ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ
И ЗАЩИТА ИНФОРМАЦИИ

DOI 10.21672/2074-1707.2020.49.4.144-155
УДК 004.855.2, 004.056

ОБЗОР ИНСТРУМЕНТОВ МАШИННОГО ОБУЧЕНИЯ И ИХ ПРИМЕНЕНИЯ В ОБЛАСТИ КИБЕРБЕЗОПАСНОСТИ

Статья поступила в редакцию 23.01.2020, в окончательном варианте – 26.02.2020.

Власенко Александра Владимировна, Кубанский государственный технологический университет, 350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
кандидат технических наук, заведующая кафедрой компьютерных технологий и информационной безопасности Института компьютерных систем и информационной безопасности, e-mail: Vlasenko@kubstu.ru

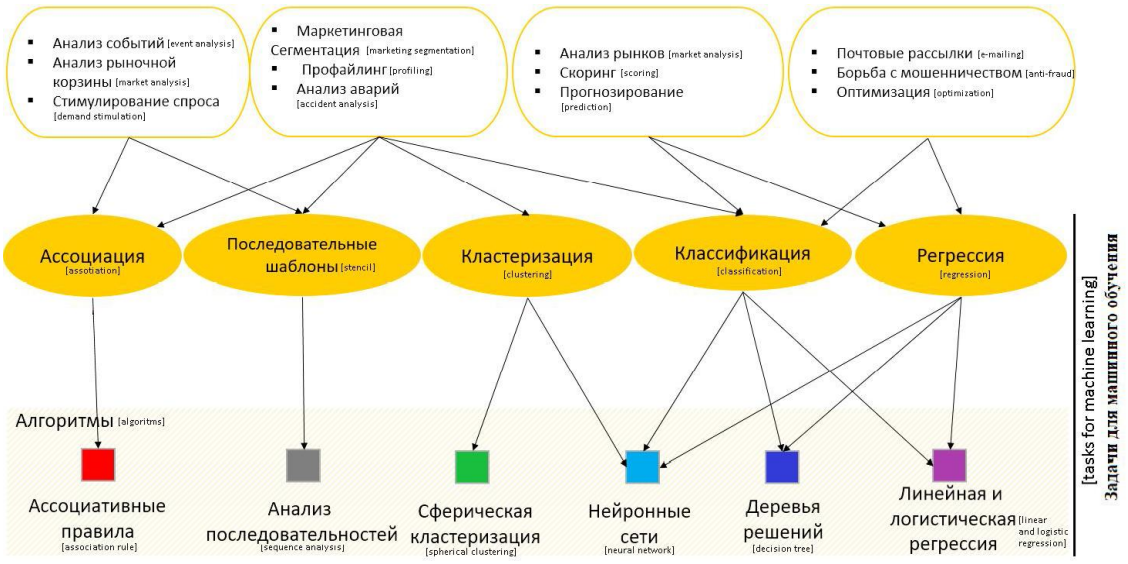
Дзьобан Павел Игоревич, Кубанский государственный технологический университет, 350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
кандидат технических наук, доцент кафедры компьютерных технологий и информационной безопасности Института компьютерных систем и информационной безопасности, e-mail: antiemoboy@mail.ru

Жук Роман Владимирович, Филиал «Макрорегион Юг» ООО ИК «СИБИНТЕК», 352800, Российская Федерация, г. Туапсе, ул. Карла Маркса, 36,
главный специалист, e-mail: goonerkrd@gmail.com

Машинное обучение заслужено привлекает интерес специалистов в области кибербезопасности. Благодаря тому, что аппаратные и вычислительные мощности становятся все более доступными, методы машинного обучения могут использоваться для анализа и классификации природы возникновения аномалий, вредоносных активностей из агрегированных метаданных. Методы машинного обучения подразделяются на контролируемое (классификация, регрессия) и неконтролируемое обучение (кластеризация, сокращение количества измерений объектов). Оба эти подхода могут быть применены в области кибербезопасности для анализа вредоносных активностей в режиме реального времени, что устраняет недостатки традиционных методов обнаружения таких активностей. В данной статье для анализа активности хостов предлагается использовать данные с применением технологии экспорта потоков NetFlow. Также будут рассмотрены принципы обнаружения аномалий в сетевом трафике с применением различных инструментов машинного обучения (экстремальное машинное обучение, случайный лес, повышение градиента, логистическая регрессия), приведены примеры и успешные практики реализации методов обнаружения аномалий в сети.

Ключевые слова: кибербезопасность, атаки, сетевые аномалии, риски, мониторинг, сеть, машинное обучение, градиент, ботнет, агрегация, кластеризация, классификация, регрессия

Графическая аннотация (Graphical annotation)



ANALYTICAL REVIEW OF MACHINE LEARNING TOOLS AND THEIR APPLICATIONS IN THE FIELD OF CYBER SECURITY

The article was received by the editorial board on 23.01.2020, in the final version – 26.02.2020.

Vlasenko Alexandra V., Kuban State Technological University, 2 Moskovskaya St., Krasnodar, 350072, Russian Federation,

Cand. Sci. (Engineering), Head of the Department of Computer Technologies and Information Security of the Institute of computer systems and information security, e-mail: Vlasenko@kubstu.ru

Dzoban Pavel I., Kuban State Technological University, 2 Moskovskaya St., Krasnodar, 350072, Russian Federation,

Cand. Sci. (Engineering), Associate Professor of the Department of Computer Technologies and Information Security of the Institute of Computer Systems and Information Security, e-mail: antiemoboy@mail.ru

Zhuk Roman V., Branch «Macroregion South» Ltd Co IC «SIBINTEK», 36 Karl Marks St., Tuapse, 352800, Russian Federation,

chief specialist, e-mail: goonerkrd@gmail.com

Machine training deservedly attracts the interest of specialists in the field of cybersecurity. With the increasing availability of hardware and computing power, machine learning methods can be used to analyze and classify the nature of anomalies and malicious activities from aggregated metadata. Machine learning methods are divided into controlled (classification, regression) and uncontrolled learning (clustering, reducing the number of measurements of objects). Both of these approaches can be applied in the area of cybersecurity to analyse malicious activities in real time, thus eliminating the shortcomings of traditional methods of detecting such activities. This article proposes to use data using NetFlow flow export technology to analyze host activity, and also discusses the principles of detecting anomalies in network traffic using various machine learning tools (extreme machine learning, random forest, gradient increase, logistic regression), and provides examples and good practices of implementing anomaly detection methods in the network.

Key words: cybersecurity, attacks, network anomalies, risks, monitoring, network, machine learning, gradient, botnet, aggregation, clustering, classification, regression

Введение. В современном обществе Российской Федерации отрасль «информационная безопасность» (ИБ) постоянно развивается. Мотиваций и причин для развития этой отрасли достаточно много, но все они между собой тесно взаимосвязаны: начиная от постоянного роста компетенций злоумышленников и заканчивая требованиями правового поля, вынужденного «успевать» за тенденциями и современными реалиями. В данной статье рассматривается один из самых актуальных блоков отрасли – кибербезопасность. Сложные и все новые виды атак уже сегодня являются нормой; они становятся все более частыми и широко распространенными. Эта постоянная эволюция также требует инноваций в области кибербезопасности, занимающей особое место в отрасли.

