (по следующим уровням: - Низкий; -Средний; - Высокий),

- Определение контрмер согласно выявленным угрозам,

- Документирование.

Таким образом, основным достоинством данного метода является использование минимальных трудозатрат. К минусам такого подхода можно отнести отсутствие глубокой декомпозиции при анализе угроз, а также отсутствие возможности оценки рисков в денежном эквиваленте.

Многие методы основное свое внимание уделяют построению комплексного процесса управления рисками, а вопросы непосредственной оценки рисков рассматривается весьма поверхностно, поэтому данную работу можно переложить на системы обнаружения вторжений.

Метод FAIR (Factor Analysis of Information Risk) разработан для устранения этого недостатка. Он не является самостоятельным методом по управлению рисками, но может использоваться в дополнение ко многим

другим методам. FAIR анализ подразумевает следующую последовательность операций:

1) Сбор данных по каждому фактору;

2) Экспертная оценка;

3) Анализ сценарий для каждого фактора (использование калиброванных значений PERT);

4) Использование инструментов стохастического моделирование (метод Монте-Карло) [15, с. 93]. Основу при факторном анализе информационных рисков составляет декомпозиция рисков на частоту появления инцидента и вероятность потерь от его наступления. А также последующую декомпозицию каждого события. Так как автор данной метода рекомендует использовать его в сочетании с другим методом, основанным на качественно оценке. Главным минусов данного метода является её несамостоятельность и узкая направленность.

Однако, данный метод позволяет проводить глубокую декомпозицию при анализе угроз.

Другой метод, представленная корпорацией Microsoft в 2006г. является методом, который использует как качественную, так и количественную оценку угроз. Аналитика рисков, согласно методу Microsoft, позволяет проводить оценку потенциальных рисков, связанных с инсайдерской деятельностью в организации, без настройки каких-либо политик риска, связанных с инсайдерской деятельностью. Эта оценка может помочь организации определить потенциальные области повышенного риска пользователей и определить тип и область политики управления рисками, которые можно настроить [16].

Данный метод включает в себя следующих 4 этапа:

1) Оценка рисков,

2) Поддержка принятия решений,

3) Реализация контроля,

4) Оценка эффективности программы.

Такой подход к управлению рисками информационной безопасности позволяет использовать непрерывный цикл реализации процесса и охватить все аспекты том числе и эффективность процесса управления рисками. Конечно, у такого подхода тоже имеются свои минусы, такие как высокая трудоемкость процесса и дополнительные затраты т.к. не подразумевает использование типовых рисковых сценарий.

Выбор метода, который необходимо использовать на предприятии является не тривиальной задачей. При выборе метода нужно учитывать такие аспекты как необходимость детального изучения рисков информационной безопасности и необходимые трудозатраты для реализации процесса. Таким образом, для различных организаций будет правильной свой метой, однако разрабатываемый компанией метод должен соответствовать потребностям организации, обеспечивать повторное использование ранних результатов и быть прозрачной для всех заинтересованных сторон в компании.

Оценка рисков зачастую является циклическим процессом. Допустим выявив некоторый риск, и затем проведя необходимые процедуры для минимизации его влияния на бизнес-процесс необходимо снова провести оценку влияния данного риска, для оценки эффективности проведенным контрмер, зачастую помогают системы обнаружения вторжений, который оценивают текущее и раннее состояние сети с точки зрения вторжений. Так же при оценке рисков необходимо использовать базу данных для хранения информации обо всех этапах оценки (документация, используемые стандарты, реестры ресурсов, текущие перечни уязвимостей и др.) и алгоритм для работы с данными.

Инструментарии, которые предлагают такие возможности существует многое количество. Например RA2[17], который поддерживает метод CRAMM и стандарты ISO 27001 или vsRisk[18], Callio Secura

17799[19], которые в своей работе запрограммированы на следование стандартов BS 7799-3. Помимо зарубежных существуют и отечественные системы для автоматизации управления рисками. Один из них это РискДетектор[20].

Использование систем для автоматизации управления рисками в совокупности с системами обнаружения вторжений позволяет проводить многократный циклический процесс переоценки рисков. Так же появляется возможность комфортной работы со всеми данными по оценки рисков бизнес-процессов, а более продвинутые инструментарии могут содержать и дополнительные средства для работы (разработка реестров ресурсов, документирование СУИБ и др.).

2 Проектирование детектора обнаружения сетевых вторжений и формирование онтологии

Основная задача практической части работы заключается в создании онтологической модели представления знаний по информационной безопасности в рамках прецендентного подхода к описанию угроз.

Для достижения данной задачи, она была разделена на следующие подзадачи:

1) Спроектировать систему обнаружения вторжений, которая использует прогностическую модель для определения “нормального” и “аномального” соединения;

2) Построить онтологическую модель представления знаний, которая обеспечить поддержку процесса анализа рисков информационной безопасности корпоративной сети компании.

2.1 Проектирование системы обнаружения вторжений

Система обнаружения вторжений – система для мониторинга сети или системы на предмет вредоносных активности[21]. Под “аномальным” соединением подразумевается так называемые вторжения в сеть или атаки на

неё, под “нормальным” же соединением являются добронамеренные подключения.

При проектировании системы обнаружения вторжений использовался классический датасет KDD CUP 1999[22]. Данные для датасета собирались программой DARPA 1998 года по оценке обнаружения вторжений.

Данный датасет имеет около 500 тыс. записей. Каждая запись имеет 41 атрибут[23] и представлена в следующем формате: 0,tcp,http,SF,181,5450,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.00,0.00,9,9,1.00,0.00,0.11,0.00,0.00,0.00,0.00,0.00,normal. Датасет содержит 23 типа соединений. Эти соединение можно разбить на 5 основных типов. 1 тип это нормальное соединение пользователя без попытки нанести какой-либо ущерб. Второй тип соединений относится к классу атак DoS – атаки «отказ в обслуживании», которые поражают современный Интернет, является серьезной проблемой для любой новой архитектуры и заслуживает полного внимания[24]. DoS атаки делают сеть или какой-либо сервис недоступным, наводняя цель траффиком или отправляя ей информацию, которая может вызвать сбои. Следующий тип атак Probe - это атака, при которой злоумышленник сканирует машину или сетевое устройство, с целью обнаружения слабых мест или уязвимостей, которые впоследствии могут быть использованы. Однако, профили атак, созданные с использованием популярных элементов поддоменов, имеют потенциальный недостаток в том, что их легко обнаружить, особенно если создается большое количество таких профилей. Хотя злоумышленник может попытаться изменить используемые элементы, тем не менее, может существовать отличительная сигнатура атаки[25]. Предпоследний тип соединений в датасете это R2L, которая используется злоумышленником для получения несанкционированного доступа к хосту в атакуемой сети. 5 тип соединения U2R - данные атаки представляют собой использование, при котором хакер начинает в системе с

обычной учетной записью пользователя и пытается воспользоваться уязвимостями в системе, чтобы получить привилегии суперпользователя[26].

