На правах рукописи

Салахутдинова Ксения Иркиновна

МЕТОДИКА ИДЕНТИФИКАЦИИ ИСПОЛНЯЕМЫХ ФАЙЛОВ НА ОСНОВЕ СТАТИЧЕСКОГО АНАЛИЗА ХАРАКТЕРИСТИК ДИЗАССЕМБЛИРОВАННОГО КОДА ПРОГРАММ

Специальность 05.13.19 – Методы и системы защиты информации, информационная безопасность

АВТОРЕФЕРАТ

диссертации на соискание ученой степени

кандидата технических наук

Санкт-Петербург – 2019

Работа выполнена в Федеральном государственном бюджетном учреждении науки Санкт-Петербургском институте информатики и автоматизации Российской академии наук (СПИИРАН).

Научный руководитель: **ЛЕБЕДЕВ Илья Сергеевич**,
доктор технических наук, профессор, главный научный сотрудник, руководитель лаборатории интеллектуальных систем Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук

Официальные оппоненты: **БУРЛОВ Вячеслав Георгиевич,**
доктор технических наук, профессор, профессор кафедры информационных технологий и систем безопасности Федерального государственного бюджетного образовательного учреждения высшего образования «Российский государственный гидрометеорологический университет»

ПРИМАКИН Алексей Иванович,
доктор технических наук, профессор, начальник кафедры специальных информационных технологий Федерального государственного казенного образовательного учреждения высшего образования «Санкт-Петербургского университета Министерства внутренних дел Российской Федерации»

Ведущая организация: Федеральное государственное бюджетное образовательное учреждение высшего образования «Государственный университет морского и речного флота имени адмирала С.О. Макарова»

Защита диссертации состоится "18" февраля 2020 г. в __:00 часов на заседании совета по защите диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук Д 002.199.01, созданного на базе Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН) по адресу: 199178, Санкт-Петербург, 14-а линия В.О., 39, комн. 401. Факс: (812)-328-44-50 тел: (812)-328-34-11.

С диссертацией можно ознакомиться в библиотеке Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук по адресу: 199178, Санкт-Петербург, В.О., 14 линия, д. 39 и на сайте http://www.spiiras.nw.ru/dissovet/

Автореферат разослан "____" декабря 2019 г.

Ученый секретарь
диссертационного совета Д 002.199.01,
кандидат технических наук А.А. Зайцева

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность работы. Свободно распространяемое программное обеспечение (ПО) является неотъемлемой частью современных информационных систем, эксплуатируемых в различных секторах экономики, оно позволяет расширить возможности по осуществлению процессов анализа, управления, принятия решений.

Применение открытого ПО обуславливает необходимость разработки дополнительных методов, систем и средств защиты информации. Возможные дефекты программного обеспечения, наличие не декларированных возможностей, нелегальное использование интеллектуальной собственности, применение специальных программ, направленных на преодоление установленной защиты, могут привести к росту числа уязвимостей и повлиять на информационную безопасность систем.

В связи с этим возникает необходимость решения ряда задач идентификации, верификации и валидации программного обеспечения. Предлагаемые решения ориентированы, в основном, на отслеживание фиксированного состояния кода программ на носителях и в оперативной памяти, что не всегда позволяет оперативно определить санкционированные модификации, изменения версий.

Данная работа направлена на решение задачи идентификации программного обеспечения, в условиях изменения версий распространяемого ПО, на основе процедуры построения информативной модели в виде математического кортежа по выбранному признаковому пространству, характеристики которой позволяют найти однозначное соответствие между анализируемой последовательностью и хранящимся эталоном исполняемого файла.

Степень разработанности темы. Различным подходам идентификации программного обеспечения посвятили свои работы такие отечественные ученые как Казарин О. В., Копыльцов А.В., Сорокин И.В., Антонов А. Е., Федулов А. С., зарубежные – Kornblum J.D., Long C., Ebringer T., Sun L., Boztas S., Schultz M. G., Eskin E., Santos I. и другие.

Существующие подходы к идентификации исполняемых файлов, по способу анализа характеристик программы, можно разделить на статические и динамические. К динамическим относятся методы, рассматривающие программу как процесс выполнения некоторого алгоритма или как последовательную смену состояний на графе переходов программы. Статические, в свою очередь, делятся на не сигнатурные (методы проверки целостности): сравнение контрольных сумм, с полной копией данных, вычисление CRC (Cyclic Redundancy Check), хэш-сумм, использование имитовставки, цифровой подписи; и сигнатурные (методы сравнения признаковых последовательностей): выделение «магического числа», сопоставление характерных последовательностей байтов и др.

В качестве информативных признаков файла в отечественных и зарубежных научных работах рассматриваются такие признаки как операторы в тексте программ (Казарин О. В.), энтропия сегментов файлов, цифровые изображения (Копыльцов А.В., Сорокин И.В.), блоки в сегментированной программе (Антонов А.Е., Федулов А.С.), хеш-функции и кусочно-зависимое хеширование (Kornblum J.D., Long C., Guoyin W., Baier H., Frank B.), статистические профили (Ebringer T., Sun L., Boztas S.), бинарные профили, последовательность строк, шестнадцатеричные дампы (Schultz M. G., Eskin E., Zadok F., Stolfo S. J.), *n*-граммы (Santos I., Brezo F., Ugarte-Pedrero X., Bringas P. G.) и многие другие.

Таким образом, существующие подходы к идентификации установленного программного обеспечения обладают рядом ограничений и недостаточными возможностями по обеспечению комплексной реализации мер по информационной безопасности.

Рассматриваемая в работе методика идентификации программного обеспечения направлена на распознавание программы, не основываясь на ее целостности. В результате создается достаточная гибкость, позволяющая определять исполняемые файлы, ранее не задействованные в процессе формирования эталонных сигнатур. Процесс идентификации сравнивает две сигнатуры: эталонную – созданную на основе обучающей выборки и сигнатуру идентифицируемого исполняемого файла – создаваемую непосредственно перед этапом сравнения.

Всевозрастающее количество версий программного обеспечения, обуславливает сложность полноценной комплексной защиты информации. В следствии чего, возникает необходимость в увеличении показателей качества идентификации программ. Противоречие, возникающее между *недостаточной точностью* идентификации ПО существующими методами и *необходимостью* по обеспечению достаточного уровня защищенности информации в информационной системе, обуславливает **актуальность данного исследования**.

Целью работы является увеличение точности идентификации установленного программного обеспечения за счет разработки и обоснования научно-методического аппарата по идентификации исполняемых файлов, устанавливаемых на средства вычислительной техники, обеспечивающего увеличение точности идентификации в условиях наличия различных версий, большого числа различных наименований программ и ограниченности числа объектов обучающей выборки.

Для достижения поставленной цели были решены следующие **задачи**:

1. Исследование и анализ существующих подходов и методов идентификации программного обеспечения, используемых отечественными и зарубежными исследователями.
2. Разработка модели нарушителя информационной безопасности организации.
3. Разработка метода выделения признакового пространства исполняемых файлов с последующим созданием сигнатур на его основе.
4. Разработка метода сравнения сигнатуры идентифицируемого исполняемого файла с эталонными сигнатурами программ, обеспечивающего увеличение точности идентификации.
5. Разработка методики идентификации исполняемых файлов на основе созданного архива эталонных сигнатур программ с использованием статистического подхода и машинного обучения.
6. Проведение вычислительного эксперимента и обоснование применимости разработанной методики идентификации программного обеспечения.

