УДК 004.7

Р.Р. Фаткиева, Д.К. Левоневский

ПРИМЕНЕНИЕ БИНАРНЫХ ДЕРЕВЬЕВ ДЛЯ АГРЕГАЦИИ СОБЫТИЙ СИСТЕМ ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ

Фаткиева Р.Р., Левоневский Д.К. **Применение бинарных деревьев для агрегации событий систем обнаружения вторжений.**

Аннотация. В статье рассматривается проблема выбора алгоритмов и структур данных для эффективной обработки событий, производимых системами обнаружения вторжений. Предложен подход к выполнению операций добавления и поиска записей с использованием сбалансированных бинарных деревьев. Приведено теоретическое и экспериментальное подтверждение эффективности разработанного подхода.

Ключевые слова: информационная безопасность, структуры данных, системы обнаружения вторжений, сетевой трафик, сетевые аномалии.

Fatkieva R.R., Levonevskiy D.K. **Application of Binary Trees for the IDS Events Aggregation Task.**

Abstract. This paper considers the problem of a choice of algorithms and data structures to achieve the effective processing of events generated by intrusion detection systems. The proposed approach is based on balanced binary trees and speeds up the operations of adding and searching records in the structure. The paper provides the theoretical and experimental confirmation of the efficiency of the developed approach.

Keywords: information security, data structures, intrusion detection systems, network traffic, network anomalies.

1. Введение. Использование крупными предприятиями межсетевых экранов, систем обнаружения атак, антивирусных средств не всегда приводит к полному пониманию процессов, происходящих в сетевой инфраструктуре. Проблема обеспечения безопасности в крупных системах имеет свои особенности. В частности, следует учитывать требования безопасности, связанные с использованием виртуализации (гипервизоров), регулировкой трафика между узлами комплекса, управлением правами доступа, определением периметра сети и его защитой. Конкретные технологии и решения для защиты распределенных систем ориентированы на узкий спектр решаемых задач. Поэтому для решения проблем безопасности в распределенных информационных системах применяют метод системной интеграции, использование которого значительно снижает количество инцидентов безопасности [1, 2]. Несмотря на это, остаются проблемы, которые требуют тщательного анализа и проработанного решения. В частности, сигнатурный и поведенческий анализ, реализованный в системах обнаружения вторжений, не всегда достаточен для детектирования новых видов угроз в автоматическом режиме. В этой ситуации, помимо применения средств активного обнаружения угроз, необходимо вести непрерывный мониторинг сетевой инфраструктуры на предмет ее нештатного функ-

ционирования, также называемого аномальной активностью. Критерии аномальной активности размыты, и ее выявление требует участия специалиста по информационной безопасности, которому должна быть предоставлена статистическая информация о сетевой активности в виде, удобном для оценки [4].

Проблема усугубляется тем, что исследуемые системы в силу своей масштабируемости содержат значительное число активно обменивающихся данными узлов, а применяемые средства анализа сетевой активности генерируют чрезмерное количество событий. К примеру, это относится к распространенным системам мониторинга и обнаружения вторжений Snort [3], Bro IDS, OSSEC HIDS, Ganglia и др. Все это приводит к возможности потери или недооценке важности отдельных событий. Для упорядочивания множества событий, протекающих в компьютерных системах, с целью их дальнейшего анализа необходимо рассмотреть существующие методы агрегации. Проблема заключается в ограниченности методов и средств анализа и визуализации приведенных выше данных о системных и сетевых событиях с точки зрения информационной безопасности.

2. Анализ релевантных работ. На практике обработка событий, происходящих в информационных системах, основана на решении двух типов задач: *классификации событий и эффективного хранения и поиска событий*.

Для классификации событий используются различные методы. Основные из них:

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

В работе [4] рассмотрены методы классификации на основе статистической модели поведения протокола прикладного уровня, а также на основе использования наивных байесовских моделей, идентификации приложений на основе изучения распределения размеров пакетов для сетевых пакетов, на основе профилей приложений. Интерес работы вызывает сравнение классификации по скорости, эффективности и стабильности. Показано, что на качество классификации влияют такие факторы, как важность выбора набора непротиворечивых и не-

избыточных атрибутов, чувствительность к выбору предполагаемых параметров распределения значений атрибутов внутри классов.