2.1.1 Анализ датасета KDD CUP 1999

Перед началом детектирования соединений был проведен анализ данных из датасета для определения преобладающего типа атак, а также выявлении некоторых закономерностей для учета их при настройке работы нейросети, с помощью которой будет осуществляться определение аномального соединения.

Для проведения анализа был написан модуль программы на языке Python с использованием библиотеки Pandas.

По результатам анализа получены следующие данные. Общее число соединений, представленных в датасете, 494021. Из них 97278 – соединения типа normal; 391458 – соединения типа атаки-DoS; 4107 – соединения типа атаки-Probe; 1126 – соединения типа атаки-R2L; 52 – соединения типа атаки-U2R. При установлении соединения происходит обмен пакетов, а также возможные попытки переподключения при нестабильном качестве коннекта. Поэтому для исключения попадания одного и того же подключения были удалены из датасета одинаковые данные. После их удаления общее число подключений сократилось до 145586. Среди которых 87832 – соединения типа normal; 54572 – соединения типа атаки-DoS; 2131 – соединения типа атаки-Probe; 999 – соединения типа атаки-R2L; 52 – соединения типа атаки-U2R. Графическое соотношение атак без дублирующих пакетов представлено на рисунке 2.

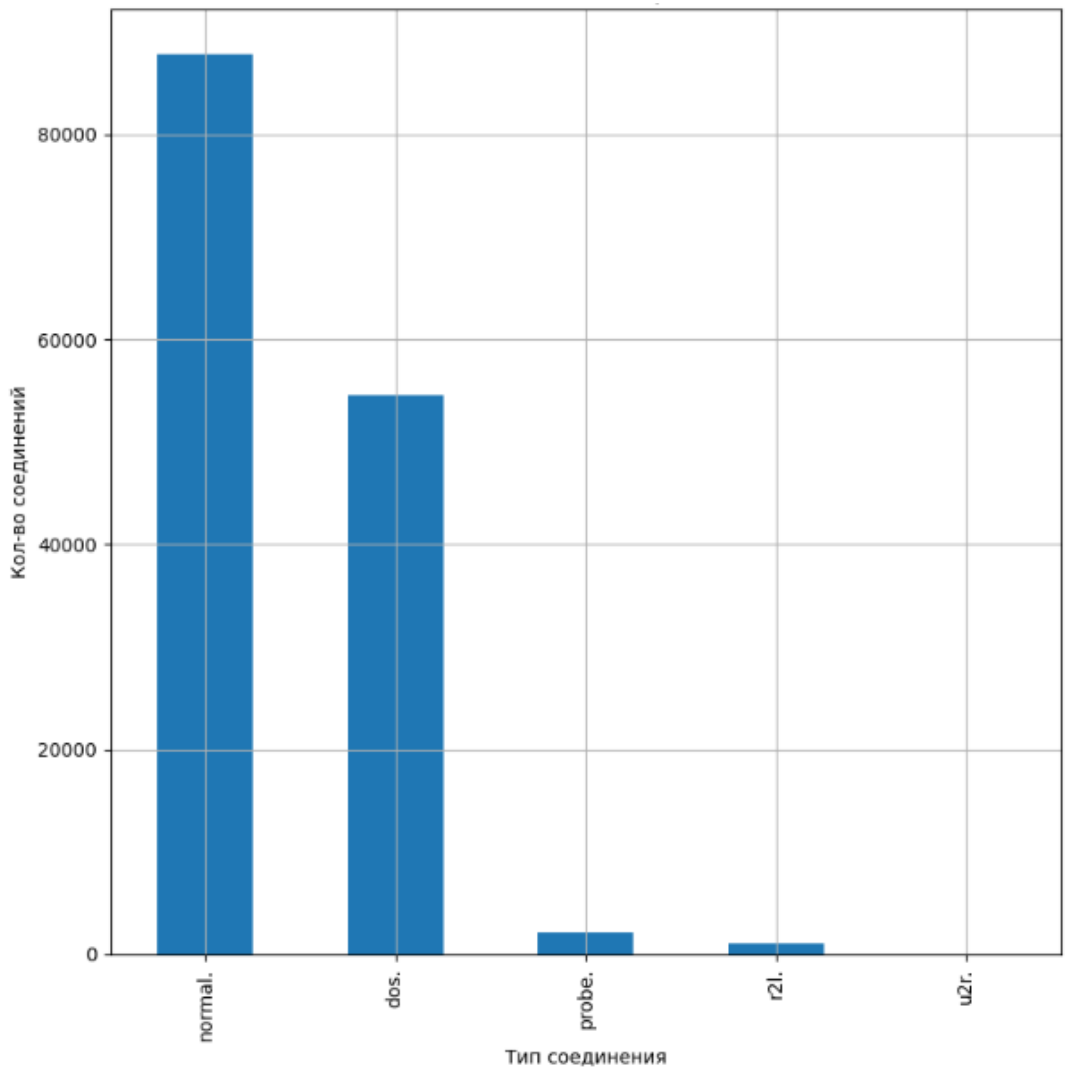

Рисунок 2 – Количество normal соединений и различных типов атак

Так же из полученных результатов можно сказать, что 145546 соединения из 145566 подключается к/от одному и тому же хосту/порту, что может вызвать подозрения о намерениях этих подключений. По результатам анализа атрибута is_host_login, который говорит, что соединение не исходит от хоста и принадлежит ни одному из подключений. Таким образом, данный атрибут можно не учитывать при дальнейшем обучении. Большинство соединений устанавливается при помощи протокола tcp. Так же результаты анализы атрибутов, представленные в таблицах 1-2, которые отвечают за аутентификацию говорит, что зачастую используются пользователи не имеющие прав “root” или других преимуществ по правам. Полные результаты анализа представлены в ПРИЛОЖЕНИИ А.

Таблица 1 – Результаты анализа атрибута root

Значение атрибута	Количество соединений
Пользователь не “root”	145531
Пользователь “root”	55

Таблица 2 – Результаты анализа атрибута su_attempted

Значение атрибута	Количество соединений
С использование команды “sudo”	145574
Без использование команды “sudo”	12

2.1.2. Применение глубокого обучения к задаче определения аномального соединения

На данный момент в области кибербезопасности научным сообществом активно используются алгоритмы глубокого обучения. Они применяются для создания систем против угроз кибербезопасности. По результатам анализа использования методов глубокого изучения, используемых в кибербезопасности, представленных авторами Гайфулина Д.А., Котенко И.В. публикации [27], большинство подходов дают хорошие результаты для решения задач безопасности и позволяют применять их на практике.

Использования глубокого обучения предполагает подготовку данных для построения модели, так называемый препроцессинг. В работе для достижения этой цели использовались модули sklearn.preprocessing и sklearn.compose. Библиотека Sklearn[28], используется для машинного обучения в Python, а также является Open Source продуктом. Реализация препроцессинга представлена в ПРИЛОЖЕНИИ Б в файле train.py.

Далее необходимо было выбрать модель. На данный момент существует множество алгоритмов. Они различаются разной степенью точности и гибкость, а также интерпретируемостью. Зависимость гибкости алгоритмов машинного обучения и интерпретируемости полученной модели представлено на рисунке 3.