В соответствии с заявленными целями и задачами работы: **объектом исследования** является разнообразие версий программного обеспечения, а **предметом исследования** методы идентификации программного обеспечения на основе статического анализа характеристик дизассемблированного кода программ.

В результате проведенных исследований были получены следующие результаты, составляющие **научную новизну** диссертации:

1. Метод формирования сигнатур исполняемых файлов, основанный на построении частотного распределения каждой из градаций выделенной

характеристики исполняемых файлов, отличающийся от существующих использованием ряда отобранных наиболее информативных и устойчивых в проявлении ассемблерных команд.

2. Метод сравнения сигнатур идентифицируемых исполняемых файлов с ранее сформированными эталонными сигнатурами программ, отличающийся от известных применением комбинированного подхода использования алгоритма машинного обучения и аддитивного критерия, способствующего снижению числа ошибочных результатов классификации и обеспечивающего увеличение точности от совокупного использования признакового пространства, а также учитывающий ряд изменений в коде исполняемых файлов, что позволяет идентифицировать не рассматриваемые на этапе обучения версии программ.

3. Разработанная методика идентификации ПО, основанная на комбинированном анализе характеристик дизассемблированного кода программ, отличающаяся от известных, применением уникального сформированного признакового пространства и теории полезности для принятия решения на основе аддитивного критерия, что позволяет распознавать версии программ, ранее не задействованных в создании эталонных сигнатур исполняемых файлов

Теоретическую и практическую значимость работы составляют разработанные методы и методика по идентификации программного обеспечения, позволяющие обнаруживать нарушения информационной безопасности при обработке конфиденциальной информации, вызванные несанкционированной установкой программ. Проведенные вычислительные эксперименты подтверждают результативность предложенных решений на реальных данных.

Методология и методы исследования поставленных задач включали теорию информационной безопасности, методы математической статистики, теорию предпочтений, методы машинного обучения, экспериментальные методы исследования.

На защиту выносятся следующие положения:

1. Метод формирования эталонных сигнатур программ и сигнатур идентифицируемых исполняемых файлов, основанный на статическом подходе анализа характеристик дизассемблированных кодов программ, обеспечивающий создание уникальных по форме и амплитуде частотных распределений, построенных на использовании ряда отобранных наиболее информативных ассемблерных команд, устойчиво проявляющихся в различных программах.

2. Метод сравнения сигнатур идентифицируемых исполняемых файлов с ранее сформированными эталонными сигнатурами программ при помощи комбинированного подхода использования алгоритма машинного обучения – градиентного бустинга деревьев решений и аддитивного критерия Фишберна, позволяющий достигать наименьшего числа ошибок неверной классификации и максимизировать эмерджентность совокупности признакового пространства.

3. Методика идентификации исполняемых файлов, включающая разработанные методы по формированию и сравнению сигнатур идентифицируемых исполняемых файлов с эталонными сигнатурами программ из архива эталонных сигнатур программ, обеспечивающая увеличение точности идентификации при комбинированном анализе характеристик из их дизассемблированного представления.

Обоснованность и достоверность полученных результатов подтверждается использованием апробированного математического аппарата; проведением сравнительного анализа с существующими методами; серией практических экспериментов по идентификации исполняемых файлов; проверкой адекватности положений и выводов; согласованностью результатов, полученных при

теоретическом исследовании с результатами проведенных экспериментов; практической апробацией результатов исследования в докладах и публикациях на отечественных и зарубежных научных конференциях.

Результаты практической **апробации работы** подтверждают адекватность и корректность разработанных методов. Основные результаты работы были представлены на следующих конференциях: 18th, 20th Conference of Open Innovations Association FRUCT and ISPIT 2017 seminar, 2016, 2017гг.; IV, VI, VII, VIII Всероссийский конгресс молодых ученых, 2015, 2017, 2018, 2019гг.; 11th IEEE International Conference on Application of Information and Communication Technologies, AICT 2017, 2017г.; IX, X Санкт-Петербургская межрегиональная конференция «Информационная безопасность регионов России (ИБРР-2017)», 2015, 2017гг.; Региональная информатика "РИ-2016", 2016г.; XLVII, XLVIII Научная и учебно-методическая конференция Университета ИТМО, 2018, 2019гг.; International Conference on Next Generation Wired/Wireless Networking Conference on Internet of Things and Smart Spaces NEW2AN 2018, ruSMART, 2018г.; 28-я научно-техническая конференция. Методы и технические средства обеспечения безопасности информации, МиТСОБИ, 2019г.

Результаты, полученные в диссертации, были реализованы в рамках выполнения следующих НИР: Проект по программе Президиума РАН № 0073-2018-0008 «Теория и распределенные алгоритмы самоорганизации группового поведения агентов в автономной миссии», 2018,2019гг.; Проект по программе Президиума РАН № 0073-2018-0007 «Разработка масштабируемых устойчивых алгоритмов построения семантических моделей больших данных и их использование для решения прикладных задач кластеризации и машинного обучения», 2018,2019гг.; НИР-ФУНД «Разработка методов интеллектуального управления киберфизическими системами с использованием квантовых технологий» №617026 (2017-2018гг.); НИР-ФУНД «Разработка методов создания и внедрения киберфизических систем» № 619296 (2018-2019гг.). Результаты работы использовались при разработке системы мониторинга состояния внутренних сетей компании АО «НПК «ТРИСТАН». Полученные результаты используются в образовательном процессе факультета БИТ Университета ИТМО по направлениям подготовки бакалавриата 10.03.01 и магистратуры 10.04.01 по дисциплинам «Организация и управление службой защиты информации», «Теория вероятностей», «Методы цифровой обработки видеоизображений», «Управление информационной безопасностью».

Публикации. По результатам диссертационного исследования автором опубликовано 32 работы, из них статей в журналах, рекомендованных ВАК РФ – 8 входящих в базы цитирования Web of Science и Scopus – 8, свидетельств о государственной регистрации программы для ЭВМ – 6, в прочих изданиях – 10.

Личный вклад автора. Результаты диссертационной работы получены автором самостоятельно. Автором проведен анализ существующих методов идентификации программного обеспечения. Проанализированы условия и ограничения применения каждого из методов.

Структура и объем диссертации. Диссертационная работа содержит введение, 4 раздела, заключение, список литературы и 6 приложений. Объем работы составляет 163 страницы. Работа включает 26 рисунков, 17 таблиц. Список литературы содержит 101 наименование.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении обоснован выбор темы настоящей работы и ее актуальность; представлена степень разработанности темы; определены объект, предмет и цель исследования; описаны частные задачи, обоснована теоретическая и практическая значимость получаемых результатов; раскрыты принципы используемых подходов и разработанной методики; сформулированы положения, выносимые на защиту и приведена апробация результатов исследования.