В работе [5] рассмотрены динамические байесовские модели, модели ближайших k-соседей, методы опорных векторов, проведен анализ существующих методов машинного обучения сетевых угроз, применимых для решения задачи обнаружения сетевых атак. Интерес представляет сравнение количества обнаружения и процессов ложных срабатываний для указанных моделей. Рассмотрен метод опорных векторов, при этом для минимизации размерности и повышения скорости обучения в качестве предобработки предложен метод главных компонент, который позволяет снизить время работы алгоритма.

Вызывает интерес подход, представленный в работе [6], как возможность агрегации и классификации событий, происходящих в сети. Решение задачи обнаружения вторжений основано на применении алгоритма AntMiner+. Конечный результат обнаружения может быть представлен в виде набора составных частей определенного вида, что позволяет в том числе определить сигнатуру атаки. С каждым набором составных частей ассоциируется вершина графа. Это позволяет при оценке характеристик нормальных событий в компьютерной системе вершины (узлы) рассматриваемого графа исключать их из рассмотрения за счет наложения ограничений (равенство значения характеристики фиксированному значению, принадлежность некоторому интервалу значений). Такой подход позволяет построить ограничения, выделяющие из непрерывного потока смешанных данных элементы, принадлежащие только целевому множеству, решая тем самым задачу обнаружения той или иной аномалии. Однако метод имеет ограничения применения к системам, требующим оперативного реагирования и в задачах, требующих высокой точности результатов в связи с низкой скоростью обработки информации. Как утверждают авторы, модификация алгоритма путем оптимизации расчета эвристики, вероятностей и качества решений позволит получить более точные результаты. Применение данного метода также возможно в системах отложенного аудита.

В работах [7, 8] исследовано применение нейронных сетей для обнаружения вторжений. Использование бинарной классификации сетевых пакетов с разделением на классы на классы «норма» / «атака» показали возможность использования сетей прямого распространения функций в многоклассовом случае.

Для решения задачи упорядочивания событий применяются алгоритмы хранения и обработки множеств однотипных и логически связанных данных, которые основаны на структурах данных. Тради-

ционно к таким структурам данных относятся: массивы, списки (связные списки, очереди, стеки), деревья, которые являются простыми в исполнении. Однако при выборе алгоритма скорость добавления и поиска является критической. Использование массивов требует поддержки массива в отсортированном состоянии для ускорения поиска. Сложность поиска становится в таком случае логарифмической, а скорость вставки остается линейной. Ввиду большого количества событий использование отсортированных массивов для решения поставленной задачи по этой причине неприемлемо. При использовании связанных списков операция вставки имеет константную сложность, но операция поиска становится линейной.

Оптимальным с точки зрения скорости является применение сбалансированных бинарных деревьев, так как это позволяет осуществить операции вставки и поиска элемента в бинарном дереве по ключу и имеет логарифмическую сложность.

3. Оценка использования структур данных для алгоритма агрегации событий. Реализация алгоритма с использованием обычных массивов приводит к тому, что сложность алгоритма в среднем случае будет $O(N^3)$, где N - количество записей в файле. Действительно, для каждой записи будет производиться линейный поиск в массиве групп, при обнаружении необходимой группы для каждого из наборов будет производиться линейный поиск требуемого элемента.

Два последовательных линейных списка поиска для одной записи дают сложность $O(X)\ O(Y)$, где X - количество групп, а Y - среднее количество элементов в каждом наборе адресов/портов. Но так как вся совокупность событий (N) разделяется на объединение непересекающихся множеств (групп), то X можно выразить через N:

$$X = x \cdot N,$$

где x - константа, обратная среднему количеству элементов в группе. Аналогично, Y тоже находится в линейной зависимости от N. В итоге, для одной записи получаем сложность обработки $O(N^2)$, для всех записей – $O(N^3)$.

Выделим следующие основные признаки применительно к оценке событий происходящих при передаче информационных потоков:

— номер правила, по которому сгенерировано событие – signature id;

— адрес источника события— ip source и source port;

— адрес цели события — ip destination и destination port;

— дата происхождения события.

Используя тот или иной признак за основу, можно объединять события в группы. Например, если взять за основной признак адрес источника события, то в одной группе окажутся все события с одинаковым адресом источника события, а вся совокупность событий разобьется на объединение непересекающихся групп событий (рисунок 1).