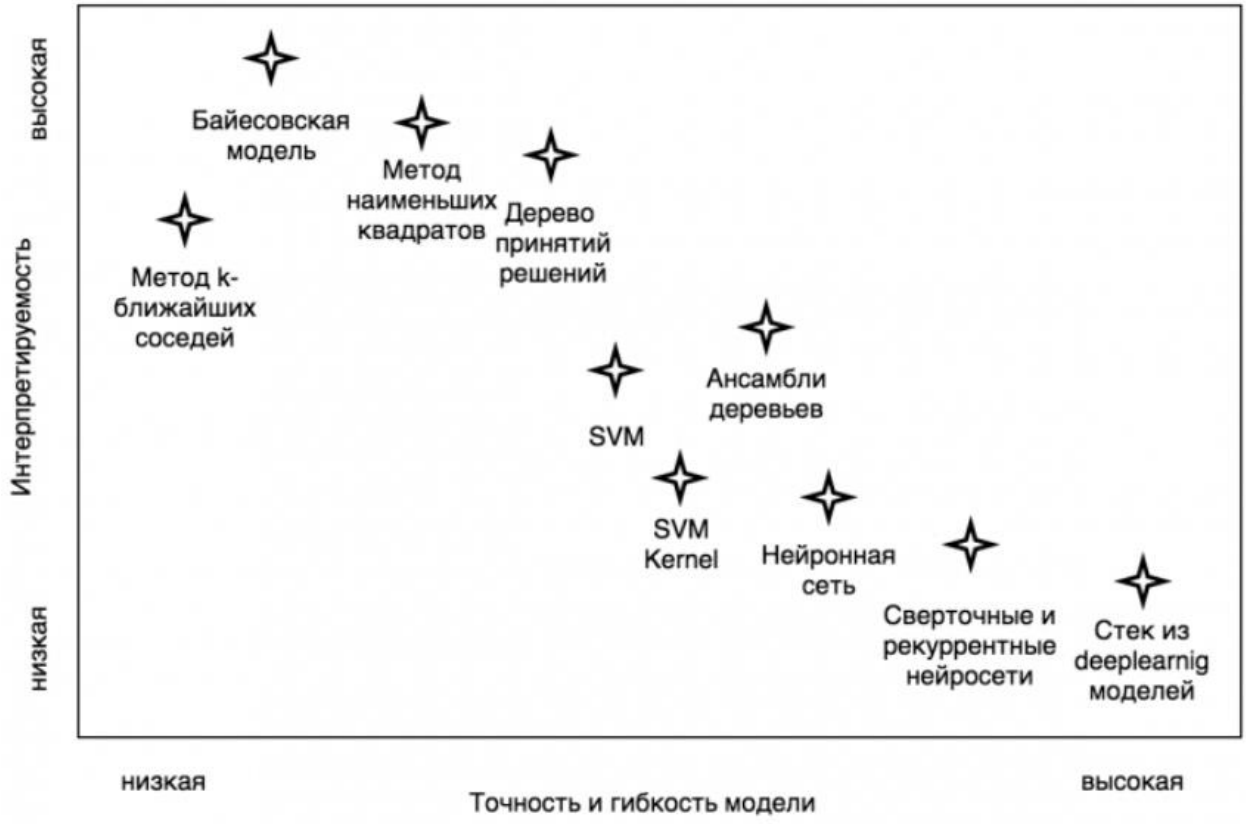

Рисунок 3 — Зависимость гибкости алгоритма машинного обучения и интерпретируемости полученной модели

В рамках данной работы было принято решение провести тестирование самой интерпретируемой модели и модели с наибольшей точностью. Т.е. Бейесовской модели и искусственной нейронной сети. Модели с наибольшей точностью были исключены т.к. нам необходимо решить задачу классификация, а они спроектированы для эффективного распознавания образов.

По результатам использования и тестирования выбранных моделей на датасете KDD CUP 1999 были получены результаты, представленные в таблицах 3-4.

Таблица 3 – Результаты использования Байесовской модели

Классификатор	Значение
Достоверность	0,948039
Полнота	0,79268
Ложноположительный	0,000234
Точность	0,999115
F-мера	0,884005
Энтропия	0,000885

Таблица 4 – Результаты использования искусственной нейронной сети

Классификатор	Значение
Достоверность	0,999346
Полнота	0,998841
Ложноположительный	0,00053
Точность	0,997855
F-мера	0,998348
Энтропия	0,002143

По полученным результатам видно, что для данных представленных в исследуемом датасете как и предполагалось наибольшая точность будет для искусственной нейронной сети.

Как было описано в подглаве 2.1 данные для датасета были собраны во время программы DAPRA 1998 года. Для сбора данных была создана локальная сеть максимально приближенная к настоящей среде ВВС США. Поэтому при помощи снифферов, возможно собрать такие данные для любой сети. Снифферы – программа для сканирования сетевого трафика[29]. Возможности данной программы позволяют просматривать данные tcp, udp,

icmp пакетов. Для наших целей используется анализатор пакетов командной строки tcpdump[30] со следующей конфигурацией:

$ tcpdump -s 66000 -F options -w datafile

«-w datafile» – имя файла дампа

«-s 66000» – для перехвата пакета целиком

«-F options» – считывание параметров фильтрации из файла

Строка в файле options имеет следующий вид:

not host 192.168.1.1 and not (host 192.168.1.1 and port ftp)

Данные настройки позволяют перехватить все пакеты на хост и с хоста 1.1

2.2 Проектирование онтологии

Для построения онтологии использовался редактор Protégé. Это бесплатный онтологический редактор с открытым исходным кодом и фреймворк для построения интеллектуальных систем Protégé, который поддерживается сообществом академических, правительственных и корпоративных пользователей, которые используют Protégé для создания основанных на знаниях решений в таких разнообразных областях, как биомедицина, электронная коммерция и организационное моделирование[31].

На основе данных представленных в датсете KDD Cup 1999 составлена онтология знаний. Её иерархия представлена на рисунке 4.

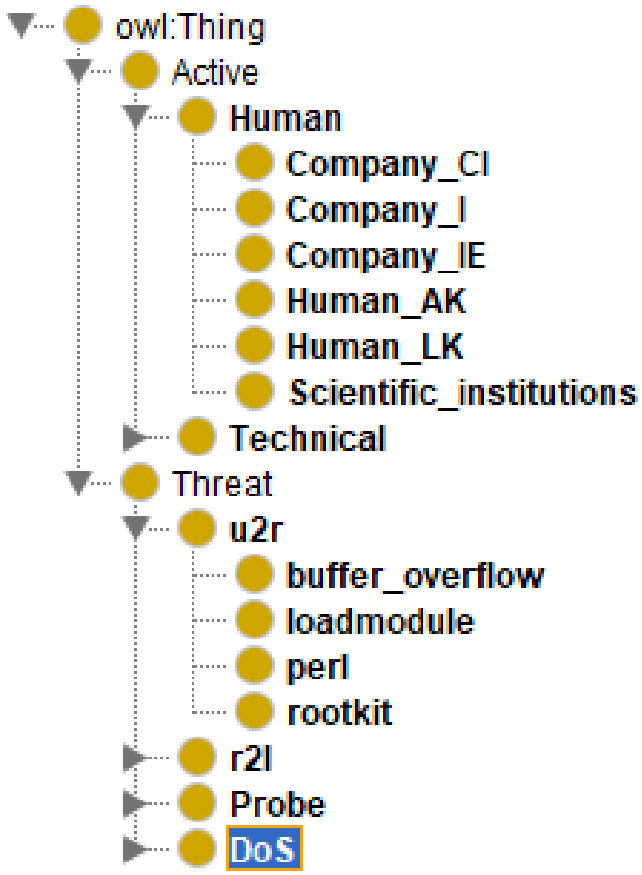

Рисунок 4 – Иерархия онтологии

Онтология состоит из 2-х основных классов «Активы» и «Угрозы». Предполагается, что обнаруженное нейронной сетью аномальное соединение поступает на вход онтологии знаний и, по результатам анализа атрибутов пользователю выдается тип угрозы и актив(ы), подверженные опасности.