В первой главе представлен анализ существующей проблематики современного подхода к идентификации программного обеспечения. На основе выделения объекта защиты, цели реализации угроз, потенциальных уязвимостей и видов ущерба разработана модель угроз и нарушителя информационной безопасности при обработке информации конфиденциального характера в информационной системе.

Во второй главе произведен обзор современных решений в области идентификации исполняемых файлов, представленных в отечественной и зарубежной литературе. Проанализированы различные подходы к сбору характеристик файлов и методы сравнения наборов данных характеристик для нескольких файлов. Выявлены достоинства и недостатки данных методов.

Исследованы способы сбора характеристик файлов по статическим, динамическим характеристикам рассматриваемого объекта и сбора метаданных об исследуемом файле, как при помощи встроенных функций операционной системы, так и функциональности внедряемого программного агента. В частности, использующие в качестве идентификационных признаков: операторы в тексте программ, энтропию сегментов файлов, цифровые изображения, блоки в сегментированной программе, хеш-функции и кусочно-зависимое хеширование, статистические профили, бинарные профили, последовательность строк, шестнадцатеричные дампы, байтовые n-граммы, последовательность ассемблерных команд, строковые константы, вектор статистических описаний файла при движении скользящего окна; отслеживание необычного поведения программы в среде выполнения, способов обработки данных, попыток получения высоких привилегий; инструментарий WMI, Active Directory Domain Services, утилит Linux ОС и другие.

Также исследованы методы распознавания программ на основе метрик схожести, машинного обучения, статистического и эвристического анализов.

Существующие подходы к идентификации установленного программного обеспечения обладают рядом недостатков и ограничений, обеспечивающих недостаточную точность идентификации исполняемых файлов, наиболее весомые из них: отсутствие способности к идентификации ранее не исследуемого файла; задействование огромного числа ресурсов; исследование подозрительного поведения программы, наличие которого присуще в основном только вредоносным программам; получение информации из автоматизированной системы, достоверность которой находится под вопросом, учитывая возможные способы ее подмены, рассмотренные в настоящей работе. Применительно к настоящему диссертационному исследованию многие методы основаны на использовании особенностей PE формата программ ОС Windows, отсутствующие в ELF-файлах или бинарной классификации и мульти-классификации с небольшим числом классов.

Проведенный анализ методов идентификации различных исполняемых файлов, позволяет сделать вывод, что идентификация на основе статического анализа характеристик программы и использование статистических и машинных

алгоритмов классификации, является наиболее перспективным направлением деятельности в рамках поставленной научной задачи исследования.

Таким образом, на основании проведенного анализа литературы в задаче выявления факта нарушения пользователем установленных правил по запрету на несанкционированную установку программного обеспечения, можно утверждать, что в современных условиях, не существует решения, позволяющего с максимальной точностью производить идентификацию различных версий программного обеспечения в автоматизированной системе. Отсюда вытекает необходимость в проведении настоящего исследования, направленного на увеличение точности идентификации исполняемых файлов, путем совершенствования существующих подходов, поиска новых, ранее не применявшихся, решений в области выделения признакового пространства файлов, методов идентификации, а также способов постобработки результатов.

Сформулирована постановка задачи обеспечения безопасности автоматизированной системы от потенциальных угроз со стороны несанкционированно установленного программного обеспечения, где определено, что $P = \{P_{\text{этал.},1}, P_{\text{этал.},2}, \ldots, P_{\text{этал.},i}\}$ – набор потенциальных эталонных программ (обучающая выборка) их имен $N = \{N_1, N_2, \ldots, N_i\}$ для формирования архива эталонных сигнатур программ, где i – число программ разных наименований. Множество версий для программ представлено как $P_{\text{этал.},i} = \{v_1, v_2, \ldots, v_m\}$, где m – число версий программы с одним наименованием, которые участвуют в формировании сигнатур $S(P_{\text{этал.},i})$. В свою очередь для формирования сигнатуры версии программы используется функция $q(v_m,F)$: $\mathrm{X} \rightarrow \mathrm{N}_0$, которая при помощи заданного алгоритма и выбранной характеристики F формирует частотную последовательность признака версии программы, где X – множество принимаемых значений характеристики F. Результаты функции формируют наборы эталонных сигнатур программ $S(P_{\text{этал.},i}) = \{(N_i, q(v_1,F)), (N_i, q(v_2,F)), \ldots, (N_i, q(v_m,F))\}$, где N_i – имя программы $P_{\text{этал.},i}$, которые заносятся в архив: $A = \{S(P_{\text{этал.},1}), S(P_{\text{этал.},2}), \ldots, S(P_{\text{этал.},i})\}$.

При проведении аудита имеется множество неизвестных исполняемых файлов $E = \{e_1, e_2, \ldots, e_j\}$ (тестовая выборка), где j – число исследуемых исполняемых файлов. С помощью той же выбранной характеристики F и функции q производится формирование частотной последовательности признака для каждого исследуемого файла e_j. Сигнатура идентифицируемого исполняемого файла $S(e_j)$ представляет собой результат функции $q(e_j,F)$: $\mathrm{X} \rightarrow \mathrm{N}_0$.

Требовалось построить алгоритм $I(S(P_{\text{этал.},i}), S(e_j))$: $e_j \rightarrow N_i$ способный определить меру сходства идентифицируемого исполняемого файла e_j с именем N_i эталонной программы $P_{\text{этал.},i}$, который бы удовлетворял следующему критерию: *точность* (метрика оценки результатов идентификации: *Accuracy* или *F-measure*) должна быть *максимальной*, при условии ограничений на: потенциальное количество различных программ, установленных на одном компьютере или на всех компьютерах организации; количество различных версий для каждой программы, измеряемое от одной до десятков, но в практике не превосходящее сотни; наличие существенных изменений в различных версиях одной программы; наличие индивидуального характера распределения признака идентифицируемого файла от других программ; возможность производить дизассемблирование исполняемого файла; не способность в выявлении закладки вредоносного кода в легитимный исполняемый файл; не способность классификации выдать корректный результат для программы, класс которой не был определен на этапе формирования модели классификации (нельзя создать класс «файл не похож ни на одну из программ»).

В третьей главе произведены анализ структуры и характеристик ELF-файла, его дизассемблирование и представление исходного кода на низкоуровневом языке ассемблера. Показано, как особенности представления программы в различных видах (бинарном, ассемблерном, на уровне инструкций высокоуровневого языка программирования, структуры формата файла и других) влияют на выбор признака и его потенциал для построения информативной модели программы в виде математического кортежа (сигнатуры).

Приведены описания исследуемых исполняемых файлов, их разрядности, разделение на обучающую и тестовую выборку. С помощью закона Бенфорда обосновано разнообразие собранных программ и их версий, подчиняющееся естественному распределению файлов различных объемов на персональном компьютере.

Описаны подход к представлению программного обеспечения и модель представления исполняемого файла в настоящем исследовании. Произведено выделение признакового пространства для объекта o, представление которого в n-мерном пространстве будет отображаться точкой с координатами $o = (f_1(o), f_2(o), \ldots, f_n(o))$, а каждый класс объектов, т.е. имя программы $P_{\text{этал.},i}$ множеством таких точек. И описано его дальнейшее использование в различных методах формирования сигнатур.