Существуют решения и их комбинации (гибридизация), которые широко используются флагманами отрасли в области цифровизации и искусственного интеллекта. Системы обнаружения (далее – IDS) и предотвращения сетевых вторжений (далее – IPS) отслеживают вредоносные активности в сети и/или нарушения политики ИБ «владельца риска» на различных уровнях – пользователь, сегмент сети, организация, провайдер и т.д.

IDS и IPS на основе сигнатур опираются на известные сигнатуры и эффективно обнаруживают вредоносные активности, которые соответствуют этим сигнатурам. IDS и IPS на основе «поведенческого анализа», с другой стороны, аккумулируют данные, что является нормальным состоянием для системы, и сообщают о любом триггере, который отклоняется от принятого значения с выходом за границы заданного интервала. Оба типа систем довольно успешно применяются в лучших мировых практиках, хотя и не лишены недостатков.

Системы на основе сигнатур полагаются на сигнатуры известных угроз и поэтому неэффективны для атак «нулевого дня» или использования злоумышленниками новых образцов вредоносного программного обеспечения (ПО). Традиционные системы, использующие «анализ поведения», основаны на стандартном статичном шаблоне, который трудно динамически изменять в связи с растущей сложностью сетей и приложений. Следовательно, этот подход может быть неэффективен для обнаружения аномалий. Полный анализ пакетов данных – это еще один вариант, однако он требует значительных затрат вычислительных ресурсов и сопряжен с риском раскрытия конфиденциальной информации пользователей.

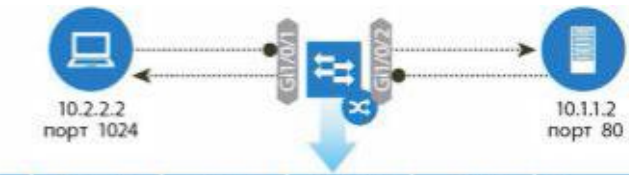
Машинное обучение заслужено привлекает интерес специалистов по теории и практике обеспечения ИБ, и в частности в области кибербезопасности. Благодаря тому, что аппаратные и вычислительные мощности становятся все более доступными, методы машинного обучения могут исполь-

зоваться для анализа и классификации природы возникновения аномалий, вредоносных активностей из агрегированных метаданных. Существуют сотни алгоритмов и подходов к машинному обучению, которые в целом подразделяются на контролируемое и неконтролируемое обучение. Подходы к обучению под наблюдением (контролируемые) выполняются в контексте классификации, (где ввод соответствует выводу) или регрессии (когда ввод отображается в непрерывный вывод). Самостоятельное обучение в основном достигается с помощью кластеризации и применяется для исследовательского анализа, сокращения количества измерений объектов. Оба эти подхода могут быть применены в области кибербезопасности для анализа вредоносных активностей в режиме реального времени, что устраняет недостатки традиционных методов обнаружения таких активностей.

Для защиты сети от угроз специалистам требуются интеллектуальные решения, которые носят всеобъемлющий характер, основаны на анализе поведения и дополняют существующие зонные средства обеспечения безопасности. Одно из таких решений заключается в использовании сетевой инфраструктуры в качестве датчика. Для анализа активности хостов можно использовать данные с применением технологии экспорта потоков sFlow и NetFlow. Изначально они разрабатывались для мониторинга и устранения неисправностей внутри сети [16].

Записи NetFlow предоставляют достаточно информации, чтобы однозначно идентифицировать трафик. NetFlow создавался как технология пакетной коммутации для маршрутизаторов Cisco. Положенная в основу NetFlow идея заключается в том, что первый пакет потока генерирует на коммутаторе или маршрутизаторе запись коммутации NetFlow. В дальнейшем эта запись используется при обработке пакетов из того же потока вплоть до его окончания. Поиск конкретного маршрута в таблице маршрутизации требуется только для первого пакета потока. NetFlow вместе с открытой версией стандарта IPFIX уже широко используется для мониторинга и управления сетью. Доступность данных NetFlow вместе с функциями конфиденциальности делает его эффективным средством вычисления сетевых аномалий.

Пример работы технологии NetFlow представлен на рисунке 1.



Кеш потока

Время начала	Интерфейс	IP-адрес отправителя	Порт отправителя	IP-адрес получателя	Порт получателя	Протокол	Отправлено пакетов	Отправлено байтов
10:20:12.221	Gig 1/0/1	10.2.2.2	1024	10.1.1.2	80	TCP	5	1025
10:20:12.871	Gig 1/0/1	10.2.2.2	80	10.2.2.2	1024	TCP	17	28712

Рисунок 1 – Пример работы NetFlow

Стандарт sFlow, называемый также выборкой потоков, поддерживается на разных платформах [15]. Он представляет собой средство для экспорта усеченных пакетов, а также счетчиков интерфейса. Последовательное развитие sFlow сделало возможным получение им статуса отраслевого стандарта для экспорта пакетов на уровне 2 [16].

sFlow – это технология выборочного захвата пакетов. Из всего трафика через интерфейс выбирается «1 из n» пакетов (где n – частота выборки). Первые *x* байт (128 байт для sFlow версии 5) захваченного пакета копируются и экспортируются в пакеты UDP, называемые дейтаграммами sFlow. Первые *x* байт – это данные заголовка, необходимые для восстановления информации о трафике. Однако поскольку sFlow ориентирован на пакеты, то некоторые IP-потоки могут пропускаться. Когда пакеты захватываются для анализа, они не всегда представляют все IP-потоки (диалоги), проходящие через интерфейс. В результате IP-потоки, пакеты которых не собраны, не учитываются, что создает пробелы в анализе картины диалогов в сети.

Так как для агрегации метаданных, дальнейшего их парсинга, построения таблиц корреляции, вывода и подачи оператору отфильтрованной информации необходимы все пакеты, то нами в данной статье будет отдано предпочтение технологии NetFlow.

Детальная характеристика технологии NetFlow. С тех пор как NetFlow стал отраслевым стандартом для сбора данных сеансов, данные NetFlow предоставляют информацию, которая может быть использована для определения использования трафика в сети и состояния ресурсов. Таким образом, технология NetFlow позволяет обнаруживать сетевые аномалии и потенциальные кибератаки. Данный инструментарий полезен для идентификации узловых устройств, которым требуется увеличить или уменьшить полосу пропускания, что скажется на повышении эффективности работы сети.

Такие инструменты, как NfSen/NfDump1, могут анализировать данные с NetFlow и отслеживать аномалии в сетевом трафике. Такой инструментарий успешно применяется для обеспечения мониторинга и управления сетью. Сегодня обеспечен широкий выбор инструментов анализа угроз и обнаружения аномалий, использующих трафик NetFlow [17].

