DOI: 10.15514/ISPRAS-2021-33(5)-5

Методика сбора обучающего набора данных для модели обнаружения компьютерных атак

[1,2] *А.И. Гетьман, ORCID: 0000-0002-6562-9008 <ever@ispras.ru>*
[3] *М.Н. Горюнов, ORCID: 0000-0003-0284-690X <max.gor@mail.ru>*
[3] *А.Г. Мацкевич, ORCID: 0000-0001-9557-3765 <mag3d.78@gmail.com>*
[3] *Д.А. Рыболовлев, ORCID: 0000-0003-4524-655X <dmitrij-rybolovlev@yandex.ru>*
[1] *Институт системного программирования им. В.П. Иванникова РАН, 109004, Россия, г. Москва, ул. А. Солженицына, д. 25*
[2] *Национальный исследовательский университет «Высшая школа экономики», 101978, Россия, г. Москва, ул. Мясницкая, д. 20*
[3] *Академия ФСО России 302015, Россия, г. Орел, ул. Приборостроительная, д. 35*

Аннотация. В работе рассмотрены вопросы обучения моделей обнаружения компьютерных атак, основанных на применении методов машинного обучения. Последовательно представлены результаты анализа общедоступных обучающих наборов данных и инструментов анализа сетевого трафика и выделения признаков сетевых сессий. Отмечены недостатки существующих инструментов и возможные ошибки в формируемых с их помощью наборах данных. Сделан вывод о необходимости сбора собственных обучающих данных в условиях отсутствия гарантий достоверности общедоступных наборов данных и ограниченного применения предобученных моделей в сетях с характеристиками, отличными от характеристик сети, в которой производился сбор обучающего трафика. Предложен практический подход к формированию данных обучения для моделей обнаружения компьютерных атак. Произведена апробация предлагаемых решений с целью оценки качества обучения модели на собранных данных и качества обнаружения атак в условиях реальной сетевой инфраструктуры.

Ключевые слова: информационная безопасность; система обнаружения атак; машинное обучение; набор данных; перенос обучения; случайный лес; сетевой трафик; компьютерная атака

Для цитирования: Гетьман А.И., Горюнов М.Н., Мацкевич А.Г., Рыболовлев Д.А. Методика сбора обучающего набора данных для модели обнаружения компьютерных атак. Труды ИСП РАН, том 33, вып. 5, 2021 г., стр. 83-104. DOI: 10.15514/ISPRAS–2021–33(5)–5

Methodology for Collecting a Training Dataset for an Intrusion Detection Model

[1,2] *A.I. Getman, ORCID: 0000-0002-6562-9008<ever@ispras.ru>*
[3] *M.N. Goryunov, ORCID: 0000-0003-0284-690X <max.gor@mail.ru>*
[3] *A.G. Matskevich, ORCID: 0000-0001-9557-3765 <mag3d.78@gmail.com>*
[3] *D.A. Rybolovlev, ORCID: 0000-0003-4524-655X <dmitrij-rybolovlev@yandex.ru>*
[1] *Ivannikov Institute for System Programming of the Russian Academy of Sciences, 25, Alexander Solzhenitsyn st., Moscow, 109004, Russia*
[2] *HSE University 20, Myasnitskaya Ulitsa, Moscow, 101978, Russia*
[3] *The Academy of Federal Security Guard Service of the Russian Federation, 35, Priborostroitelnaya st., Oryol, 302015, Russia*

Abstract. The paper discusses the issues of training models for detecting computer attacks based on the use of machine learning methods. The results of the analysis of publicly available training datasets and tools for analyzing network traffic and identifying features of network sessions are presented sequentially. The drawbacks of existing tools and possible errors in the datasets formed with their help are noted. It is concluded that it is necessary to collect own training data in the absence of guarantees of the public datasets reliability and the limited use of pre-trained models in networks with characteristics that differ from the characteristics of the network in which the training traffic was collected. A practical approach to generating training data for computer attack detection models is proposed. The proposed solutions have been tested to evaluate the quality of model training on the collected data and the quality of attack detection in conditions of real network infrastructure.

Keywords: information security; network intrusion detection system; machine learning; dataset; transfer learning; random forest; network traffic; computer attack

For citation: Getman A.I., Goryunov M.N., Matskevich A.G., Rybolovlev D.A. Methodology for Collecting a Training Dataset for an Intrusion Detection Model. Trudy ISP RAN/Proc. ISP RAS, vol. 33, issue 5, 2021, pp. 83-104 (in Russian). DOI: 10.15514/ISPRAS–2021–33(5)–5

1. Введение

Развитие современного общества неразрывно связано с применением широкого спектра информационных технологий: поисковых систем, веб-сервисов, служб обмена файлами, сервисов онлайн-платежей и многих других, что неизбежно сопровождается ростом числа угроз информационной безопасности. Одной из угроз, представляющих наибольшую опасность, является угроза осуществления компьютерных атак на информационные сервисы. Данный вид воздействия, как правило, приводит к нарушению корректного функционирования сервисов, раскрытию конфиденциальной информации пользователей, финансовым потерям и другим негативным последствиям.