Описаны подход и необходимость в выборе определенных ассемблерных команд, рассчитана информативность ряда ассемблерных команд. Представлено подробное описание разработанных методов формирования сигнатур – эталонных и идентифицируемых. В работе рассматриваются несколько подходов к созданию информативной модели программы: на основе 118 ассемблерных команд; на основе одной ассемблерной команды на неравных интервалах; на основе одной ассемблерной команды на равных интервалах; а также унификация сигнатур версий программ при формировании эталонных сигнатур программ. Различные подходы к формированию сигнатур вызваны особенностями использования разработанных методов идентификации.

Первый выносимый на защиту результат представляет собой формирование эталонных сигнатур программ на основе распределения одной ассемблерной команды на равных интервалах:

В качестве характеристики F файла представлен набор выделенного числа ассемблерных команд и $F = (f_1, f_2, \ldots, f_n) = (ac_1, ac_2, \ldots, ac_n)$, ac_n – ассемблерная команда $ac_n \in As.com$.

Процесс создания сигнатур для каждой рассматриваемой программы в отдельности представлен следующими этапами:

1. Каждый файл v_m дизассемблируется и разбивается на интервалы равных длин;

2. Из полученного представления кода выделяется частотная характеристика выбранного признака ac_n (одной ассемблерной команды), являющаяся основой для формирования сигнатуры одного исполняемого файла $S(v_m) = (L(ac_{n,1}), L(ac_{n,2}), \ldots, L(ac_{n,k}))$, где $ac_{n,k}$ – значение частоты появления признака ac_n в k-ом интервале, k – число интервалов разбиения дизассемблированного кода программы $P_{этал,i}$, где значение признака $L(ac_n) \in \mathbb{N}_0$ и показывает количественное значение для признака ac_n.

Данный вид сигнатуры используется при идентификации с помощью градиентного бустинга деревьев решений.

3. На основании сигнатур версий v_m программы $P_{этал,i}$ строится эталонная сигнатура программы (ЭСП):

$$S(P_{этал,i}) = (\{N_i, L(ac_{n,1}), L(ac_{n,2}),\ldots, L(ac_{n,k})\},\ldots,\{N_i, L(ac_{n,1}), L(ac_{n,2}),\ldots, L(ac_{n,k})\})$$

где N_i – наименование программы.

4. Сигнатура идентифицируемого исполняемого файла (СИИФ), в соответствии с расчетами под пунктом 2) имеет вид:

$$S(e_j) = (L(ac_{n,1}), L(ac_{n,2}),\ldots, L(ac_{n,k})).$$

Сформулированы различные методы сравнения сигнатур, основанные как на статистическом подходе (критерий однородности хи-квадрат, критерий согласия Колмогорова), так и на машинном обучении (искусственная нейронная сеть, градиентный бустинг деревьев решений), работоспособные в ограничивающих условиях наличия большого числа наименований программ (классов), отсутствия класса вредоносных программ как такового, ограниченного набора проверяемых программ при проведении аудита, постоянного выпуска новых версий программного обеспечения разработчиками. Представленный метод сравнения сигнатур, показывающий наилучшие результаты идентификации и включающий этап постобработки результатов сравнения сигнатур для повышения достигаемой точности идентификации исполняемых файлов, представляет собой второй, выносимый на защиту результат.

Описана методика идентификации исполняемых файлов на основе статического анализа характеристик дизассемблированного кода программ (рисунок 1).

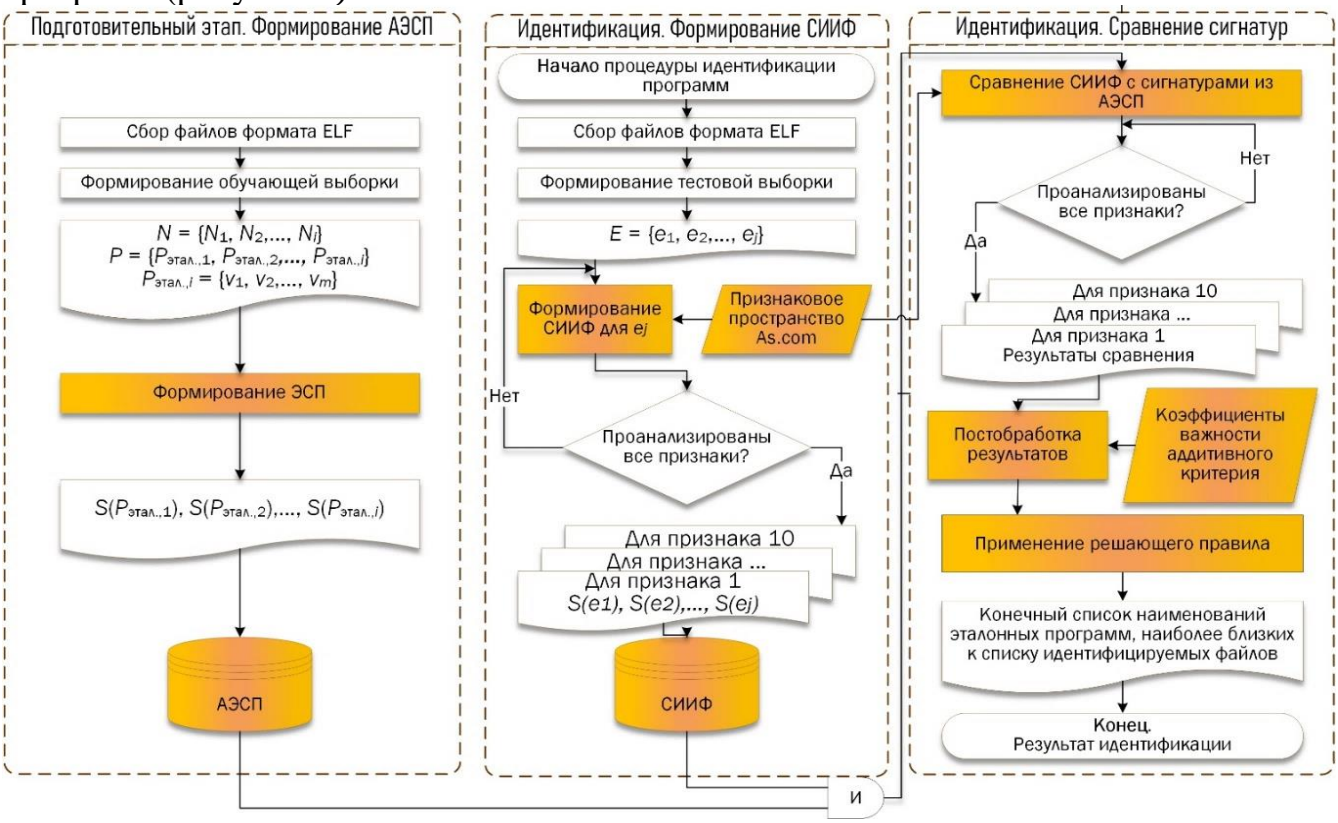


Рисунок 1 – Блок-схема методики идентификации исполняемых файлов

Заключительным результатом, выносимым на защиту, является методика, которая включает в себя обязательный подготовительный этап, на котором производится формирование архива эталонных сигнатур программ (АЭСП). И этап идентификации исполняемого файла, состоящий из формирования сигнатур идентифицируемых исполняемых файлов (СИИФ), их непосредственного сравнения с ЭСП из АЭСП, постобработку с применением аддитивного критерия и принятие итогового решения о подлинности предъявленного идентификатора. В