NetFlow v5, NetFlow v9 и открытый стандарт IPFIX широко используются для решения различных задач построения защищенной IT-инфраструктуры. Записи NetFlow v5 включают данные, документирующие IP-адреса источника и назначения, порты источника и назначения и транспортный протокол. Процесс формирования отчетов о данных NetFlow включает захват потоков IP, агрегирование потоков на коммутаторе или маршрутизаторе и экспорт их в коллектор NetFlow. Процесс состоит из следующих этапов:

а) конфигурация NetFlow настраивается для захвата потоков и помещения их в кеш NetFlow;

б) экспорт NetFlow настраивается для отправки потоков коллектору;

в) в кеше NetFlow ведется поиск устаревших потоков с помощью активных таймеров, неактивных таймеров и ограничений на кеш;

г) одновременно в кеше NetFlow осуществляется поиск завершенных потоков – путем анализа флагов сброса TCP [TCP RST] и завершения [FIN];

д) найденные в пунктах «в» и «г» потоки экспортируются на сервер коллектора NetFlow;

е) 20–25 потоков объединяются в пакет и транспортируются на сервер коллектора NetFlow в формате User Datagram Protocol (UDP);

ж) программное обеспечение коллектора NetFlow создает из полученных данных отчеты в реальном времени или исторические отчеты.

Инструменты NetFlow v9 и IPFIX обладают возможностью расширенной настройки. Это позволяет использовать дополнительные информативные поля, такие как имена пользователей, MAC-адреса и URL-адреса [16, 17].

Данные сетевого трафика. Сетевой трафик определяется как однонаправленная последовательность совокупности передаваемых пакетов с некоторыми общими свойствами, которые проходят через сетевое устройство. Записи трафика включают в себя различную информацию: IP-адреса, количество пакетов и байтов, временную метку, тип обслуживания, порты приложения, интерфейсы ввода и вывода и другую информацию [3,5]. Данные сеанса, содержащие IP-адреса клиента, номера порта клиента, IP-адреса сервера, номера порта сервера и протокола, включенного в данные потока, важны для идентификации соединения. Исследуя корреляцию запросов и ответов клиент-серверных диалогов, анализируя трафик, можно найти значимые траектории и связь сетевой аномалии с тем или иным событием/инцидентом ИБ [8, 11].

Выходные данные NetFlow включают, но не ограничиваются (в зависимости от версии инструмента), следующие атрибуты [16, 17]:

- время начала записанного потока;
- продолжительность потока;
- используемый протокол (TCP, UDP и т.д.);
- IP-адрес источника;
- исходный порт;
- направление общения;
- адрес назначения;
- порт назначения;
- протокол состояния;
- тип используемого сервиса;
- тип услуги назначения;
- общее количество обмененных пакетов;
- общее количество обмененных байтов;
- количество байтов, отправленных источником;
- метка, назначенная этому сетевому потоку.

Обнаружение аномалий в сетевом трафике. Традиционные методы обнаружения аномалий в сетевом трафике, такие как обнаружение вторжений и глубокая проверка пакетов, обычно требуют использования необработанных данных или подписей, опубликованных производителями [2].

Глубокая проверка пакетов предоставляет более точные данные, но требует проведения значительных объемов вычислений [4]. Данный инструмент «не работает» с зашифрованными данными. В современных реалиях это является негативным признаком, противоречит политике ИБ и стандартам охраны сведений конфиденциального характера, содержащих информацию о пользователях.

Данные Netflow, в сравнении с инструментами глубокой проверки пакетов, не содержат такой конфиденциальной информации и широко используются администраторами сетей. При использовании правильных методов анализа данные с NetFlow могут стать информативным источником для обнаружения аномалий. Один из основных недостатков NetFlow связан с объемом генерируемых данных. Это прямо влияет на точность результатов, количество информации и соответственно на вычислительные ресурсы, требуемые для обработки данных (хранения, накопления, изменения, передачи и пр.) [15].

Методы машинного обучения. Машинное обучение – это инструмент анализа данных, который используется для эффективного выполнения конкретных задач без применения явных инструкций, а опирается на шаблоны и умозаключения [2]. Возможности машинного обучения используются для решения различных проблем, в том числе и в области кибербезопасности. Постоянный анализ и обучение позволит генерировать различные прогнозы развития кибератак, моделировать различные ситуации и обнаруживать аномальные активности в сети.

Крупные отечественные провайдеры используют машинное обучение, интегрированное в облачные интеллектуальные системы, для выявления вредоносного и нелегитимного контента, изоляции зараженных хостов и вывода информативных графиков и дашбордов об общем состоянии того или иного сегмента [7]. Одной из основных трудностей в машинном обучении является создание интеллектуальных систем, способных изучать последовательные задачи, а затем передавать знания из ранее изученного «фундамента» для решения новых задач. Такая возможность называется непрерывным машинным обучением или интеллектуальными системами непрерывного обучения [9]. Применение машинного обучения для обнаружения ботнетов было широко исследовано, в том числе с использованием контролируемого машинного обучения [3, 7, 8, 11, 15, 16, 17]. Машинное обучение рассматривается как решение для аналитики метаданных, полученных с NetFlow. При этом основной проблемой будет выбор параметров для парсинга и корреляции, для обеспечения наилучших результатов при решении поставленных задач. Некоторые из распространенных методов машинного обучения подвергаются анализу в данной статье [10].

Экстремальное машинное обучение (Extreme Learning Machine, ELM) – это алгоритм обучения, который использует нейронные сети с прямой связью с одним или несколькими слоями скрытых узлов. Эти скрытые узлы настраиваются случайным образом, и соответствующие им выходные веса аналитически определяются алгоритмом. По словам создателей [18], этот алгоритм обучения может дать хорошую производительность обобщения и может учиться в тысячу раз быстрее, чем обычные алгоритмы обучения для нейронных сетей с прямой связью. Также стоит отметить, что аномалия характеризуется, как правило, не только экстремальными значениями отдельных признаков. Иллюстрация алгоритма экстремального машинного обучения (ELM) представлена на рисунке 2.

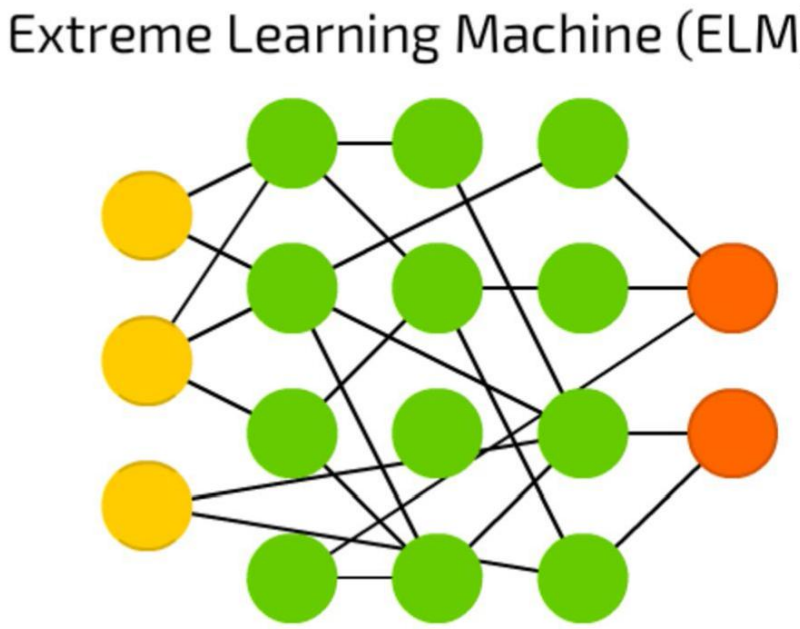


Рисунок 2 – Упрощенная иллюстрация алгоритма экстремального машинного обучения (ELM)

Случайный лес (Random forest, RF) – это контролируемый алгоритм машинного обучения, который предусматривает использование нескольких деревьев решений для выполнения задач классификации и регрессии. Алгоритм случайного леса считается ансамблевым алгоритмом машинного обучения, так как он включает в себя концепцию большинства рёбер нескольких деревьев. Выходные данные алгоритма, представленные как прогнозирование классов, определяются из совокупного результата всех классов, прогнозируемых отдельными деревьями. В последних исследованиях [7, 16, 18] изучались возможности использования алгоритма случайного леса в анализе кибератак, в частности, инъекционных атак; для фильтрации спама, для обнаружения вредоносных программ и многое другое.