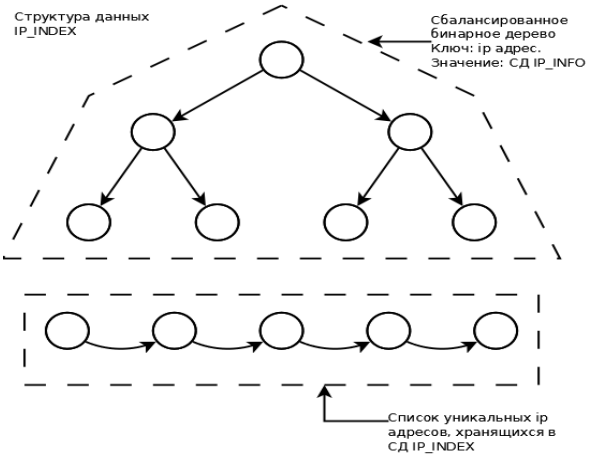

Рис. 1. Бинарное дерево с IP-адресом в качестве ключа

Однако стоит заметить, что если брать за основной признак адреса узлов сети или дату происхождения событий, то можно получить внушительное количество групп: количество узлов сети, адресуемое протоколом IPv4, равно нескольким миллиардам. В связи с этим целесообразно рассматривать как основной признак значение signature id — каждая группа будет соответствовать одному конкретному типу события, например, одному из видов сканирования портов. Бинарное дерево строится так, что в левом поддереве каждого элемента располагаются записи с меньшим значением signature id, в правом поддереве – с большим. Расположение элементов в бинарном дереве никак не связано с классификацией событий и их связью друг с другом.

Такой подход позволяет хранить все события, которые относятся к группам, каждая из которых будет соответствовать ровно одному типу событий. Если в каждой группе также произвести разбиение по другим признакам, то получим двухуровневую структуру. В качестве примера — на верхнем уровне события делятся на группы по типу события; на нижнем уровне в каждой группе два набора адресов и два набора портов делятся на группы по признаку адреса или порта, хранящие в себе количество событий с этими адресами/портами, а также временной интервал, в течение которого происходили события данно-

го типа. Такое структурирование позволяет быстро и эффективно проанализировать все сгенерированные сообщения о событиях.

4. Тестирование. Для оценки возможности использования сбалансированного бинарного дерева была разработана программная реализация метода. При тестировании применены материалы крупнейшей в мире конференции DEFCON, на официальном сайте которой доступны дампы сетевого трафика [9]. Результаты тестирования производительности программы на компьютерах с процессорами разной производительности представлены в таблицах 1 и 2. В таблицах приводится время построения бинарного дерева для исходного множества событий, т. е. время агрегации событий.

Таблица 1. Тестирование программы на реальных данных, процессор: Intel Core 2 Duo

Количество событий в файле (в миллионах)	Время работы программы (ms)
0.2	250
0.3	380
0.4	550
0.5	680
0.6	800
1.0	1260
1.8	2220
3.654978	4490
10.456365	14220
14.619912	17710
18.375484	24060

Файл для тестирования содержит 36 миллионов записей и имеет размер больше двух гигабайт. В качестве примера приведем, что для генерации такого файла из исходных 800 Гб дампов сетевого трафика системе обнаружения Snort потребовалось около 10 часов.

Таблица 2. Тестирование программы на реальных данных, процессор: Intel Core i3

Количество событий в файле (в миллионах)	Время работы программы (ms)
0.279322	232
0.609163	340
1.827489	1067
3.654987	2109
7.309956	4139
10.456365	6286
14.619912	8276
18.375484	10994

Зависимости времени работы от количества событий на разных процессорах и разных наборах данных приведены на рисунке 2.