качестве решающего правила выступает выбор наибольшей вероятности принадлежности сигнатуры идентифицируемого исполняемого файла к эталонной сигнатуре программы из архива, по десяти ассемблерным командам:

$$Name(e_j) = \max\{p_{e_j}(N_i) = \sum_{n=1}^{10} \lambda_n \cdot p_{e_j}^{ac_n}(N_i)\}.$$

Приводится три способа оценки точности идентификации: матрица ошибок (*Confusion Matrix*), точность (*Accuracy*), бикубическая мера (*F-measure*), позволяющие производить оценку результатов, полученных с применением статистических критериев, так и алгоритмов машинного обучения, а также их сравнение с другими исследованиями.

Представлены ограничения, накладываемые на методику и условия ее использования.

В четвертой главе с целью проверки достигаемой точности идентификации исполняемых файлов при помощи разработанной методики, а также для выделения наиболее результативных методов по формированию и сравнению сигнатур ELF-файлов, была проведена серия экспериментов, направленных на каждый разработанный метод сравнения сигнатур файлов в отдельности. Ниже графически изображены подходы к сравнению нескольких сигнатур (распределений частот признака) при помощи статистических критериев (рисунки 2-4) и подходы к классификации на основе алгоритмов машинного обучения (рисунок 5).

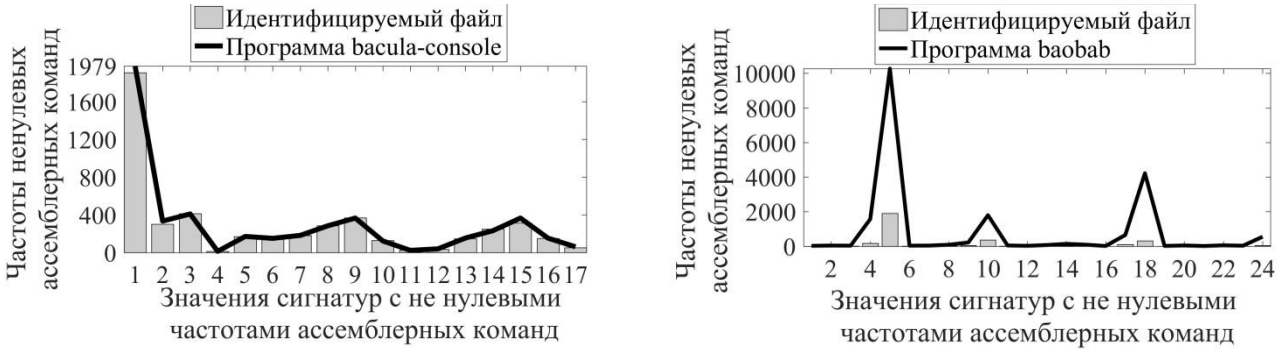
– Подход к построению унифицированных ЭСП и СИИФ на основе распределения 118 ассемблерных команд и идентификации при помощи критерия однородности хи-квадрат (рисунок 2). Из рисунка видно, что для версий одной и той же программы, гистограмма сигнатуры идентифицируемого файла и кривая эталонной сигнатуры программы отличаются не значительно;

– Подход к построению унифицированных ЭСП и СИИФ на основе распределения одной ассемблерной команды (в примере использованы 10 команд) на равных интервалах и идентификации при помощи критерия однородности хи-квадрат (рисунок 3). Из рисунка видно, что для версий одной и той же программы, разница сигнатур по десяти ассемблерным командам минимальна;

– Подход к построению унифицированных ЭСП и СИИФ на основе распределения одной ассемблерной команды (в примере использована команда *cmp*) на неравных интервалах и идентификации при помощи критерия согласия Колмогорова (рисунок 4). Из рисунка видно, что для версий одной и той же программы, гистограмма сигнатуры идентифицируемого файла и кривая эталонной сигнатуры программы отличаются не значительно;

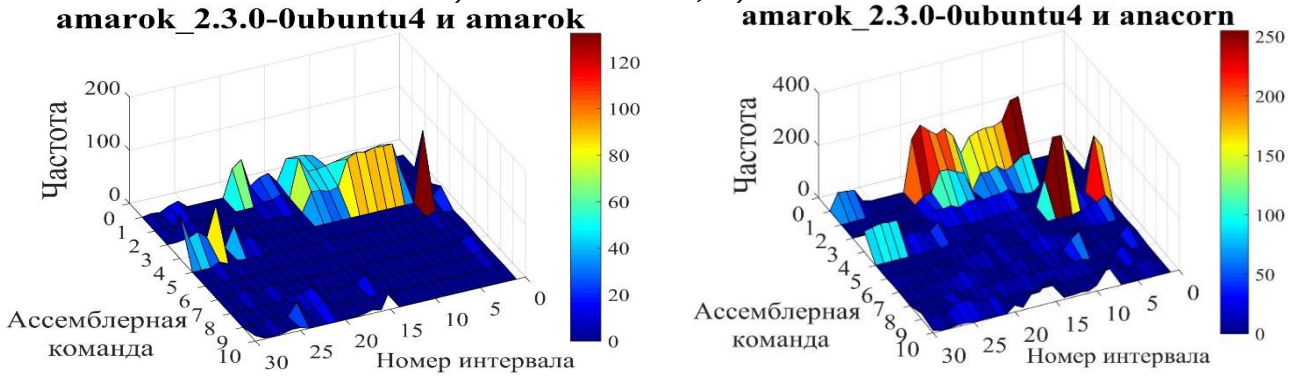
– Подход к построению унифицированных ЭСП и СИИФ на основе распределения одной ассемблерной команды (в примере использована команда *jmp*) на равных интервалах и идентификации при помощи искусственной нейронной сети (рисунок 5.а). Из рисунка видно, что для версий одной и той же программы, при верной классификации, метка на координатной плоскости располагается на диагонали квадрата;

– Подход к построению ЭСП и СИИФ на основе распределения одной ассемблерной команды (в примере использована команда *je*) на равных интервалах и идентификации при помощи градиентного бустинга деревьев решений (рисунок 5.б). Из рисунка видно, что для версий одной и той же программы, при верной классификации, метка на координатной плоскости располагается на диагонали квадрата.