Класс «Активы» разделен на 2 вида (человеческие и технические активы). К человеческим относятся специалисты предприятия, через которых, есть возможность у злоумышленников нанести вред, а к техническим - аппаратная часть сети компании, такие как устройства использующие сеть IoT, роутер, сервер, ПК и др.

Класс «Угрозы» на данный момент включает группы угроз, содержащихся в датасете KDD CUP 1999, каждая из которых состоит из конкретный атак, такие как satan, portsweep, nmap, ipsweep для Probe.

Смоделируем ситуацию, когда на сеть производится атака portsweep. При portsweep злоумышленник использует различные методы и средства для сканирования портов устройств. Данная атака обычно является своеобразным предшественником другой, скорее всего более значительной атаки. После того, как нейронная сеть определила аномальное соединения и отправила его для анализа в онтологии проверяются параметрические свойства соединения подобно обычной фильтрации. Данная атака не будет подходить к другим классам атак, кроме как Probe, т.к. остальные атаки влияют на другие атрибуты или группы атрибутов.

При угрозах Probe возможны изменения следующих атрибутов: protocol_type; service; flag; count; rerror_rate; srv_rerror_rate; same_srv_rate; diff_srv_rate; srv_diff_host_rate; dst_host_count; dst_host_srv_count; dst_host_same_srv_rate; dst_host_diff_srv_rate; dst_host_same_src_port_rate; dst_host_srv_diff_host_rate; dst_host_serror_rate; dst_host_rerror_rate; dst_host_srv_rerror_rate. Если соединения будет иметь параметры dst_host_rerror_rate и dst_host_srv_rerror_rate значение которых более 0.095 то

такое соединения является подключением Probe и разновидностью атаки portsweep. Остальные атрибуты необходимы для определение остальных разновидностей Probe.

Когда определен конкретный тип атаки его необходимо сопоставить с активами, которые подвержены атаке. Т.к. атака portsweep обычно является показателем возможной разведки сети, то под угрозой могут быть все технические активы. Поэтому пользователю будет рекомендовано проверить актуальные версии безопасности для программного обеспечения на устройствах сети, а также проанализировать открытые и используемые порты устройств и по возможности закрыть неиспользуемые.

На текущий момент в онтологии учтены правила для атак DoS и Probe. Правила для фильтрации взяты из статей [32,33] и представлены в таблицах 5-6 соответственно.

Таблица 5 – Правила для определения разновидностей DoS

Правило	Разновидность
{service, hot}	Back
{flag, diff_srv_rate}	Neptune
{protocol_type, wrong_fragment}	Teardrop
{land, dst_host_srv_diff_host_rate}	Land
{source_byte, wrong_fragment}	Pod
{source_byte, count}	Smurf

Таблица 6 – Правила для определения разновидностей Probe

Правило	Разновидность
If dst_host_same_src_port_rate <=0.005	Satan
If dst_host_count <=71.5 and dst_host_rerror_rate > 0.49	Ipsweep
If dst_host_rerror_rate and dst_host_srv_rerror_rate > 0.095	Portsweep
If dst_host_srv_diff_host_rate > 0.295	Ipsweep
If dst_host_count > 254.5	Satan
If service = private and dst_host_same_src_port_rate > 0.785	Nmap
If 58.5 < dst_host_srv_count <= 201 and srv_diff_host_rate > 0.5	Nmap
If 51 < dst_host_srv_count <= 202.5 and dst_host_same_srv_rate > 0.75	Ipsweep
If 29 < dst_host_srv_count <= 42.5 and srv_diff_host_rate > 0.5	Nmap
If dst_host_same_srv_rate > 0.75 and dst_host_srv_count <= 3.5 and count < 23.5	Ipsweep
If dst_host_srv_count >= 48.5 and dst_host_count < 3.5 and srv_diff_host_rate > 0.5	Nmap
If dst_host_same_srv_rate > 0.75	Ipsweep
If flag = SF and dst_host_diff_srv_rate > 0.035	Satan
If dst_host_rerror_rate <= 0.295	Nmap
If dst_host_rerror_rate > 0.295	Portsweep

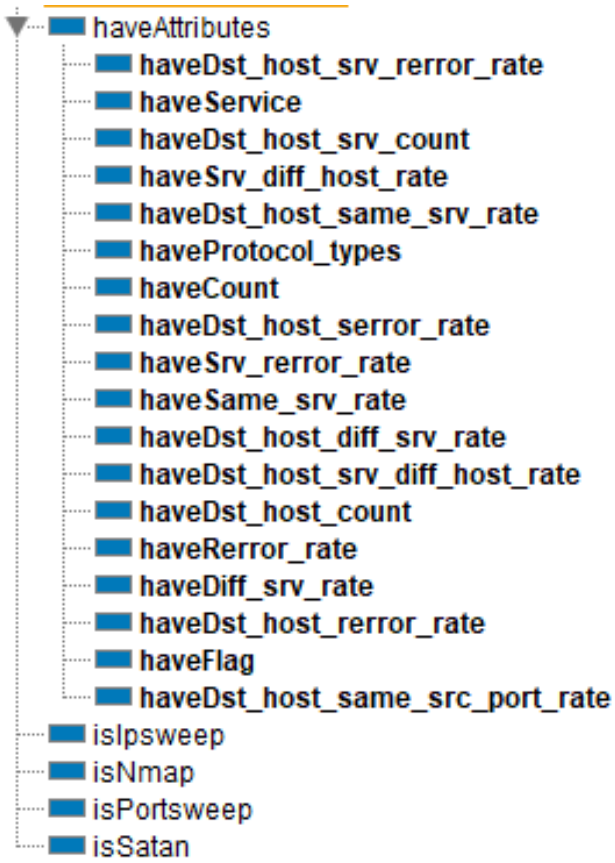

Рисунок 5 – Свойства атак Probe

Таким образом, на данный момент построенная онтология позволяет представить формализованное описание для типов соединений DoS и Probe, представленных в датасете KDD Cup 1999.