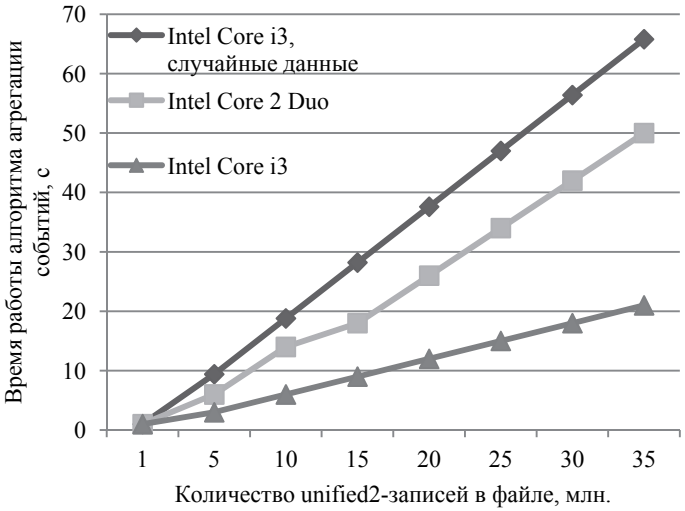

Рис. 2. Зависимость времени работы программы от процессора и исходных данных

Ранее при выборе структуры данных для эффективной реализации разработанного метода агрегации событий упоминалась возможность использования таких структур данных, как отсортированные массивы и линейные списки. Рассмотрим реализации алгоритма, использующего отсортированные массивы или линейные списки как структуру данных. Тестирование на производительность этих двух реализаций производилось на тестах, соответствующих худшему случаю (т. е. случайной выборке данных). В таблицах 3 и 4 представлены результаты этого тестирования.

Таблица 3. Тестирование производительности реализации алгоритма с использованием отсортированных массивов

Количество событий в файле (в миллионах)	Время работы программы (ms)
0.099855	2990
0.199808	8628
0.299754	16319
0.399423	25183
0.499848	35500
0.599848	47299
0.699730	59513
0.799235	74199
0.899650	88374
0.999000	103493

Последняя строка таблицы 3 показывает, что 1 миллион записей был обработан за 103,5 секунды, что в 50 раз медленнее аналогичного времени обработки записей на бинарных деревьях.

Таблица 4. Тестирование производительности алгоритма на основе линейных списков

Количество событий в лог файле (в миллионах)	Время работы программы (ms)
0.099855	1056
0.199808	2969
0.299754	5637
0.399423	8530
0.499848	11977
0.599848	16941
0.699730	20489
0.799235	25605
0.899650	34682
0.999000	35333
10.999170	1558871 (примерно 26 минут)

Для наглядного сравнения различных реализаций приведем графики времени обработки записей в зависимости от их количества (рисунок 3) при использовании различных алгоритмов обработки.

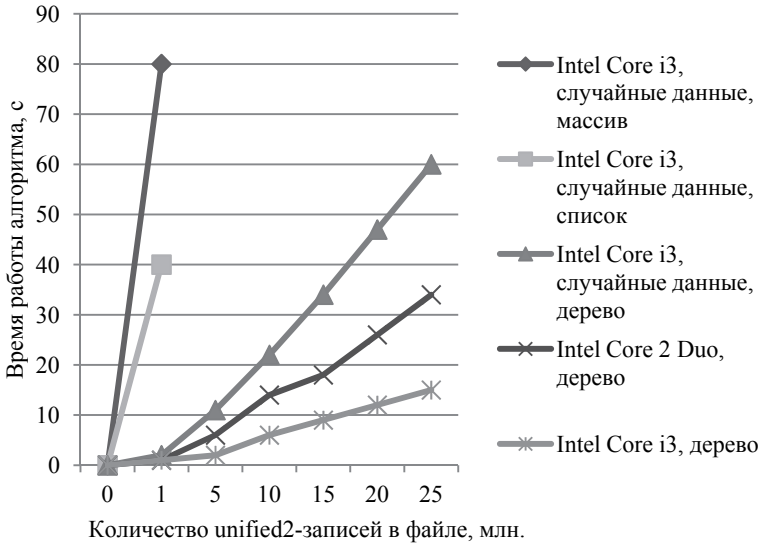

Рис. 3. Скорость обработки событий для различных алгоритмов

5. Заключение. Для организации эффективной обработки и быстрого доступа к базам событий критически важно найти такие структуры данных и алгоритмы, которые обеспечивают максимальную скорость добавления и поиска записей. Исследование показало, что этому условию соответствуют алгоритмы, основанные на бинарных деревьях, которые позволяют сделать сложность вышеприведенных операций логарифмической, что обеспечивает меньшее время работы по сравнению с аналогами.

Недостатком предлагаемого подхода является повышенное использование оперативной памяти. Этот недостаток может быть устранен путем использования баз данных на основе технологии NoSQL [10], которые позволяют организовывать базы данных с древовидной структурой.