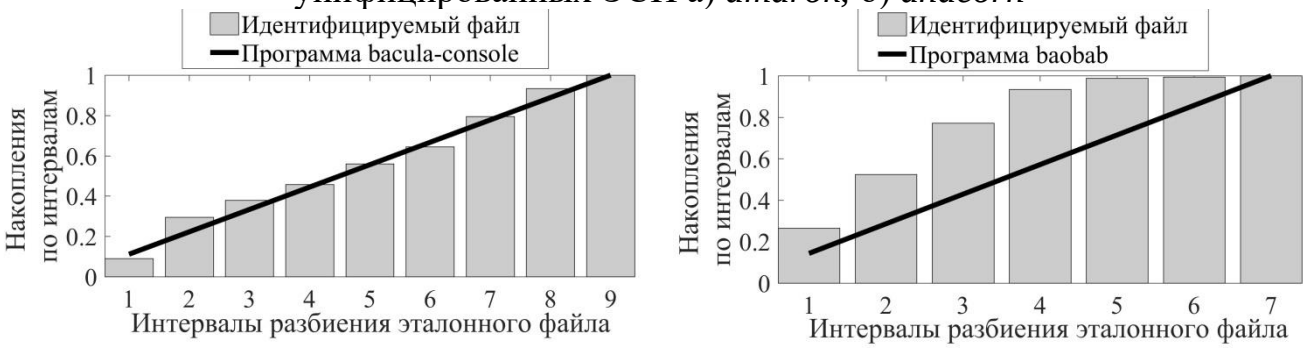
а) б)

Рисунок 2 – Распределение СИИФ *bacula-console_5.2.5-0ubuntu6* и ЭСП а) *bacula-consol*; б) *baobab*



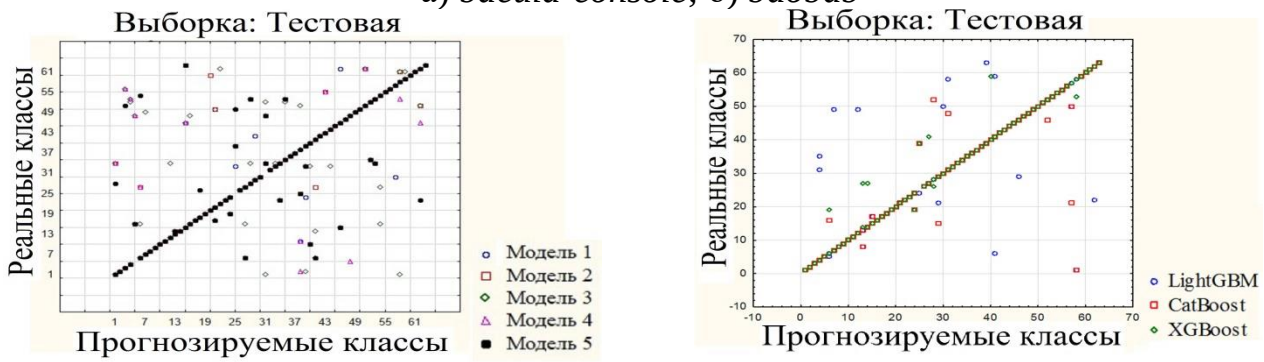
а) б)

Рисунок 3 – Абсолютная разница между СИИФ *amarok_2.3.0-0ubuntu4* и унифицированных ЭСП а) *amarok*; б) *anacorn*



а) б)

Рисунок 4 – Распределение СИИФ *bacula-console_5.2.5-0ubuntu6* и ЭСП а) *bacula-console*; б) *baobab*



а) б)

Рисунок 5 – График зависимости результатов идентификации тестовой выборки а) искусственной нейронной сети идентификации (для пяти моделей обучения); б) градиентного бустинга деревьев решений от истинной принадлежности файлов классам (для трех библиотек)

Таблица 1 – Показатель *Accuracy* по каждой ассемблерной команде, в процентах и *F-measure*, (в скобках)

Ассемблерные команды	Статистические критерии, $p = 0{,}05$			Машинное обучение			
					Градиентный бустинг деревьев решений		
	однородности хи-квадрат	согласия Колмогорова	однородности хи-квадрат	Нейронная сеть MLP	LightGBM	CatBoost	XGBoost
add	79,55	-	40,65	76,42	82,92	85,37	92,68
and		-	60,16	69,11	78,87	86,18	91,87
call		-	65,85	74,80	81,3	84,55	93,50
cmp		50,41	70,73	*82,11*	78,04	85,37	84,55
je		-	69,11	78,05	*86,99* (0,84)	*89,43* (0,88)	91,05
jmp		-	65,85	73,98	83,74	83,74	*95,93* (0,96)
lea		-	*78,05*	56,10	73,99	76,42	91,06
mov		-	47,97	74,80	73,99	85,37	94,30
pop		-	60,16	69,11	78,86	83,74	88,60
push		-	56,91	53,66	74,79	82,93	89,40
Фишберн	-	-	-	84,93	87,81	91,87	*99,19* (0,99)

Таблица 2 – Наилучшие результаты других исследований

Признаковое пространство и метод идентификации	Число классов	Оценка качества метода	
		Accuracy, %	F-measure
Печатаемые строки + Naive Bayes	Бинарная классификация	97,11	-
Набор шестнадцатеричных чисел + Voting Naive Bayes		96,88	-
Последовательность частоты встречаемости очередности ($n = 2$) ассемблерных команд + SVM: normalised polynomial		95,90	-
Последовательность частоты встречаемости очередности ($n = 1$) ассемблерных команд + SVM: Pearson VII		92,92	-
Последовательность частоты встречаемости очередности вызовов API ($n = 3$)+ Мера Жаккара и кластеризация	Кластеризация	-	0,81
Последовательность вызовов API + Сравнение строк программными средствами (diff, CCFinder) и кластеризация		-	0,78
Вектора для блоков постоянного размера побайтового кода программы + Редакционное расстояние	Мульти-классификация	-	0,69

Результаты описанных методов идентификации представляются в одном из двух видов: матрица попарного сравнения сигнатур тестовой выборки с сигнатурами обучающей выборки со значениями статистики на пересечении строк и столбцов данной матрицы; и последовательность предсказанных классов (наименований программ) для файлов тестовой выборки. Первый вид описывает результаты применения статистических критериев при идентификации, второй – результаты машинного обучения.

Описан этап постобработки получаемых результатов, также составляющий второй результат, выносимый на защиту, по повышению точности идентификации путем использования аддитивного критерия Фишберна к совокупности результатов по десяти ассемблерным командам и следующей аддитивной функции:

$$\varphi(\text{jmp},\ldots,\text{cmp})=0{,}182\cdot\text{jmp}+0{,}164\cdot\text{mov}+0{,}145\cdot\text{call}+0{,}118\cdot\text{add}+0{,}118\cdot\text{and}+$$

$$+0{,}082\cdot\text{je}+0{,}082\cdot\text{lea}+0{,}055\cdot\text{push}+0{,}036\cdot\text{pop}+0{,}018\cdot\text{cmp}$$

применение которой к результатам классификации XGBoost позволяет достигать точности идентификации *Accuracy* равным 99,19%, а в измерении бикубической меры *F-measure* достигается значение 0,99. Полученные результаты (таблица 1) прошли сравнение с результатами других исследователей (таблица 2), на данном основании сделан вывод о том, что в условиях наличия множества классов, и ограниченности объема обучающей выборки, разработанная методика обеспечивает более высокую точность идентификации.