ЗАКЛЮЧЕНИЕ

Процесс анализа рисков является не тривиальной задачей. Для успешности данного процесса необходимо учесть множество факторов, поэтому использование одних лишь методических рекомендаций будет недостаточно для получения требуемого результата. Необходимо помимо методических рекомендаций выбрать инструментарий с поддержкой необходимых методов и стандартов и соблюсти регламент управления рисками. Несоблюдение совокупности данных факторов может привести к таким последствиям как:

- Потеря имиджа, которая повлечет снижение доверия инвесторов и клиентов;
- Потере конфиденциальных данных;
- Снижению результативности активным бизнес-процессам компании.

Для минимизации влияния данных факторов создаются системы управления рисками, которые автоматизируют процесс обнаружения атак и предлагают рекомендации для их устранения или снижения ущерба.

В работе реализовалась система обнаружения вторжений и онтология, построенная на основе датасета KDD CUP 1999.

На начальном этапе был произведен анализ датасета и выявлено количество соединений нормальных и аномальных соединений, а также их тип. Полученные результаты представляют следующее: 87832 – соединения типа normal; 54572 – соединения типа атаки-DoS; 2131 – соединения типа атаки-Probe; 999 – соединения типа атаки-R2L; 52 – соединения типа атаки-U2R.

При создании системы обнаружения вторжений производился выбор модели для использования нейронной сети. По результатам проведенных опытов для обнаружения нормальных и аномальных соединений в датасете KDD CUP 1999, как и предполагалось лучшей результат для классификации

данных выдает искусственная нейронная сеть со следующими классификаторами: - достоверность – 0,999346; - полнота – 0,998841;
- ложноположительный – 0,00053; - точность – 0,997855; - F-мера – 0,998348;
- энтропия – 0,002143. Данный метод был выбран как основной для системы обнаружения вторжений.

Так же был предложен один из способов захвата трафика для последующего анализа нейронной сетью.

Определены правила для обнаружения DoS и Probe атак, которые представляют параметрические свойства в онтологии и по сути являются фильтрами. По результатам данной фильтрации пользователю предлагаются рекомендации действий.

Таким образом, в результате работы был реализован модуль программы для обнаружения типа вторжений, а также начато проектирование онтологии знаний.

.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Состояние преступности в России за январь-июнь 2020 года: статистический сборник. М.: ГИАЦ МВД РФ, 2020.
2. Sutherland J., Canwell D. Tangible asset. – 2004.
3. Исакова С. А. Финансовые активы и финансовые обязательства в условиях перехода на МСФО (НСФО 2) //Международный бухгалтерский учет. – 2012. – №. 24.
4. Акимов В. А., Воронов С. П., Радаев Н. Н. Концепции риска и концепции анализа риска //Стратегия гражданской защиты: проблемы и исследования. – 2013. – Т. 3. – №. 2.
5. Диев В. С. Неопределенность как атрибут и фактор принятия решении. – 2010.
6. BS 7799-3:2006 – Information security management systems – Guidelines for information security risk management и ГОСТ Р ИСО/МЭК 27005–2010 Информационная технология. Методы и средства обеспечения безопасности. Менеджмент риска информационной безопасности.
7. Галатенко В.А. Основы информационной безопасности. М., 2004. 264 с.
8. Теоретические основы компьютерной безопасности / П.Н. Девянин [и др.]. М.: Радио и связь. 2001.
9. Муханова А., Ревнивых А. В., Федотов А. М. Классификация угроз и уязвимостей информационной безопасности в корпоративных системах //Вестник Новосибирского государственного университета. Серия: Информационные технологии. – 2013. – Т. 11. – №. 2.
10. FIRST [Electronic resource]. – URL: https://www.first.org/cvss/ (accessed 11.12.2021)
11. BS 7799-3: 2006. Information security management systems – Part 3. Guidelines for information security risk management (Руководство по управлению рисками ИБ)

12. ISO/IEC 27001 — Information security management. [Electronic resource]. – URL: https://www.iso.org/isoiec-27001-information-security.html (accessed 11.12.2021)

13. Alberts C. J., Dorofee A. J. Managing information security risks: the OCTAVE approach. – Addison-Wesley Professional, 2003.

14. Mullerova J., Orincak M. RM/RA CRAMM-Quantitative Risk Assessment Method for Prevention of Criminality //Security Dimensions. ISSN. – 2017. – Т. 23537000. – С. 131-144.

15. Freund J., Jones J. Measuring and managing information risk: a FAIR approach. – Butterworth-Heinemann, 2014.

16. Управление внутренними рисками [Электронный ресурс]. – URL: https://docs.microsoft.com/ru-ru/microsoft-365/compliance/insider-risk-management (доступ 11.02.2022).

17. EUROPIAN UNION AGENCY FOR CUBERSECURITY. [Electronic resource]. – URL: https://www.enisa.europa.eu/topics/threat-risk-management/risk-management/current-risk/risk-management-inventory/rm-ra-tools/t_ra2.html (accessed 11.02.2022).

18. IT Governance vsRisk. [Electronic resource]. – URL: https://store.softline.ru/itgovernance/it-governance-vs-risk/ (accessed 12.06.2021).

19. Искусство управления информационными рисками. Callio Secura 17799 [Electronic resource]. – URL: http://xn----7sbab7afcqes2bn.xn--p1ai/content/callio-secura-17799 (accessed 11.02.2022).

20. РискДетектор. Система автоматизации управления рисками, аудита, мониторинга, контроля фундаментальности безопасности, обеспечения качества критериев и нормативной базы. [Electronic resource]. – URL: http://www.srisks.ru/ (accessed 11.02.2022).

21. Ребро И. В., Шарлаев Е. В. Применение искусственных иммунных систем для обнаружения сетевых вторжений //ИВ Ребро, ЕВ Шарлаев.–Барнаул: Ползуновский альманах–АлтГТУ. – 2015. – С. 144-146.

22. KDD Cup 1999 Data [Electronic resource]. –
URL: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (accessed 12.02.2022)
23. DERIVED FEATURES [Electronic resource]. –
URL: http://kdd.ics.uci.edu/databases/kddcup99/task.html (accessed 12.02.2022)
24. Gasti P. et al. DoS and DDoS in named data networking //2013 22nd International Conference on Computer Communication and Networks (ICCCN). – IEEE, 2013. – С. 1-7.
25. Gu X. et al. Probe Request Based Device Identification Attack and Defense //Sensors. – 2020. – Т. 20. – No 16. – С. 4620.
26. Paliwal S., Gupta R. Denial-of-service, probing & remote to user (R2L) attack detection using genetic algorithm //International Journal of Computer Applications. – 2012. – Т. 60. – №. 19. – С. 57-62.
27. Гайфулина Д. А., Котенко И. В. Применение методов глубокого обучения в задачах кибербезопасности. Часть 2 //Вопросы кибербезопасности. – 2020. – №. 4 (38).
28. Scikit-learn [Electronic resource]. –
URL: https://scikit-learn.ru/ (accessed 11.02.2022)
29. Залетдинов А. В., Локтев А. А. Методы и системы обнаружения сетевых атак //Математика, информатика, естествознание в экономике и в обществе. – 2009. – С. 48.
30. TCPDUMN & LibPCAP [Electronic resource]. –
URL: https://www.tcpdump.org/ (accessed 12.02.2022)
31. Protégé [Electronic resource]. –
URL: https://protege.stanford.edu/ (accessed 12.02.2022)
32. Аведьян Э. Д. Двухуровневая система обнаружения DoS-атак и их компонентов на основе нейронных сетей CMAC //Информационные технологии. – 2016. – Т. 22. – №. 9. – С. 711.