Все деревья строятся независимо по следующему алгоритму:

1. Выбирается подвыборка обучающей выборки произвольного размера, по ней строится дерево (для каждого дерева – своя подвыборка).
2. Для построения каждого расщепления в дереве просматриваются максимальные значения случайных признаков (для каждого нового расщепления – свои случайные признаки).
3. Далее определяется наилучший признак и расщепление по нему (по заранее заданному критерию).

Дерево строится, как правило, до исчерпания выборки (пока в листьях не останутся представители только одного класса). Однако в современных реализациях алгоритма есть параметры, которые ограничивают высоту дерева, число объектов в листьях и число объектов в подвыборке, при котором проводится расщепление. Схематично, алгоритм случайного леса представлен на рисунке 3.

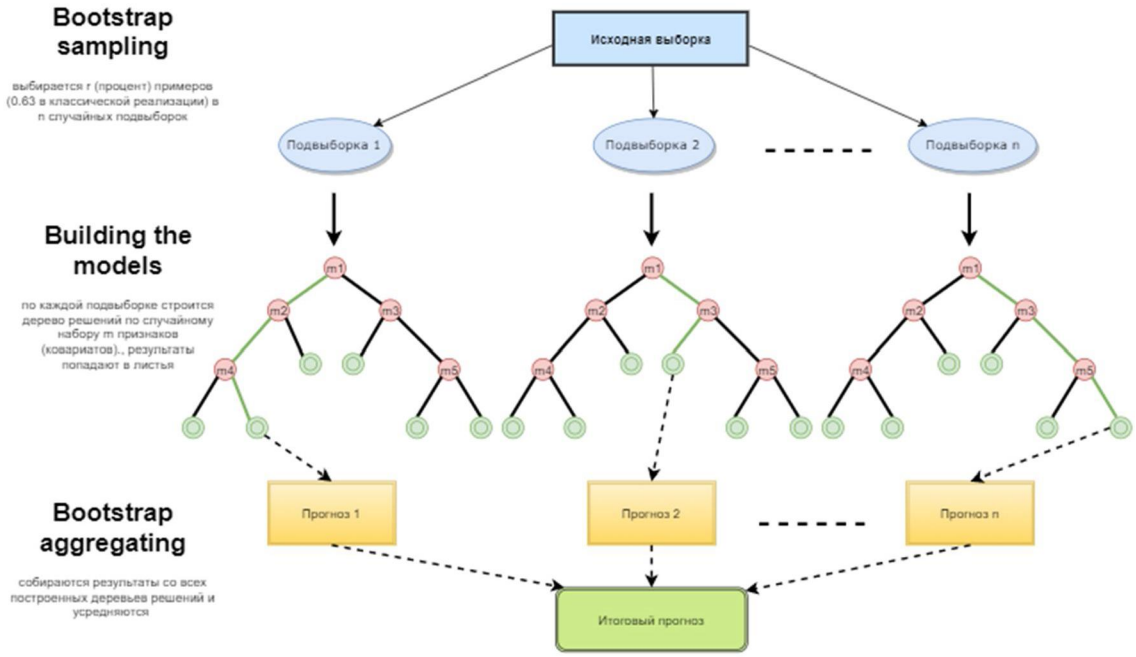


Рисунок 3 – Схема работы алгоритма случайного леса

Повышение градиента — это метод машинного обучения для задач регрессии и классификации, который создает модель прогнозирования в форме множества моделей слабого прогнозирования [15, 17], обычно деревьев решений. Этот метод использует идею о том, что следующая модель будет учиться на ошибках предыдущей. Модели имеют неравную вероятность появления в последующих моделях, и чаще появятся те, которые дают наибольшую ошибку. Предсказатели могут быть выбраны из широкого ассортимента моделей, например, деревья решений, регрессия, классификаторы и т.д. Из-за того, что предсказатели обучаются на ошибках, совершенных предыдущими, требуется меньше времени для того, чтобы добраться до реального ответа. Однако выбирать критерий остановки алгоритма следует с осторожностью, иначе это может привести к переобучению модели и искажению результатов. Цель любого алгоритма обучения – определить функцию потерь и минимизировать её. Когда алгоритм достигает стадии, на которой остатки не имеют какого-либо шаблона, который можно было бы смоделировать, то моделирование остатков будет остановлено – в противном случае это может привести к переобучению. Иными словами,

предсказания обновляются таким образом, чтобы сумма отклонений стремилась к нулю и предсказанные значения были близки к реальным. Математически это означает минимизацию функции потерь так, чтобы тестовые потери достигли своего минимума.

Исходя из вышеизложенного, за алгоритмом повышения градиента необходимо итеративно применять паттерны отклонений и улучшать предсказания, как показано на рисунке 4. Как только будет достигнут момент, когда отклонения не имеют никакого паттерна, стоит прекратить модернизировать модель (иначе это может привести к переобучению) [9]. Таким образом, для применения метода повышения градиента необходимо соблюдать следующую последовательность действий:

- построение простой модели;
- анализ ошибок;
- идентификация точек, которые не вписываются в простую модель;
- модификация моделями, обрабатывающими сложные случаи, которые были выявлены на начальной модели;
- агрегация и наложение построенных моделей, с определением весов каждого предсказателя [4, 17].

Пример работы алгоритма повышения градиента представлен на рисунке 4.