На практике методика идентификации программного обеспечения может быть применена для повышения безопасности информационной системы организации, путем внедрения ее в качестве технической меры по аудиту программного обеспечения на электронных носителях информации, она позволяет осуществлять автоматизированную идентификацию ELF-файлов в соответствии с формируемыми архивами легитимных или нелегитимных программ. Выделен ряд смежных задач, решаемых разработанной методикой.

Стоит отметить, что методика эффективно показывает себя на ограниченном наборе данных и способна идентифицировать исполняемый файл, не задействованный при создании архива сигнатур, имеющий новую версию или внесенные в него изменения, а также исправленные, или отсутствующие вообще, метаданные.

Таким образом, методика может быть применена для выявления факта нарушения установленных организационных мер о запрете на установку определенных программ, а также решать ряд смежных задач.

Заключение. В диссертационной работе решена научная задача увеличения точности идентификации исполняемых файлов, устанавливаемых на средства вычислительной техники, в условиях наличия различных версий ПО, большого числа различных наименований программ и ограниченности числа объектов обучающей выборки, обусловленных реальным состоянием выпускаемых версий того или иного ПО. Использование разработанной методики позволяет увеличить точность идентификации исполняемых файлов, что предоставляет возможность производить более тщательный анализ электронных носителей информации на предмет наличия на них нелегитимно установленного ПО. Методика реализуется как часть технических мер по обеспечению ИБ и предотвращению потенциального возникновения новых уязвимостей ИС, способных повлечь за собой негативные воздействия на защищаемую информацию и персонал АС. В том числе получены следующие **научные результаты**, составляющие **итоги** исследования:

1) Разработан метод представления идентификатора исполняемого файла в виде сигнатуры, позволяющий реализовывать гибкий подход к идентификации версий ПО.

2) Разработан метод сравнения сигнатур программ на основе использования библиотеки градиентного бустинга деревьев решений с последующей постобработкой результатов классификации, на основе аддитивного критерия.

3) Разработана методика идентификации исполняемых файлов, позволяющая повысить точность идентификации в условиях большого числа классов и малого числа элементов обучающей выборки, соответствующих реальным условиям эксплуатации ПО в организации. Достигается значение *Accuracy* в 99,19%, что превышает бинарный подход классификации на основе использования UNIX команды strings, показывающая печатаемые строки в объекте или двоичном файле, и сравнения получаемых сигнатур при помощи наивного Байесовского классификатора. Достигаемые результаты также превышают измеренный показатель *F-measure* в среднем на 18% чем у мульти-классификационного подхода на основе использования вектора для блоков постоянного размера побайтового кода программы и сравнения получаемых сигнатур при помощи редакционного расстояния и подхода, основанного на кластеризации последовательностей выводов API.

Рекомендации по применению результатов работы для идентификации исполняемых файлов в автоматизированных системах включают в себя указания по формированию архива сигнатур; применению метода и методики идентификации исполняемых файлов, позволяющих обнаруживать нарушения установленных мер политики безопасности в плане запрета на несанкционированную установку программного обеспечения. Также рекомендации включают разработку подходов, направленных на повышение эффективности процесса аудита электронных носителей информации и его автоматизации, позволяющие идентифицировать исполняемые файлы, не основываясь на методах оценки их целостности и анализе метаданных встроенными средствами операционных систем.

В качестве **перспектив дальнейшей разработки тематики** можно выделить исследования, связанные с разработкой динамического подхода к идентификации программного обеспечения в близких к системе реального времени ограничениях на скорость реагирования при загрузке ELF-файлов на компьютер пользователя, до момента окончательного скачивания полной версии файла. Такой подход позволит производить превентивную меру по защите автоматизированной системы от несанкционированной установки запрещенного программного обеспечения.

Положения, выносимые на защиту, соотнесены с пунктами паспорта специальности 05.13.19 — «Методы и системы защиты информации, информационная безопасность»: «6. Модели и методы формирования комплексов средств противодействия угрозам хищения (разрушения, модификации) информации и нарушения информационной безопасности для различного вида объектов защиты вне зависимости от области их функционирования» (результаты 2-3); «13. Принципы и решения (технические, математические, организационные и др.) по созданию новых и совершенствованию существующих средств защиты информации и обеспечения информационной безопасности» (результаты 1); «14. Модели, методы и средства обеспечения внутреннего аудита и мониторинга состояния объекта, находящегося под воздействием угроз нарушения его информационной безопасности» (результат 3).

СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ДИССЕРТАЦИИ

В научных журналах, **рекомендованных ВАК:**

1. Салахутдинова К.И., Лебедев И.С., Кривцова И.Е. Подход к выбору информативного признака в задаче идентификации программного обеспечения // Научно-технический вестник информационных технологий, механики и оптики -2018. - Т. 18. - № 2(114). - С. 278–285
2. Салахутдинова, К.И. Лебедев И.С., Кривцова И.Е., Сухопаров М.Е. Исследование влияния выбора признака и коэффициента (ratio) при формировании сигнатуры в задаче по идентификации программ // Проблемы информационной безопасности. Компьютерные системы. 2018. № 1. С. 136–141.
3. Салахутдинова, К.И. Лебедев И.С., Кривцова И.Е. Алгоритм градиентного бустинга деревьев решений в задаче идентификации программного обеспечения // Научно-технический вестник информационных технологий, механики и оптики. 2018. Т. 18, № 6.
4. Салахутдинова К.И., Малков В.В., Кривцова И.Е. Сравнительный анализ подходов к идентификации программного обеспечения // Безопасность информационных технологий - 2019. - Т. 26. - № 2. - С. 58-66
5. Салахутдинова К.И., Лебедев И.С., Кривцова И.Е., Анисимов А.С. Идентификации программного обеспечения в задаче аудита электронных носителей информации // Авиакосмическое приборостроение - 2019. - № 9. - С. 20-28
6. Салахутдинова К.И. Повышение точности идентификации программного обеспечения путем использования аддитивного критерия Фишберна // Информационные технологии -2019. - Т. 25. -№ 10. - С. 609-614
7. Бажаев Н., Давыдов А.Е., Кривцова И.Е., Лебедев И.С., Салахутдинова К.И. Подход к анализу состояния информационной безопасности беспроводной сети // Прикладная информатика - 2016. - Т. 11. - № 6(66). - С. 121-128
8. Кривцова И.Е., Салахутдинова К.И., Юрин И.В. Метод идентификации исполняемых файлов по их сигнатурам // Вестник Государственного университета морского и речного флота имени адмирала С.О. Макарова - 2016. - № 1(35). - С. 215-224

В изданиях, индексируемых в международных базах цитирования **Web of Science, Scopus**:

9. Lebedev I.S., Korzhuk V., Krivtsova I., Salakhutdinova K., Sukhoparov M.E., Tikhonov D. Using Preventive Measures for the Purpose of Assuring Information Security of Wireless Communication Channels // Proceedings of the 18th Conference of Open Innovations Association FRUCT - 2016, pp. 167-173
10. Bazhayev N., Lebedev I.S., Krivtsova I.E., Sukhoparov M.E., Salakhutdinova K., Davydov A.E., Shaparenko I.M. Evaluation of the available wireless remote devices subject to the information impact // 10th IEEE International Conference on Application of Information and Communication Technologies, AICT 2016 - Conference Proceedings (Azerbaijan, Baku, 12-14 October 2016) - 2016, pp. 1-6
11. Lebedev I.S., Krivtsova I.E., Korzhuk V., Bazhayev N., Sukhoparov M.E., Pecherkin S., Salakhutdinova K. The Analysis of Abnormal Behavior of the System Local Segment on the Basis of Statistical Data Obtained from the Network Infrastructure Monitoring // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) - 2016, Vol. 9870, pp. 503-511
12. Krivtsova I.E., Lebedev I.S., Salakhutdinova K.I. Identification of Executable Files on the basis of Statistical Criteria//Proceedings of the 20th Conference of Open Innovations Association FRUCT, IET - 2017, pp. 202-208
13. Salakhutdinova K., Lebedev I.S., Krivtsova I.E., Bazhayev N., Sukhoparov M.E., Smirnov P.I., Markelov D.V., Davydov A.E., Pecherkin S., Kolcherin D.V., Shaparenko I.M., Iskanderov Y. A Frequency Approach to Creation of Executable File Signatures for their Identification//11th IEEE International Conference on Application of Information and Communication Technologies, AICT 2017 - Conference Proceedings (Moscow, 20-22 september 2017), IET - 2017, pp. 261-267.
14. Salakhutdinova, K.I. Krivtsova I.E., Lebedev I.S., Sukhoparov M.E. An Approach to Selecting an Informative Feature in Software Identification // Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 2018. С. 318–327.
15. Salakhutdinova K.I., Lebedev I.S., Krivtsova I.E., Sukhoparov M.E. Studying the Effect of Selection of the Sign and Ratio in the Formation of a Signature in a Program Identification Problem // Automatic Control and Computer Sciences - 2018, Vol. 52, No. 8, pp. 1101–1104

16. Semenov, Viktor V., Ilya S. Lebedev, Mikhail E. Sukhoparov and Kseniya I. Salakhutdinova. Application of an Autonomous Object Behavior Model to Classify the Cybersecurity State. Internet of Things, Smart Spaces, and Next Generation Networks and Systems. NEW2AN 2019, ruSMART 2019. Lecture Notes in Computer Science - 2019, Vol. 11660, pp. 104-112

В **других научных журналах** и изданиях:

17. Салахутдинова К.И., Овсяникова В.В., Трофимов А.А., Бессонова Е.Е., Ефремов А.А., Настека А.В. Анализ защищенности систем "Умный дом"//Региональная информатика (РИ-2014). XIV Санкт-Петербургская международная конференция «Региональная информатика (РИ-2014)». Санкт-Петербург, 29-31 октября 2014 г.: Материалы конференции. \ СПОИСУ. – СПб, 2014. - 2014. - С. 124

18. Ефремов А.А., Трофимов А.А., Настека А.В., Салахутдинова К.И., Овсяникова В.В. Защита управляющих сигналов в системе «Умный дом»//Сборник тезисов докладов конгресса молодых ученых. Электронное издание. – СПб: Университет ИТМО, 2015. – 2015

19. Овсяникова В.В., Настека А.В., Ефремов А.А., Салахутдинова К.И., Трофимов А.А. Защита системы «Умный дом» от программных сбоев//Сборник тезисов докладов конгресса молодых ученых. Электронное издание. – СПб: Университет ИТМО, 2015. – 2015

20. Druzhinin N.K., Salakhutdinova K.I. Identification of executable file by dint of individual feature//ISPIT-2015. International Conference on Information Security and Protection of Information Technology. St. Petersburg, Russia, November 5-6, 2015, IET - 2015, pp. 45-47

21. Кривцова И.Е., Салахутдинова К.И. Применение x2-критерия для идентификации elf-файлов. V Всероссийский конгресс молодых ученых//Сборник ВКМУ – 2016

22. Салахутдинова К.И., Кривцова И.Е. Условия применения критерия Пирсона для идентификации исполняемых файлов. Региональная информатика "РИ-2016" - 2016

23. Салахутдинова К.И., Дружинин Н.К. Идентификация исполняемых файлов по их ассемблерным командам // Научные работы участников конкурса «Молодые ученые Университета ИТМО» 2016 года. 2017.

24. Салахутдинова К.И., Лебедев И.С. Применение методов статистического анализа для идентификаций версий программного обеспечения удаленных автономных объектов транспортных систем //Информационная безопасность регионов России (ИБРР-2017). 2017.

25. Салахутдинова К.И., Лебедев И.С. Использование градиентного бустинга в задаче сравнения сигнатур программ //Сборник тезисов докладов конгресса молодых ученых. 2018. 136-141 с.

26. Салахутдинова, К.И. Обзор существующих подходов по аудиту электронных носителей информации // Сборник тезисов докладов конгресса молодых ученых. Электронное издание. 2019.

Свидетельства о государственной регистрации **программы для ЭВМ**:

27. Ильин А.Г., Салахутдинова К.И., Юшковский А.В., Катаева В.А., Первушин А.О., Павлов К.С. Программа мониторинга Google Apps for Business. №2016614206, 18.04.2016.

28. Ильин А.Г., Салахутдинова К.И., Юшковский А.В., Катаева В.А., Первушин А.О., Павлов К.С. Программа для стеганографического сокрытия информации в медиа файлах. №2016614207,18.04.2016.

29. Ильин А.Г., Салахутдинова К.И., Юшковский А.В., Катаева В.А., Первушин А.О., Павлов К.С. Программа для поиска и анализа криптоконтейнеров с носителя информации. №2016611975, 15.02.2016.

30. Ильин А.Г., Салахутдинова К.И., Юшковский А.В., Катаева В.А., Первушин А.О., Павлов К.С. Программа для криминалистического анализа IM ICQ и Jabber. №2016614048, 13.04.2016.

31. Ильин А.Г., Салахутдинова К.И., Юшковский А.В., Катаева В.А., Первушин А.О., Павлов К.С. Программа для аудита событий информационной безопасности на основе модели MapReduce. №2016614208, 18.04.2016.

32. Салахутдинова К.И. Сравнение сигнатур исполняемых файлов. Свидетельство о государственной регистрации программы для ЭВМ №2019619363, 16.07.2019.

Автореферат диссертации

САЛАХУТДИНОВА
Ксения Иркиновна

МЕТОДИКА ИДЕНТИФИКАЦИИ ИСПОЛНЯЕМЫХ ФАЙЛОВ НА ОСНОВЕ СТАТИЧЕСКОГО АНАЛИЗА ХАРАКТЕРИСТИК ДИЗАССЕМБЛИРОВАННОГО КОДА ПРОГРАММ

Текст автореферата размещен на сайтах:
Высшей аттестационной комиссии Министерства науки и высшего образования Российской Федерации
https://vak.minobrnauki.gov.ru/
Федерального государственного бюджетного учреждения науки Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН)
http://www.spiiras.nw.ru/dissovet/

Подписано в печать "11" декабря 2019 г.
Формат 60x84 1/16. Бумага офсетная. Печать офсетная.
Усл.печ.л. 1,0. Тираж 100 экз.
Заказ №