Для обнаружения компьютерных атак в основном используются сигнатурные анализаторы сетевого трафика, эффективность которых ограничена полнотой базы решающих правил обнаружения известных информационных воздействий. Очевидным недостатком такого подхода является практически нулевая вероятность обнаружения неизвестных или модифицированных сетевых атак, а также невозможность анализа зашифрованного трафика (например, защищенного с помощью протокола TLS), что подтверждает необходимость разработки новых эвристических алгоритмов классификации сетевого трафика.

Перспективным подходом к созданию несигнатурных методов обнаружения компьютерных атак является применение технологий машинного обучения. Вместе с тем, разработка эвристического анализатора сетевого трафика на основе машинного обучения требует решения ряда сложных задач:

- выбора признакового пространства для адекватного описания того или иного сетевого

трафика;

- формирования обучающего набора данных;
- выбора и обучения модели машинного обучения;
- разработки алгоритма классификации трафика, оценки его эффективности;
- практической реализации, апробации и внедрения алгоритма.

Решению одной из вышеперечисленных задач – задачи формирования обучающего набора данных – и посвящена данная статья. Цель настоящего исследования состоит в разработке методики сбора данных для обучения модели обнаружения сетевых компьютерных атак. Для достижения сформулированной цели необходимо решить следующие задачи:

- провести анализ существующих общедоступных наборов данных, предназначенных для обучения эвристических алгоритмов обнаружения компьютерных атак;
- выявить особенности функционирования инструментов анализа сетевого трафика и выделения признаков сетевых сессий;
- выработать требования, предъявляемые к создаваемым наборам данных для обучения;
- разработать стенды для моделирования компьютерных атак и фонового трафика пользователей;
- предложить методику сбора обучающего набора данных и провести ее апробацию в отношении выбранного объекта защиты с последующей оценкой качества обнаружения атак.

Новизна работы заключается в применении системного подхода к решению вопроса обучения модели обнаружения компьютерных атак в условиях отсутствия гарантий достоверности общедоступных обучающих наборов данных, а также вариативности характеристик защищаемой сети и возникающих по этой причине сложностей практического применения предобученных моделей.

2. Анализ релевантных работ

Вопросы применения методов машинного обучения для обнаружения компьютерных атак и возникающие при этом сложности в получении аккуратно размеченных данных для обучения активно обсуждаются в последние годы. По указанной тематике опубликовано достаточное количество работ, которые могут служить основой дальнейших исследований.

В статье [1] сформулированы перспективные направления исследований в области кибербезопасности, среди которых выделена задача развития практики применения методов искусственного интеллекта и машинного обучения. Отмечается важность выбора признакового пространства, разметки данных для обучения. Приводится список наиболее используемых общедоступных наборов данных для задач кибербезопасности, однако отсутствуют сведения о практической применимости моделей, предобученных на таких данных.

В работе [2] подчеркивается, что в условиях постоянного появления новых типов компьютерных атак актуальной является задача разработки наборов данных, содержащих современные типы атак. Такая задача подразумевает наличие у разработчика соответствующих экспертных знаний в области построения распределенных тестовых стендов, современных сетевых технологий, моделирования компьютерных атак и др. Проведен анализ общедоступных наборов данных, формализованы требования к создаваемым наборам данных для обучения: разнородности представленных атак, наличия полной конфигурации сети, полного сетевого взаимодействия, разнородности протоколов и др. Сформирован публичный набор данных CICIDS2017, который впоследствии получил широкое распространение в исследовательских проектах. Вместе с тем в статье не приводятся результаты апробации предлагаемых решений в реальной сети и не оценивается возможное снижение качества обнаружения атак по причине различия характеристик защищаемой сети и сети, в которой производился сбор обучающего трафика.

В исследовании [3] представлен обзор 34 общедоступных наборов данных с указанием их отличительных особенностей, используемых сценариев атак, имеющихся недостатков. Отдельно отмечается недостаточная репрезентативность существующих наборов данных для обучения, что, по мнению авторов, является одним из основных препятствий при построении систем обнаружения атак. Предложена методология оценки применимости наборов данных к различным задачам информационной безопасности. Среди проанализированных наборов данных для использования в практических задачах авторы рекомендуют наборы CICIDS 2017, CIDDS-001, UGR-16 и UNSW-NB15. Вместе с тем не приводятся практические предложения по устранению известных недостатков этих наборов данных, а представленные рекомендации по сбору собственных наборов носят общий характер.