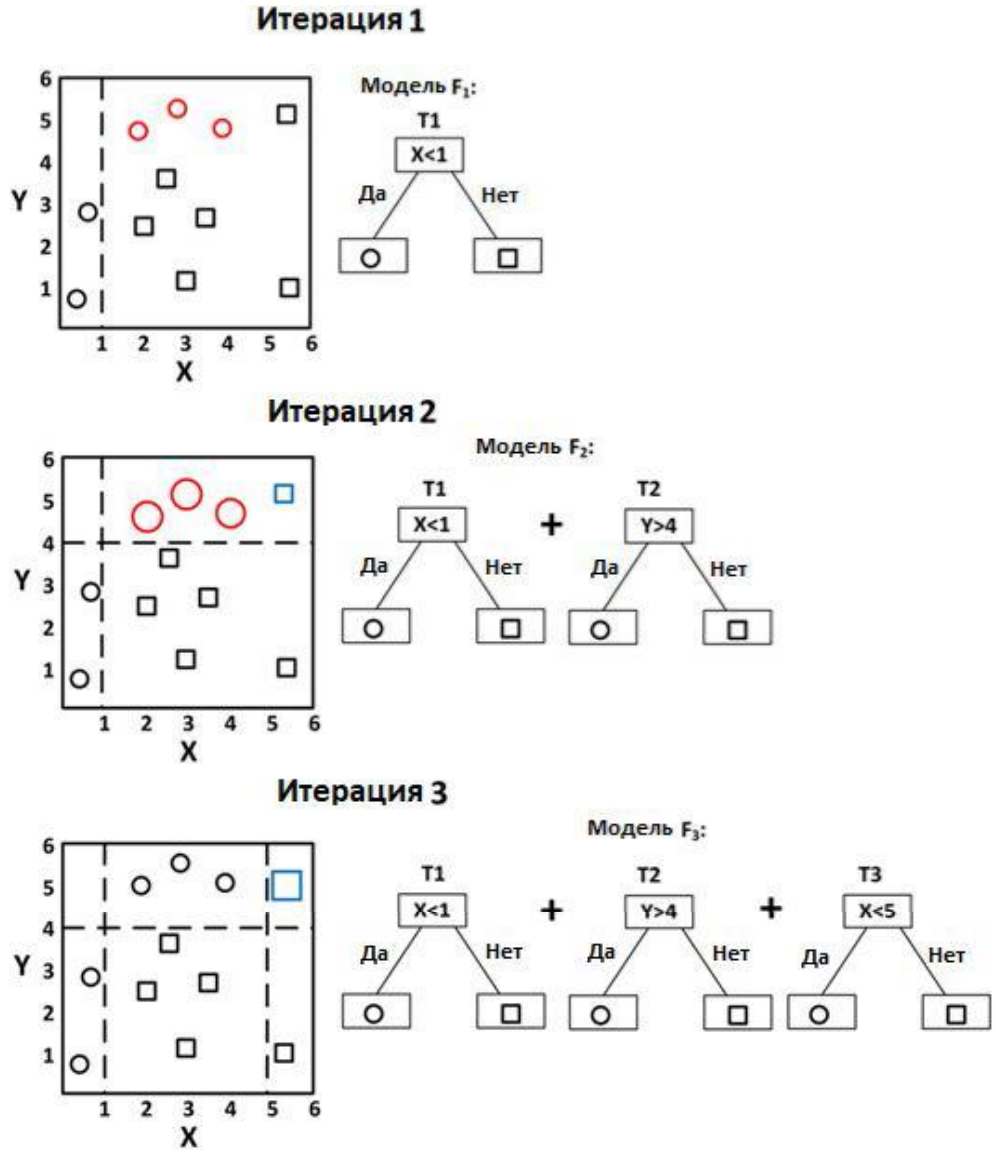


Рисунок 4 –Пример работы алгоритма повышения градиента

Логистическая регрессия является контролируемой моделью обучения, которая используется в качестве метода для двоичной классификации. Сам термин заимствован из статистики. В основе метода лежат логистические функции – сигмовидная кривая, которая полезна для ряда областей, включая нейронные сети. Логистическая регрессия моделирует вероятность проблем классификации с двумя возможными результатами и может использоваться для идентификации сетевого трафика как вредоносного (true) или нет (false).

Основная идея логистической регрессии заключается в том, что пространство исходных значений может быть разделено линейной границей (т.е. прямой) на две области, соответствующие классам. Под линейной границей подразумевается прямая линия – в случае двух измерений, в случае трех измерений – плоскость, и т.д. Эта граница задается в зависимости от имеющихся исходных данных и обучающего алгоритма. Для того чтобы алгоритм отрабатывал корректно, точки исходных данных должны разделяться линейной границей на две вышеупомянутых области. Если точки исходных данных удовлетворяют этому требованию, то их можно назвать линейно разделяемыми.

Пример работы метода логистической регрессии представлен на рисунке 5.

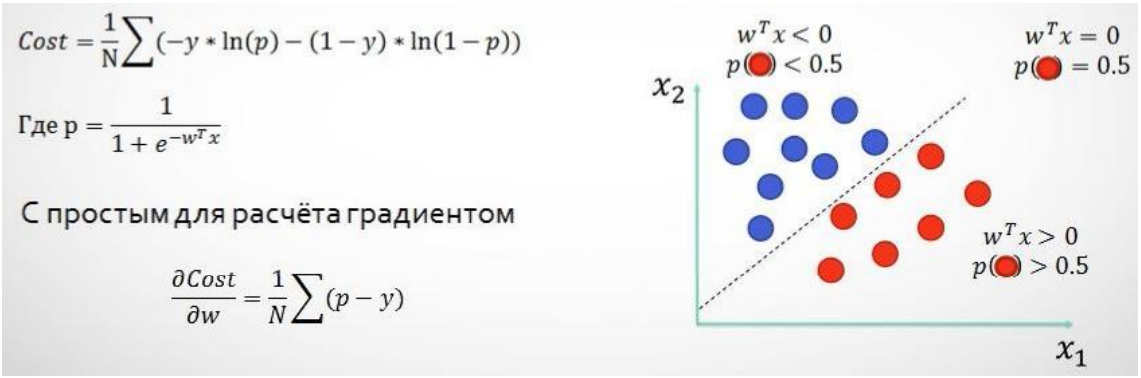


Рисунок 5 – Пример работы метода логистической регрессии

Методы обнаружения аномалий в сети. Аномалии – это объекты или инциденты, которые отличаются на более чем максимально допустимое отклонение (погрешность) от принятого значения. Таким образом, обнаружение сетевых аномалий сводится к их выявлению. Выявлять аномалии предлагается с использованием построенных триггеров на основе связи с редкими событиями при наблюдении за трафиком. Такие связи вызывают подозрение аналитиков, так как они существенно отличаются от большинства идентичных состояний легитимного сетевого трафика.

В машинном обучении обнаружение аномалий применяется в различных областях, включая обнаружение вторжений, обнаружение мошенничества и обнаружение нарушений в экосистеме.

Существует три широких категории обнаружения аномалий: неконтролируемые, контролируемые и полууправляемые [2, 13]. Некоторые из популярных методов обнаружения включают, среди прочего, основанный на плотности k-ближайший сосед, одноклассовый Support Vector Machine (SVM, набор схожих алгоритмов обучения с учителем, использующихся для задач классификации и регрессионного анализа), Байесовские сети, обнаружение выбросов на основе кластерного анализа [8,14]. Ряд систем анализа используют вышеуказанные методы обнаружения.

Кооперативный адаптивный механизм защиты сети (CAMNEP) – это система обнаружения вторжений в сеть. Система CAMNEP использует набор моделей обнаружения аномалий, которые поддерживают модель ожидаемого трафика в сети и сравнивают ее с реальным трафиком, чтобы выявить расхождения, которые определены как возможные атаки [1, 9]. Механизм имеет три основных уровня, которые оценивают трафик: детекторы аномалий, модели доверия и агрегаторы аномалий [3, 7, 16].

Устройства IDS/IPS применяются для оперативного реагирования на кибератаки. С учетом того факта, что легитимный трафик значительно превосходит по объему вредоносный, низкая частота ложных срабатываний делает систему непригодной для использования, так как нарушает отказоустойчивость и непрерывность IT-сервисов.

Корректность обрабатывания доверительной модели CAMNEP заключается в перекрестной проверке аномалий в моделях тригеров. Каждая из этих моделей основана на различных характеристиках трафика. Для классификации кибератаки потоки из набора должны находиться вблизи центроидов [19]. На практике большинство атакующих потоков приходится на окрестности одного центроида, как показано на рисунке 6.