Литература

1. *Котенко И. В., Юсупов Р. М.* Текущее состояние и тенденции развития в области построения безопасных компьютерных систем // Часть 5-й Российской мультиконференции по проблемам управления (МКПУ-2012) – конференция "Информационные технологии в управлении" (ИТУ-2012). Материалы конференции. СПб. 2012. С. 671–675.
2. *Левоневский Д.К., Фаткиева Р.Р.* Разработка системы обнаружения аномалий сетевого трафика // Научный вестник Новосибирского государственного технического университета. 2014. № 3(56). С. 108–114.
3. *Szmit M., Węzyk R., Skowroński M., Szmit A.* Traffic Anomaly Detection with Snort // Information Systems Architecture and Technology. Information Systems and Computer Communication Networks. Wrocław: Wydawnictwo Politechniki Wrocławskiej. 2007. pp. 181–187.
4. *Щербакова Н.Г.* Анализ IP-трафика методами Data Mining. Проблема классификации // Вычислительные и сетевые ресурсы. URL: http://problem-info.sscc.ru/2012-4/5.pdf (дата обращения: 17.02.2015).
5. *Носков А.Н., Чечулин А.А, Тарасова Д.А.* Исследование эвристических подходов к обнаружению атак на телекоммуникационные сети на базе методов интеллектуального анализа данных // Труды СПИИРАН. 2014. Вып. 37. С. 208–224.
6. *Таран А.А.* Приложения алгоритма Antminer+ к задаче классификации событий при анализе сетевого трафика // Известия ЮФУ. Технические науки. 2012. № 12(137). С. 60–67. URL: http://cyberleninka.ru/article/n/prilozheniya-algoritma-antminer-k-zadache-klassifikatsii-sobytiy-pri-analize-setevogo-trafika (дата обращения: 17.02.2015).
7. *Swimmer M.* Using the danger model of immune systems for distributed defense in modern data networks // Computer Networks. 2007. vol. 51. pp. 1315–1333.
8. *Клионский Д.М., Большев А.К., Геппенер В.В.* Применение искусственных нейронных сетей в сетевых технологиях // Нейроинформатика. 2011.
9. DEF CON Hacking Conference. URL: https://www.defcon.org/ (дата обращения: 17.02.2015).

10. *Ohene-Kwofie D., Otoo E.J., Nimako G.* O2-Tree: A Fast Memory Resident Index for NoSQL Data-Store // Proceedings of the 2012 IEEE 15th International Conference on Computational Science and Engineering (CSE). 2012. pp. 50–57.

References

1. Kotenko I. V., Yusupov R. M. [Current situation and trends of secure computer system development]. *Chast' 5-j Rossijskoj mul'tikonferencii po problemam upravlenija (MKPU-2012) – konferencija "Informacionnye tehnologii v upravlenii" (ITU-2012)* [5th part of Russian multiconference on management problems (MKPU-2012) – conference "Information Technologies in Management" (ITU-2012)]. Saint-Petersburg. 2012. pp. 671–675. (In Russ.).
2. Levonevskiy D.K., Fatkieva R.R. [Development of network anomaly detection system architecture]. *Nauchnyj vestnik Novosibirskogo gosudarstvennogo tehnicheskogo universiteta – Bulletin of Novosibirsk State Technical University*. 2014. vol. 3(56). pp. 108–114. (In Russ.).
3. Szmit M., Wężyk R., Skowroński M., Szmit A. Traffic Anomaly Detection with Snort. *Information Systems Architecture and Technology. Information Systems and Computer Communication Networks*. Wrocław: Wydawnictwo Politechniki Wrocławskiej. 2007 pp. 181–187.
4. Shcherbakova N.G. [IP traffic analysis by means of Data Mining. Classification problem]. *Vychislitel'nye i setevye resursy – Computational and network resources*. Available at: http://problem-info.sscc.ru/2012-4/5.pdf (accessed: 17.02.2015). (In Russ.).
5. Noskov A.N., Chechulin A.A, Tarasova D.A. [Research of heuristic approaches to the network attack detections on the basis of intellectual data analysis]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2014. vol. 37. pp. 208–224. (In Russ.).
6. Taran A.A. [Applications of Antminer+ algorithm to the task of network events classification]. *Izvestija JuFU. Tehnicheskie nauki – UFU Proceedings. Technical science*. 2012. vol. 12(137). pp. 60–67. Available at: http://cyberleninka.ru/article/n/prilozheniya-algoritma-antminer-k-zadache-klassifikatsii-sobytiy-pri-analize-setevogo-trafika (accessed: 17.02.2015). (In Russ.).
7. Swimmer M. Using the danger model of immune systems for distributed defense in modern data networks. *Computer Networks*. 2007. vol. 51. pp. 1315–1333.
8. Klionskiy D.M., Bolshev A.K., Geppener V.V. Primenenie iskusstvennykh nejronnykh setej v setevykh tekhnologiyakh [Application of artificial neural networks in ICT]. *Nejroinformatika - Neuroinformatics*, 2011. (In Russ.).
9. DEF CON Hacking Conference. Available at: https://www.defcon.org/ (accessed: 17.02.2015).
10. Ohene-Kwofie D, Otoo E.J., Nimako G. O2-Tree: A Fast Memory Resident Index for NoSQL Data-Store. Proceedings of the 2012 IEEE 15th International Conference on Computational Science and Engineering (CSE). 2012. pp. 50–57.