33. Ambedkar C., Babu V. K. Detection of probe attacks using machine learning techniques //International Journal of Research Studies in Computer Science and Engineering (IJRSCSE). – 2015. – Т. 2. – №. 3. – С. 25-29.

ПРИЛОЖЕНИЕ А

Данные по результатам анализа датасета KDD CUP 1999 с дублирующими соединениями:

Таблица А.1 – Количество параметра Type connection

target	number
smurf.	280790
neptune.	107201
normal.	97278
back.	2203
satan.	1589
ipsweep.	1247
portsweep.	1040
warezclient.	1020
teardrop.	979
pod.	264
nmap.	231
guess_passwd.	53
buffer_overflow.	30
land.	21
warezmaster.	20
imap.	12
rootkit.	10
loadmodule.	9
ftp_write.	8
multihop.	7
phf.	4
perl.	3
spy.	2

Таблица А.2 – Количество параметра Type protocol

protocol_type	number
icmp	283602
tcp	190065
udp	20354

Таблица А.3 – Значение параметра Type service

service	number
ecr_i	281400
private	110893
http	64293
smtp	9723
other	7237

Продолжение таблицы А.3 – Количество параметра Type service

service	number
domain_u	5863
ftp_data	4721
eco_i	1642
ftp	798
finger	670
urp_i	538
telnet	513
ntp_u	380
auth	328
pop_3	202
time	157
csnet_ns	126
remote_job	120
gopher	117
imap4	117
discard	116
domain	116
iso_tsap	115
systat	115
shell	112
echo	112
rje	111
whois	110
sql_net	110
printer	109
nntp	108
courier	108
sunrpc	107
netbios_ssn	107
mtp	107
vmnet	106
uucp_path	106
uucp	106
klogin	106
bgp	106
ssh	105
supdup	105
nnsp	105
login	104
hostnames	104
efs	103
daytime	103
link	102

Продолжение таблицы А.3 – Количество параметра Type service

service	number
netbios_ns	102
pop_2	101
ldap	101
netbios_dgm	99
exec	99
http_443	99
kshell	98
name	98
ctf	97
netstat	95
Z39_50	92
IRC	43
urh_i	14
X11	11
tim_i	7
pm_dump	1
tftp_u	1
red_i	1

Таблица А.4 – Количества параметра Type flag

flag	number
SF	378440
S0	87007
REJ	26875
RSTR	903
RSTO	579
SH	107
S1	57
S2	24
RSTOS0	11
S3	10
OTH	8

Таблица А.5 – Количество параметра Land

land	number
0	493999
1	22

Таблица А.6 – Количество параметра Logged in

logged_in	number
0	420784
1	73237

Таблица А.7 – Количество авторизаций Root

root	number
0	493966
1	55

Таблица А.8 – Количество использований sudo

su_attempted	number
0	494009
1	6
2	6

Таблица А.9 - Host login

host_login	number
0	494021

Таблица А.10 - Guest login

guest_login	number
0	493336
1	685

Данные по результатам анализа датасета KDD CUP 1999 без дублирующих соединений:

Таблица А.11 – Количество параметра Type connection

target	number
normal.	87832
neptune.	51820
back.	968
teardrop.	918
satan.	906
warezclient.	893
ipsweep.	651
smurf.	641

Продолжение таблицы А.11 – Количество параметра Type connection

target	number
portsweep.	416
pod.	206
nmap.	158
guess_passwd.	53
buffer_overflow.	30
warezmaster.	20
land.	19
imap.	12
rootkit.	10
loadmodule.	9
ftp_write.	8
multihop.	7
phf.	4
perl.	3
spy.	2

Таблица А.12 – Количество параметра Type protocol

protocol_type	number
tcp	130913
udp	12267
icmp	2406

Таблица А.13 – Значение параметра Type service

service	number
http	62054
private	49057
smtp	9721
domain_u	5425
other	4769
ftp_data	4592
ecr_i	1027
eco_i	916
ftp	798
finger	668
telnet	512
urp_i	443
auth	328
ntp_u	290
pop_3	200
time	139
csnet_ns	126
remote_job	120

Продолжение таблицы А.13 – Количество параметра Type service

service	number
gopher	117
imap4	117
discard	116
iso_tsap	115
systat	115
domain	114
echo	112
rje	111
shell	111
whois	110
sql_net	110
courier	108
printer	108
nntp	108
sunrpc	107
netbios_ssn	107
mtp	107
klogin	106
vmnet	106
uucp_path	106
supdup	105
ssh	105
nnsp	105
uucp	105
bgp	104
hostnames	103
daytime	103
login	103
link	102
netbios_ns	102
efs	101
pop_2	101
ldap	101
http_443	99
exec	98
netbios_dgm	98
kshell	98
name	98
ctf	97
netstat	95
Z39_50	91
IRC	43
urh_i	14
X11	11

Продолжение таблицы А.13 – Количество параметра Type service

service	number
tim_i	5
pm_dump	1
tftp_u	1
red_i	1

Таблица А.14 – Количества параметра Type flag

flag	number
SF	87459
S0	42278
REJ	14712
RSTO	569
RSTR	425
S1	57
SH	34
S2	24
RSTOS0	11
S3	10
OTH	7