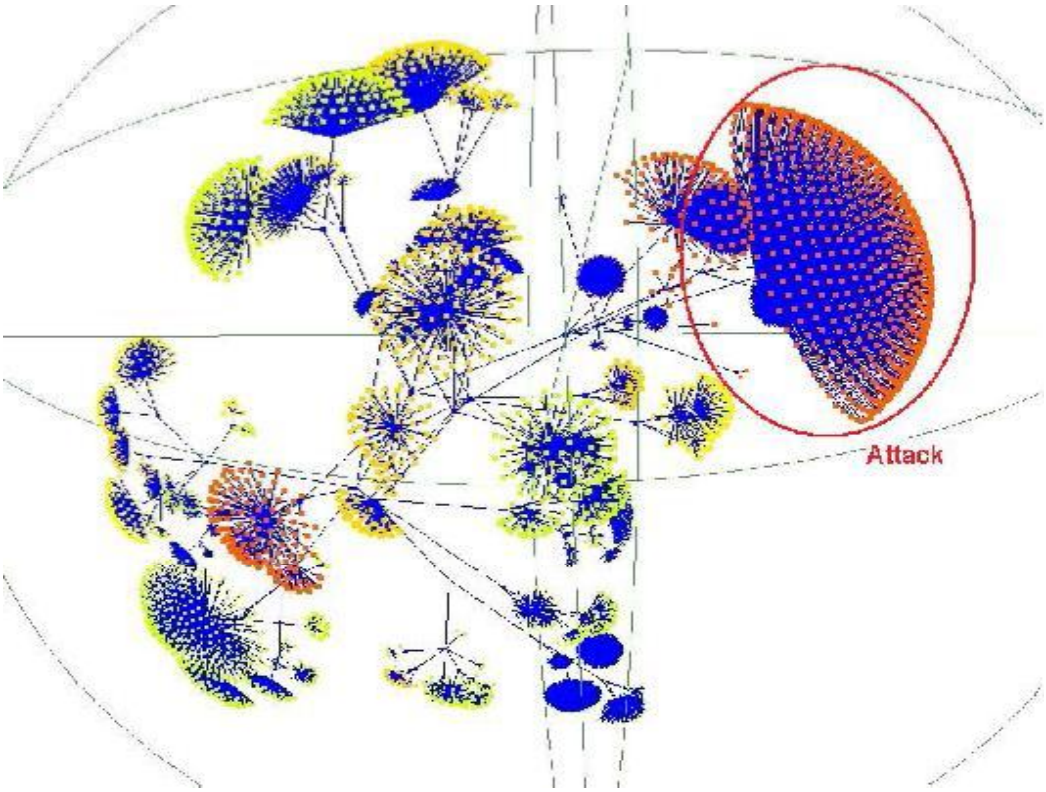


Рисунок 6 – Пример визуализации системы обнаружения вторжений CAMNEP

Уровень обнаружения аномалий анализирует NetFlow с использованием различных алгоритмов обнаружения аномалий, каждый из которых использует свой набор функций. Выходные данные агрегируются в события и отправляются в модели доверия. Модель доверия отображает NetFlow в транспортные кластеры. NetFlow с похожими поведенческими паттернами сгруппированы вместе. Уровень агрегатора создает композитный выход, который объединяет индивидуальное мнение нескольких детекторов аномалий [17].

Миннесотская система обнаружения вторжений (MINDS) использует набор методов интеллектуального анализа данных для автоматического обнаружения атак. Система MINDS создает контекстную информацию для каждого оцениваемого NetFlow с использованием следующих функций: количество сеансов NetFlow с исходного IP-адреса относительно других активных хостов из данного сегмента сети, то же по отношению к хосту назначения, то же по отношению к порту источника, активность NetFlow от одного хоста источника к тому же порту назначения. Значение аномалии для NetFlow основано на корреляции со значениями эталонной выборки [9, 16].

Методы кластеризации на основе поведения ботнета. Современный ботнет является флагманом среди наиболее эффективных методов атаки, доступных для современного киберпреступника. Ботнет – это совокупность зараженных рабочих станций (АРМ и прочих хостов) с сетевыми интерфейсами, вычислительные ресурсы которых задействованы в теневом режиме для выполнения вредоносного кода [5, 10]. Каждая из этих конечных точек или «ботов» регулярно взаимодействует с сервером контроля и управления (C & C), и весь ботнет может быть использован для управления гигантскими атаками DDoS (распределенного отказа в обслуживании), а также для нарушения конфиденциальности и целостности информации или распространения спама в массовом масштабе. Из-за размеров и сложностей их распознавания, ботнеты могут функционировать в дрейф-режиме (не проявляя активности) на протяжении долгого времени, аккумулируя нужную информацию. Ботнеты используются злоумышленниками для воровства финансовой информации и переводов денег в онлайн-банкинге, в расчетных палатах и системах начисления заработной платы пользователей сети [5, 11]. Например, ботнет «Zeus» проработал в дрейф-режиме более трех лет. Затем он был удаленно активирован злоумышленниками, когда было накоплено достаточное количество сведений, что принесло преступникам 70 миллионов долларов украденных средств [7, 15].

Метод BClus – это подход, использующий поведенческое обнаружение ботнета. BClus создает модель поведения с множеством триггеров известных ботнетов и использует их для обнаружения

аналогичного трафика в сети. Целью метода является кластеризация трафика, отправляемого каждым IP-адресом, и распознавание того, какие кластеры отличаются аномальной активностью, схожей с ботнетом [5, 6].

Обнаружение на стадии вредоносного заражения. Метод BotHunter полезен для обнаружения заражений и для координации передачи данных от ботнета к злоумышленникам. Он состоит из механизма корреляции, целью которого является выявление определенных этапов процесса заражения вредоносным программным обеспечением. Используется адаптивная версия Snort IDS с двумя запатентованными плагинами, которые обладают механизмом обнаружения аномалий статистического сканирования (SCADE) и механизмом обнаружения аномалий статистической нагрузки (SLADE) [15, 17, 18].

Система сбалансированного эффективного обучения (B-ELLA) является новым подходом к обнаружению кибератак на основе практического применения эффективной системы кибербезопасности с непрерывным обучением. Такое решение позволит оперативно решить текущие проблемы в области кибербезопасности, где каждая новая кибератака может рассматриваться как новый объект для анализа и изменения/формирования эталонного значения. Данный подход является расширением платформы ELLA. Он справляется с проблемой корреляции данных на основе уже сформированного набора данных по вредоносным активностям [14, 17].

Заключение. Итак, в связи с большим количеством метаданных, подлежащих анализу, применяемые сегодня методы обнаружения и предотвращения вторжений требуют модификации с использованием различных подходов машинного обучения для обнаружения вредоносных активностей в сетевом трафике или следов ботнетов из набора данных NetFlow в области кибербезопасности. Эталонное решение, предназначенное для обнаружения и предотвращения кибератак, должно анализировать поступающий набор данных любого объема, качества и глубины, и применяя методы искусственного интеллекта классифицировать трафик как легитимный или как вредоносный. В последнем случае ПО должно предоставить перечень угроз, с их классификацией, вероятностями и траекториями атак.