Фаткиева Роза Равильевна — к-т техн. наук, доцент, старший научный сотрудник лаборатории информационно-вычислительных систем и технологии программирования, Федеральное государственное бюджетное учреждение науки Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: моделирование информационных систем. Число научных публикаций — 50. rrf@iias.spb.su; 14-я линия В.О., д. 39, Санкт-Петербург, 199178; р.т.: +7-812-328-43-69, Факс: +7 (812)350-1113.

Fatkieva Roza Ravilievna — Ph.D., associate professor, senior researcher of computer and information systems and software engineering laboratory, St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS). Research interests: modeling of information systems. The number of publications — 50. rrf@iias.spb.su; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7-812-328-43-69, Fax: +7 (812)3501113.

Левоневский Дмитрий Константинович — младший научный сотрудник лаборатории информационно-вычислительных систем и технологии программирования, Федеральное государственное бюджетное учреждение науки Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН). Область научных интересов: исследование сетевых атак, статистический анализ и моделирование трафика локальных сетей. Число научных публикаций — 10. DLewonewski.8781@gmail.com; 14-я линия В.О., д. 39, Санкт-Петербург, 199178; р.т.: +7-812-328-43-69, Факс: +7 (812)350-1113.

Levonevskiy Dmitriy Konstantinovich — junior researcher of computer and information systems and software engineering laboratory, St. Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS). Research interests: network attacks research, statistical analysis and modeling of the network traffic. The number of publications — 10. DLewonewski.8781@gmail.com; 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone: +7-812-328-43-69, Fax: +7 (812)350-1113.

РЕФЕРАТ

Фаткиева Р.Р., Левоневский Д.К. **Применение бинарных деревьев для агрегации событий систем обнаружения вторжений.**

Методы, реализованные в системах обнаружения вторжений, не всегда достаточны для обнаружения новых видов угроз в автоматическом режиме. В этой ситуации, помимо применения средств активного обнаружения угроз, необходимо вести мониторинг сетевой инфраструктуры на предмет фактов подозрительной активности. Ее выявление требует участия специалиста по информационной безопасности, которому должна быть предоставлена статистическая информация о сетевой активности в виде, удобном для оценки.

Для упорядочивания множества событий IDS с целью их дальнейшего анализа рассматриваются различные структуры данных: массивы, списки, деревья. При выборе структуры данных критическим параметром является скорость алгоритма добавления и поиска.

Оптимальным с точки зрения скорости является применение сбалансированных бинарных деревьев, так как это позволяет осуществить операции вставки и поиска элемента в бинарном дереве по ключу и имеет логарифмическую сложность. Программная реализация подтверждает эффективность разработанного подхода по сравнению с аналогами.

Недостатком предлагаемого подхода является повышенное использование оперативной памяти. Этот недостаток может быть устранен путем использования баз данных на основе технологии NoSQL, которые позволяют организовывать базы данных с древовидной структурой.

SUMMARY

Fatkieva R.R., Levonevskiy D.K. **Application of Binary Trees for the IDS Events Aggregation Task.**

The methods implemented in intrusion detection systems are often not sufficient to detect new types of threats in the automatic mode. Therefore, one should, besides using intrusion detection and prevention systems, monitor the network infrastructure for any suspicious traffic. This activity needs the participation of an information security specialist who should be provided with the network statistics in a convenient form.

This paper considers various data structures (arrays, linked lists, binary trees) to put the variety of IDS events in order for their further analysis. When choosing data structure the critical parameter is the speed of the algorithm for adding and search.

The usage of balanced binary trees is optimal regarding the speed criterion because it enables performing the tasks of adding and searching in a tree using a key and guarantees the logarithmical complexity of these operations. The implementation of the algorithm proves the efficiency of this approach in comparison with other data structures.

The disadvantage of the proposed approach is the high use of the random access memory. This disadvantage can be eliminated by using the NoSQL technology that enables organizing tree-type databases.