Таблица А.15 – Количество параметра Land

land	number
0	145566
1	20

Таблица А.16 – Количество параметра Logged in

logged_in	number
0	74032
1	71554

Таблица А.17 – Количество авторизаций Root

root	number
0	145531
1	55

Таблица А.18 – Количество использований sudo

su_attempted	number
0	145574
1	6
2	6

Таблица А.19 - Host login

host_login	number
0	145586

Таблица А.20 - Guest login

guest_login	number
0	144901
1	685

ПРИЛОЖЕНИЕ Б

Листинг Б.1 – Файл main.py

```
from analysis.analys import *
from training.train import *
import argparse

def parse_args():
    parser = argparse.ArgumentParser(description=__doc__)
    parser.add_argument("mode", help='Выберите режим: "analysis"')
    parser.add_argument("input")
    return parser.parse_args()


def analysis(args):
    print("Загрузка данных из: %s" % args.input)
    df = read(args.input)
    df_target = target_connection(df)
    graphics(df_target, name="with_double")
    print("Формирование файла analysis_with_double.xlsx")
    write_xlsx("./analysis/result/analysis_with_double.xlsx",            "Type
connection", df_target)
    df_protocol = protocol_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",            "Type
protocol", df_protocol)
    df_service = service_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx", "Type service",
df_service)
    df_flag = flag_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",   "Type   flag",
df_flag)
    df_land = land_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",          "Land",
df_land)
    df_logged_in = logged_in_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",     "Logged_in",
df_logged_in)
    df_root = root_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",          "Root",
df_root)
    df_su = su_attemped_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",  "Su_attempted",
df_su)
    df_host = host_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",   "Host  login",
df_host)
    df_guest = guest_connection(df)
    write_xlsx("./analysis/result/analysis_with_double.xlsx",  "Guest  login",
df_guest)
    print("Удаление дупликатов")
    df_non_double = delete_double(df)
    df_non_double_target = target_connection(df_non_double)
    graphics(df_non_double_target, name="without_double")
    print("Формирование файла analysis_without_double.xlsx")
    write_xlsx("./analysis/result/analysis_without_double.xlsx",         "Type
connection", df_non_double_target)
    df_none_double_protocol = protocol_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",         "Type
protocol", df_none_double_protocol)
    df_none_double_service = service_connection(df_non_double)
```

```
    write_xlsx("./analysis/result/analysis_without_double.xlsx",         "Type
service", df_none_double_service)
    df_none_double_flag = flag_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",  "Type  flag",
df_none_double_flag)
    df_none_double_land = land_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",       "Land",
df_none_double_land)
    df_none_double_logged_in = logged_in_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",  "Logged_in",
df_none_double_logged_in)
    df_none_double_root = root_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",       "Root",
df_none_double_root)
    df_none_double_su = su_attemped_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",
"Su_attempted", df_none_double_su)
    df_none_double_host = host_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",         "Host
login", df_none_double_host)
    df_none_double_guest = guest_connection(df_non_double)
    write_xlsx("./analysis/result/analysis_without_double.xlsx",        "Guest
login", df_none_double_guest)
    grarh4(df_non_double)

def train(args):
    model = int(input("Выберите модель (1-ANN, 2-Bayes): "))
    print("Загрузка данных из: %s" % args.input)
    df = read_for_train(args.input)
    if model == 1:
        print("Обучение и тестирование модели ANN")
        result = train_ANN(df)
        write_xlsx("./training/result_models.xlsx", "ANN", result)
    else:
        print("Обучение и тестирование модели Bayes")
        result = train_bayes_model(df)
        write_xlsx("./training/result_models.xlsx", "bayes_model", result)

modes = {
    "analysis": analysis,
    "train": train,
}

if __name__ == '__main__':
    args = parse_args()
    if args.mode not in modes.keys():
        raise Exception("Программа не имеет такого режима. Укажите режим из
следующего списка: %s" % ", ".join(modes.keys()))
    modes[args.mode](args)
```

Листинг Б.2 – Файл analys.py

```
import pandas as pd
import matplotlib.pyplot as plt
import os.path

def read(file):
    file = open("names.txt", "r")
    names = file.readline()
    file.close()
```

```
    df = pd.read_csv(file, names=names)
    return df


def write_xlsx(file, sheet_name, df):
    if os.path.exists(file):
        with pd.ExcelWriter(file, mode="a") as writer:
            df.to_excel(writer, sheet_name=sheet_name, index=False)
    else:
        with pd.ExcelWriter(file, mode="w") as writer:
            df.to_excel(writer, sheet_name=sheet_name, index=False)

def graph1(df, name):
    df = df[:3]
    df.plot(x="target", y="number", kind="bar", legend=None, figsize=(9, 13))
    plt.xlabel("Тип соединения")
    plt.ylabel("Кол-во соединений")
    plt.title("Количество normal соединений и различных типов атак", pad=20)
    plt.grid(True)
    plt.savefig("./analysis/result/graphics/graphs1_%s.png" % (name))


def graph2(df, name):
    df = df[3:12]
    df.plot(x="target", y="number", kind="bar", legend=None, figsize=(9, 13))
    plt.xlabel("Тип соединения")
    plt.ylabel("Кол-во соединений")
    plt.title("Количество различных типов атак")
    plt.grid(True)
    plt.savefig("./analysis/result/graphics/graphs2_%s.png" % (name))


def graph3(df,name):
    df = df[12:]
    df.plot(x="target", y="number", kind="bar", legend=None, figsize=(9, 13))
    plt.xlabel("Тип соединения")
    plt.ylabel("Кол-во соединений")
    plt.title("Количество различных типов атак")
    plt.grid(True)
    plt.savefig("./analysis/result/graphics/graphs3_%s.png" % (name))


def graphics(df, name):
    graph1(df, name)
    graph2(df, name)
    graph3(df, name)


def target_connection(df):
    df = df["target"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"target", df.columns[1]:
"number"})
    return df


def protocol_connection(df):
    df = df["protocol_type"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"protocol_type", df.columns[1]:
"number"})
    return df
```

```
def service_connection(df):
    df = df["service"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"service", df.columns[1]:
"number"})
    return df


def flag_connection(df):
    df = df["flag"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"flag", df.columns[1]: "number"})
    return df


def land_connection(df):
    df = df["land"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"land", df.columns[1]: "number"})
    return df


def logged_in_connection(df):
    df = df["logged_in"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"logged_in", df.columns[1]:
"number"})
    return df


def root_connection(df):
    df = df["root_shell"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"root", df.columns[1]: "number"})
    return df


def su_attemped_connection(df):
    df = df["su_attempted"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"su_attempted", df.columns[1]:
"number"})
    return df


def host_connection(df):
    df = df["is_host_login"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"host_login", df.columns[1]:
"number"})
    return df


def guest_connection(df):
    df = df["is_guest_login"].value_counts().reset_index()
    df = df.rename(columns={ df.columns[0]:"guest_login", df.columns[1]:
"number"})
    return df


def delete_double(df):
    df = df.drop_duplicates(keep="first")
    return df


def n_and_4g(df):
    df = target_connection(df)
```

```
    df_normal = df[df["target"] == "normal."]
    df_neptune = df[df["target"] == "neptune."]
    df_back = df[df["target"] == "back."]
    df_teardrop = df[df["target"] == "teardrop."]
    df_satan = df[df["target"] == "satan."]
    df_warezclient = df[df["target"] == "warezclient."]
    df_ipsweep = df[df["target"] == "ipsweep."]
    df_smurf = df[df["target"] == "smurf."]
    df_portsweep = df[df["target"] == "portsweep."]
    df_pod = df[df["target"] == "pod."]
    df_nmap = df[df["target"] == "nmap."]
    df_guess_passwd = df[df["target"] == "guess_passwd."]
    df_buffer_overflow = df[df["target"] == "buffer_overflow."]
    df_warezmaster = df[df["target"] == "warezmaster."]
    df_land = df[df["target"] == "land."]
    df_imap = df[df["target"] == "imap."]
    df_rootkit = df[df["target"] == "rootkit."]
    df_loadmodule = df[df["target"] == "loadmodule."]
    df_ftp_write = df[df["target"] == "ftp_write."]
    df_multihop = df[df["target"] == "multihop."]
    df_phf = df[df["target"] == "phf."]
    df_perl = df[df["target"] == "perl."]
    df_spy = df[df["target"] == "spy."]
    df_dos = pd.concat([df_back, df_land, df_neptune, df_pod, df_smurf,
df_teardrop])
    df_dos = pd.DataFrame({"target": ["dos."], "number":
[df_dos["number"].sum()] })
    df_probe = pd.concat([df_ipsweep, df_nmap, df_portsweep, df_satan])
    df_probe = pd.DataFrame({"target": ["probe."], "number":
[df_probe["number"].sum()]})
    df_r2l = pd.concat([df_ftp_write, df_guess_passwd, df_imap, df_multihop,
df_phf, df_spy, df_warezclient, df_warezmaster])
    df_r2l = pd.DataFrame({"target": ["r2l."], "number":
[df_r2l["number"].sum()]})
    df_u2r = pd.concat([df_buffer_overflow, df_loadmodule, df_perl,
df_rootkit])
    df_u2r = pd.DataFrame({"target": ["u2r."], "number":
[df_u2r["number"].sum()]})
    df = pd.concat([df_normal, df_dos, df_probe, df_r2l, df_u2r],
ignore_index=True)
    return df


def grarh4(df):
    df = n_and_4g(df)
    write_xlsx("./analysis/result/analysis_without_double.xlsx", "5 groups
connection", "a", df)
    df.plot(x="target", y="number", kind="bar", legend=None, figsize=(9, 9))
    plt.xlabel("Тип соединения")
    plt.ylabel("Кол-во соединений")
    plt.title("Количество normal соединений и различных типов атак")
    plt.grid(True)
    plt.savefig("./analysis/result/graphics/graphs_5_groups_connection.png")
```