Обработка терабайтов метаданных для обеспечения кибербезопасности в вычислительном отношении является весьма трудоемкой. Во-первых, поступающие на анализ данные очень несбалансированные, то есть большая часть сетевого трафика безвредна, и только его меньшая часть является вредоносной. Это приводит к тому, что сложно сформировать корректную обучающую выборку на данный отрезок времени, с учетом того факта, что вектора кибератак все динамичны и постоянно изменяются. Более того, риск переобучения в процессе обучения высок, поскольку структура сети влияет на способ обучения модели, в то время как требуется независимый от сети алгоритм. Любая сетевая инфраструктура является динамической, постоянно подлежит различным изменениям: связь узлов зависит от системного времени, и связи между серверами могут появляться и исчезать вместе с новыми запросами и новыми пользователями в сетевом сегменте.

Библиографический список

1. Ажмухамедов И. М. Методика формирования обучающего множества при использовании статических антивирусных методов эвристического анализа / И. М. Ажмухамедов, Р. Ю. Демина // Инженерный вестник Дона. – 2015. – № 37 (3). – С. 74.

2. Брумштейн Ю. М. Математические модели и методы решения задач информационного обеспечения, управления и оценки качества работы операторов в сложных человеко-машинных системах / Ю. М. Брумштейн, Д. А. Молимонов // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. – 2019. – № 3. – С. 73–89.

3. Власенко А. В. Анализ уязвимостей и моделирование атак на данные трафика “https”/ А. В. Власенко, П. И. Дзьобан // Вестник Адыгейского государственного университета. Серия 4: Естественно-математические и технические науки. – 2017. – № 2 (201). – С. 109–115.

4. Глухарев М. Л. Программа для автоматизированной верификации ограничений целостности баз данных / М. Л. Глухарев, А. Д. Хомоненко, А. П. Косаренко // Программные продукты и системы. – 2011. – № 1. – С. 91–95.

5. Дзьобан П. И. Свидетельство о государственной регистрации программы для ЭВМ № 2018618364. Программная среда криптографических преобразований / П. И. Дзьобан, А. В. Власенко, Б. В. Леваньков. – Зарегистрировано 11.06.2018.

6. Дзьобан П. И. Свидетельство о государственной регистрации изобретения и патент на изобретение № RU 0002699259 C1. Генератор псевдослучайных последовательностей / П. И. Дзьобан, А. В. Власенко, Б. В. Леваньков. – Зарегистрировано 04.09.2019.

7. Дзьобан П. И. Идентификация DDOS-атак на web-серверы / П. И. Дзьобан, А. В. Власенко // Прикаспийский журнал: управление и высокие технологии. – 2019. – № 1 (45). – С. 181–187.

8. Жук Р. В. Модель нарушителя комплексной системы обеспечения информационной безопасности объектов защиты / Р. В. Жук, А. С. Чебанов, А. В. Власенко, С. Ю. Сазонов // Известия Юго-Западного

государственного университета. Серия: управление, вычислительная техника, информатика, медицинское приборостроение. – 2013. – № 1. – С. 171–173.

9. Ключко В. И. Архитектуры систем поддержки принятия решений / В. И. Ключко, Е. А. Шумков, Власенко, Р. О. Карнизьян // Научный журнал КубГАУ. – 2013. – № 86. – С. 290–299.

10. Ключко В. И. Теория информации и сигналов : учеб. пособие / В. И. Ключко, А. В. Власенко, Н. В. Кушнир, А. В. Кушнир. – Краснодар : КубГТУ, 2011. – С. 132.

11. Корниенко А. А. Методика обнаружения и разрешения конфликтов программных средств защиты от кибератак на железнодорожном транспорте / А. А. Корниенко, М. А. Поляничко // Интеллектуальные технологии на транспорте. – 2015. – № 1. – С. 18–21.

12. Симанков В. С. Методологические аспекты поддержки принятия решений для организации функционирования интеллектуальной системы ситуационного центра / В. С. Симанков, А. Н. Черкасов // Глобальный научный потенциал. – 2015. – № 12 (45). – С. 114–122.

13. Хомоненко А. Д. Динамические модели отладки программ с вероятностным обнаружением ошибок и распределением Эрланга длительности их исправления / А. Д. Хомоненко, А. И. Данилов, А. А. Данилов // Научно-технический вестник информационных технологий, механики и оптики. – 2016. – № 16 (4). – С. 655–662.

14. Шарай В. А. Мониторинг состояния надежности и безопасности структурно-сложных систем на основе логико-числовых моделей/ В. А. Шарай, О. С. Бурангулова, М. В. Андриуца // Известия Южного федерального университета. Технические науки. – 2011. – № 125 (12). – С. 35–49.

15. Antoine Delplace Cyber Attack Detection thanks to Machine Learning Algorithms / Antoine Delplace, Sheryl Hermoso, Kristofer Anandita // University of Queensland / COMS7507: Advanced Security. – P. 3–15.

16. Cisco IOS NetFlow Command Reference. – Режим доступа: https://www.cisco.com/c/en/us/td/docs/ios/netflow/command/reference/nf_book/nf_01.html, свободный. – Заглавие с экрана. – Яз. англ. (дата обращения: 04.01.2020).

17. Kozik Rafal. Machine Learning Techniques for Cyber Attacks Detection / Kozik Rafal and Michal Choras // Image Processing and Communications Challenges. – 2014. – Vol. 5, № 233. – P. 391–398.

18. Tang J. Compressed-Domain Ship Detection on Spaceborne Optical Image Using Deep Neural Network and Extreme Learning Machine / J. Tang, C. Deng, G.-B. Huang & B. Zhao // IEEE Transactions on Geoscience and Remote Sensing. – 2015. – № 53 (3). – P. 1174–1185.

19. Rehák Martin. CAMNEP: Agent-Based Network Intrusion Detection System / Rehák Martin, Michal Pechoucek, Pavel Čeleda and Pavel Minarík. – 2008.

References

1. Azhmukhamedov I. M., Demina R. Yu. Metodika formirovaniya obuchayushchego mnozhestva pri ispolzovanii staticheskikh antivirusnykh metodov evristicheskogo analiza [The methodology for the formation of the training set using static antiviral methods of heuristic analysis]. *Inzhenernyy vestnik Dona* [Engineering Bulletin of the Don], 2015, no. 37 (3), p. 74.

2. Brumshteyn Yu. M., Malimono D. A. Matematicheskie modeli i metody resheniya zadach informatsionnogo obespecheniya, upravleniya i otsenki kachestva raboty operatorov v slozhnykh cheloveko-mashinnykh sistemakh [Mathematical models and methods for solving the problems of information support, management and quality assessment of operators in complex human-machine systems]. *Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: Upravlenie, vychislitelnaya tekhnika i informatika* [Bulletin of Astrakhan State Technical University. Series: Management, computer engineering and informatics], 2019, no. 3, pp. 73–89.

3. Vlasenko A. V., Dzoban P. I. Analiz uyazvimostey i modelirovanie atak na dannye trafika “https” [Vulnerability analysis and attack modeling for “https” traffic data]. *Vestnik Adygeyskogo gosudarstvennogo universiteta. Seriya 4: Estestvenno-matematicheskie i tekhnicheskie nauki* [Vestnik of Adygea State University. Series 4: Natural-mathematical and technical sciences], 2017, no. 2 (201), pp. 109–115.

4. Glukharev M. L., Khomonenko A. D., Kosarenko A. P. Programma dlya avtomatizirovannoy verifikatsii ogranicheniy tselostnosti baz dannykh [Program for the automated verification of database integrity constraints]. *Programmnye produkty i sistemy* [Software products and systems], 2011, no. 1, pp. 91–95.

5. Dzoban P. I., Vlasenko A. V., Ivankov B. V. *Svidetelstvo o gosudarstvennoy registratsii programmy dlya EVM. Programmnaya sreda kriptograficheskikh preobrazovaniy* [The certificate of state registration of computer programs. Software environment for cryptographic transformations], no. 2018618364, registered 11.06.2018.

6. Dzoban P. I., Vlasenko A. V., Ivankov B. V. *Svidetelstvo o gosudarstvennoy registratsii izobreteniya i patent na izobretenie. Generator psevdosluchaynykh posledovatelnostey* [The certificate on the state registration of the invention and the patent for the invention. Pseudorandom sequence generator], no. RU0002699259 C1, registered 04.09.2019.

7. Dzoban P. I., Vlasenko A. V. Identifikatsiya DDOS-atak na web-servery [Identification of DDOS attacks on web servers]. *Prikaspiyskiy zhurnal: upravlenie i vysokie tekhnologii* [Caspian Journal: Management and High Technologies], 2019, no. 1 (45), pp. 181–187.

8. Zhuk R. V., Chebanov A. S., Vlasenko A. V., Sazonov S. Yu. Model narushitelya kompleksnoy sistemy obespecheniya informatsionnoy bezopasnosti obektov zashhity [A model of an intruder of an integrated system for ensuring information security of objects of protection]. *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta. Seriya: upravlenie, vychislitelnaya tekhnika, informatika, meditsinskoe priborostroenie* [Proceedings of the South-western State University. Series: control, computer engineering, informatics, medical instrument making], 2013, no. 1, pp. 171–173.

9. Klyuchko V. I., Shumkov E. A., Vlasenko A. V., Kernizan R. O. Arkhitektury sistem podderzhki prinyatiya resheniy [Architecture of decision support systems]. *Nauchnyy zhurnal Kubanskogo gosudarstvennogo agrarnogo universiteta* [Scientific Journal of the Kuban State Agrarian University], 2013, no. 86, pp. 290–299.

10. Klyuchko V. I., Vlasenko A. V., Kushnir N. V., Kushnir A. V. *Teoriya informatsii i signalov: uchebnoe posobie* [Theory of information and signals: textbook]. Krasnodar, Kuban State Technical University], 2011, p. 132.

11. Kornienko A. A., Polyanichko M. A. Metodika obnaruzheniya i razresheniya konfliktov programmnykh sredstv zashhity ot kiberatak na zheleznodorozhnom transporte [A technique for detecting and resolving conflicts of software protection against cyber attacks in railway transport]. *Intellektualnye tekhnologii na transporte* [Intelligent Technologies in Transport], 2015, no. 1, pp. 18–21.

12. Simankov V. S., Cherkasov A. N. Metodologicheskie aspekty podderzhki prinyatiya resheniy dlya organizatsii funktsionirovaniya intellektualnoj sistemy situatsionnogo tsentra [Methodological aspects of decision support for organizing the functioning of the intellectual system of a situational center]. *Globalnyy nauchnyy potentsial* [Global Scientific Potential], 2015, no. 12 (45), pp. 114–122.

13. Khomonenko A. D., Danilov A. I., Danilov A. A. Dinamicheskie modeli otladki programm s veroyatnostnym obnaruzheniem oshibok i raspredeleniem Erlanga dlitelnosti ikh ispravleniya [Dynamic models of program debugging with probabilistic error detection and Erlang distribution of the duration of their correction]. *Nauchno-tekhnicheskiy vestnik informatsionnykh tekhnologiy, mekhaniki i optiki* [Scientific and Technical Journal of Information Technologies, Mechanics and Optics], 2016, no. 16 (4), pp. 655–662.

14. Shari V. A., Burangulova O. S., Andriutsa M. V. Monitoring sostoyaniya nadezhnosti i bezopasnosti strukturno-slozhnykh sistem na osnove logiko-chislovykh modeley [Monitoring the state of reliability and safety of structurally complex systems based on logical-numerical models]. *Izvestiya Yuzhnogo federalnogo universiteta. Tekhnicheskie nauki* [News of Southern Federal University. Engineering], 2011, no. 125 (12), pp. 35–49.

15. Antoine Delplace, Sheryl Hermoso, Kristofer Anandita. Cyber Attack Detection thanks to Machine Learning Algorithms. *University of Queensland / COMS7507: Advanced Security*, pp. 3–15.

16. Cisco IOS NetFlow Command Reference. Available at: https://www.cisco.com/c/en/us/td/docs /ios/netflow/command/reference/nf_book/nf_01.html (accessed 04.01.2020).

17. Rafal Kozik and Michal Choras. Machine Learning Techniques for Cyber Attacks Detection. *Image Processing and Communications Challenges*, 2014, vol. 5, no. 233, pp. 391–398.

18. Tang J., Deng C., Huang G.-B. & Zhao B. Compressed-Domain Ship Detection on Spaceborne Optical Image Using Deep Neural Network and Extreme Learning Machine. *IEEE Transactions on Geoscience and Remote Sensing*, 2015, no. 53 (3), pp. 1174–1185.

19. Rehák, Martin, Michal Pechoucek, Pavel Čeleda and Pavel Minarík. *CAMNEP: Agent-Based Network Intrusion Detection System*, 2008.

DOI 10.21672/2074-1707.2020.49.4.155-161
УДК 004.77

СОВРЕМЕННЫЕ МЕТОДЫ АТАК ДЕАНОНИМИЗАЦИИ НА СЕТЬ TOR

Статья поступила в редакцию 05.12.2019, в окончательном варианте – 11.03.2020.

Новосельцева Алёна Вячеславовна, Краснодарский университет Министерства внутренних дел Российской Федерации, 350072, Российская Федерация, г. Краснодар, ул. Ярославская, 128,
курсант, e-mail: AlenaNov98@mail.ru

Клюев Станислав Геннадиевич, Краснодарский университет Министерства внутренних дел Российской Федерации, 350072, Российская Федерация, г. Краснодар, ул. Ярославская, 128,
кандидат технических наук, начальник кафедры информационной безопасности, e-mail: s.g.klyuev@mail.ru

В данной статье представлены современные методы атак деанонимизации на анонимную сеть Tor, а также предлагается их классификация. Изучены принципы взаимодействия узлов и построение цепочки луковой маршрутизации сети Tor. Рассмотрены более подробно атаки на клиентскую сторону сети (Raptor-атака, Torben-атака), атаки на сервер (атака с пометкой ячеек, Off-path MitM) и атаки на канал (Timing-атака, CellFlood DoS-attack). Также рассмотрены атаки в зависимости от воздействия на перехватываемый трафик (активные атаки, при которых происходит модификация трафика, и пассивные атаки, при которых трафик просто перехватывается и анализируется, но не модифицируется). Приведены примеры (прецеденты) некоторых атак и последствия. Сделан вывод о высокой значимости наличия больших вычислительных мощностей и ресурсов в осуществлении всех видов атак на Tor-сети.

Ключевые слова: анонимные сети, CellFlood DoS-атака, Tor, анализ трафика, деанонимизация, луковая маршрутизация, коррумпированные узлы, даркнет