Листинг Б.3 – Файл train.py

```
import pandas as pd


from keras.layers import Dense
from keras.models import Sequential

from math import log

from sklearn.compose import ColumnTransformer
from sklearn.model_selection import cross_val_score, train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder, OneHotEncoder, StandardScaler

def read_for_train(file):
    df = pd.read_csv(file)
    return df

def train_ANN(df):
    file = open("attack.txt", "r")
    attack = file.readline()
    file.close()
    df["normal."] = df["normal."].replace(attack, "attack")
    x = df.iloc[:, :-1].values
    y = df.iloc[:, 41].values
    le_x1 = LabelEncoder()
    le_x2 = LabelEncoder()
    le_x3 = LabelEncoder()
    x[:, 1] = le_x1.fit_transform(x[:, 1])
    x[:, 2] = le_x2.fit_transform(x[:, 2])
    x[:, 3] = le_x3.fit_transform(x[:, 3])
    ct_1 = ColumnTransformer(
        transformers=[
            ("OneHot1",
            OneHotEncoder(),
            [1]
            )
        ],
        remainder="passthrough"
    )
    x = ct_1.fit_transform(x)
    ct_2 = ColumnTransformer(
        transformers=[
            ("OneHot2",
            OneHotEncoder(),
            [4]
            )
        ],
        remainder="passthrough"
    )
    x = ct_2.fit_transform(x)
    ct_3 = ColumnTransformer(
        transformers=[
            ("OneHot3",
            OneHotEncoder(),
            [70]
            )
        ],
        remainder="passthrough"
```

```
    )
    x = ct_3.fit_transform(x)
    le_y = LabelEncoder()
    y = le_y.fit_transform(y)
    X_train, X_test, y_train, y_test = train_test_split(x, y, train_size =
0.67, random_state = 0)
    scaler = StandardScaler()
    X_train = scaler.fit_transform(X_train)
    X_test = scaler.transform(X_test)
    model = Sequential([
        Dense(128, activation="relu"),
        Dense(128, activation="relu"),
        Dense(128, activation="relu"),
        Dense(1, activation="sigmoid"),
    ])
    model.compile(loss = "binary_crossentropy", optimizer = "SGD")
    model.fit(X_train, y_train, batch_size = 10, epochs = 20)
    y_pred = model.predict(X_test)
    y_pred = (y_pred > 0.5)
    cm = confusion_matrix(y_test, y_pred)
    print("Формирование результатов...")
    accuracy = (cm[0,0]+cm[1,1])/(cm[0,0]+cm[0,1]+cm[1,0]+cm[1,1])
    recall = cm[1,1]/(cm[0,1]+cm[1,1])
    fp = cm[1,0]/(cm[0,0]+cm[1,0])
    precision = cm[1,1]/(cm[1,0]+cm[1,1])
    F1 = 2*((precision*recall)/(precision+recall))
    entropy = -precision*log(precision)
    df_result = pd.DataFrame(
        {"Классификатор": ["Достоверность", "Полнота", "Ложноположительный",
"Точность", "F-мера", "Энтропия"],
         "Значение": [accuracy, recall, fp, precision, F1, entropy]})
    return df_result

def train_bayes_model(df):
    file = open("attack.txt", "r")
    attack = file.readline()
    file.close()
    df["normal."] = df["normal."].replace(attack, "attack")
    x = df.iloc[:, :-1].values
    y = df.iloc[:, 41].values
    le_x1 = LabelEncoder()
    le_x2 = LabelEncoder()
    le_x3 = LabelEncoder()
    x[:, 1] = le_x1.fit_transform(x[:, 1])
    x[:, 2] = le_x2.fit_transform(x[:, 2])
    x[:, 3] = le_x3.fit_transform(x[:, 3])
    ct_1 = ColumnTransformer(
        transformers=[
            ("OneHot1",
             OneHotEncoder(),
             [1]
             )
        ],
        remainder="passthrough"
    )
    x = ct_1.fit_transform(x)
    ct_2 = ColumnTransformer(
        transformers=[
            ("OneHot2",
             OneHotEncoder(),
             [4]
```

```
            )
        ],
        remainder="passthrough"
    )
    x = ct_2.fit_transform(x)
    ct_3 = ColumnTransformer(
        transformers=[
            ("OneHot3",
             OneHotEncoder(),
             [70]
             )
        ],
        remainder="passthrough"
    )
    x = ct_3.fit_transform(x)
    le_y = LabelEncoder()
    y = le_y.fit_transform(y)
    X_train,    X_test,    y_train,    y_test    =    train_test_split(x,    y,
train_size=0.67, random_state=0)
    scaler = StandardScaler()
    X_train = scaler.fit_transform(X_train)
    X_test = scaler.transform(X_test)
    classifier = GaussianNB()
    classifier.fit(X_train, y_train)
    y_pred = classifier.predict(X_test)
    cm = confusion_matrix(y_test, y_pred)
    accuracies = cross_val_score(estimator=classifier, X=X_train, y=y_train,
cv=10)
    accuracies.mean()
    accuracies.std()
    print("Формирование результатов...")
    accuracy = (cm[0, 0] + cm[1, 1]) / (cm[0, 0] + cm[0, 1] + cm[1, 0] +
cm[1, 1])
    recall = cm[1, 1] / (cm[0, 1] + cm[1, 1])
    fp = cm[1, 0] / (cm[0, 0] + cm[1, 0])
    precision = cm[1, 1] / (cm[1, 0] + cm[1, 1])
    F1 = 2 * ((precision * recall) / (precision + recall))
    entropy = -precision * log(precision)
    df_result = pd.DataFrame(
        {"Классификатор": ["Достоверность", "Полнота", "Ложноположительный",
"Точность", "F-мера", "Энтропия"],
         "Значение": [accuracy, recall, fp, precision, F1, entropy]})
    return df_result
```