В работе [4] рассматриваются задача классификации сетевого трафика и возможности применения методов машинного обучения для ее решения. Исследуется вопрос формирования признакового пространства, обсуждаются существующие проблемы получения данных для обучения и основные компромиссы в этом вопросе. Перечисляются часто используемые общедоступные наборы данных и их характеристики. Авторы отмечают, что одним из вариантов получения адекватных данных для обучения является формирование своего собственного набора данных. Однако в статье не представлена система требований к создаваемым наборам данных.

В статье [5] рассматривается один из аспектов проблемы transfer learning в исследуемой предметной области – изменение качества работы классификатора сетевого трафика, предобученного в сети с характеристиками, отличными от характеристик защищаемой сети. Продемонстрировано снижение качества обнаружения сетевых атак при переносе предобученной модели в другую сеть. Проведены эксперименты по определению дополнительного объема данных дообучения, достаточного для восстановления исходного качества предобученного классификатора. Вместе с тем в качестве данных для обучения в работе используются публичные наборы NIMS2018 и UNB2015 с соответствующими архитектурами сетей, и не представлены практические предложения по дообучению моделей обнаружения атак в сетях с отличными характеристиками.

В работе [6] представлены результаты анализа актуальных наборов данных для обучения систем обнаружения сетевых атак. Авторы рекомендуют оценивать качество обнаружения атак, применяя несколько наборов данных, чтобы избежать переобучения. Для использования в практических задачах авторы рекомендуют наборы UNSW-NB15, CIDDS-001, CICIDS 2017 и CSE-CIC-IDS 2018, указывая при этом их недостатки.

В исследовании [7] подчеркивается важность этапа формирования признакового пространства, сбора и разметки данных. На примере широко используемого, но устаревшего набора данных KDD Cup 1999, отмечаются недостатки существующих публичных наборов данных. Однако для оценки предлагаемых решений авторы также используют публичный набор данных NGIDS-DS, не подтверждая возможность применения предобученной модели в защищаемой сети с характеристиками, отличными от характеристик сети, в которой производился сбор трафика.

Достаточная интерпретируемость широко распространенных моделей машинного обучения позволяет применить к их практическим реализациям известный в информатике принцип «garbage in, garbage out» и подчеркнуть при этом важность использования при обучении адекватных, аккуратно размеченных данных. В отмеченных выше работах подробно рассматриваются вопросы применения моделей машинного обучения для решения задачи классификации сетевого трафика в различных постановках, подчеркивается значимость этапа сбора и подготовки обучающего набора данных в виду прямой зависимости качества обнаружения финальной модели от качества данных для обучения. Однако опубликованные результаты носят недостаточно полный и системный характер с точки зрения формализации требований к создаваемым наборам данных; практической реализации этапов сбора и

разметки; апробации моделей, обученных на собственных наборах данных; встраивания разрабатываемых программных модулей в действующие системы и комплексы и др.

Настоящая работа является логическим продолжением исследования [8], в котором по результатам апробации синтезированной модели обнаружения атак на реальных данных показана ее состоятельность только при условии обучения на данных, собранных в конкретной защищаемой сети, в виду зависимости ряда значимых признаков от физической структуры сети и настроек используемого оборудования. В качестве одного из основных выводов статьи отмечена необходимость предварительного обучения моделей на наборах данных, полученных на основе анализа сетевого трафика в защищаемой сети (аналоге с соответствующими характеристиками) и содержащем признаки классифицируемых компьютерных атак.

Решаемая в данном исследовании основная задача заключается в разработке практического подхода к формированию данных обучения для моделей обнаружения компьютерных атак, основанных на применении методов машинного обучения. Отдельной важной подзадачей при этом является апробация предлагаемых решений с целью оценки качества обучения модели на собранных данных и качества обнаружения атак в условиях реальной сетевой инфраструктуры.

3. Общедоступные наборы данных

Для обучения систем обнаружения компьютерных атак уровня сети, основанных на применении несигнатурных методов обнаружения, применяются специализированные наборы размеченных данных. К наиболее известным и опубликованным в открытых источниках наборам данных можно отнести следующие: DARPA1998, KDD Cup 1999, Kyoto 2006, NSL-KDD 2009, ISCX 2012, CTU-13, UNSW-NB15, CIDDS-001, UGR-16, CICIDS 2017, CICIDS 2018 и другие [3]. Данные наборы используются подавляющим большинством исследователей для апробации исследуемых алгоритмов обнаружения. Для описания наборов данных воспользуемся следующими характеристиками набора данных.

- Число признаков в наборе данных. В число признаков входят: признаки, характеризующие общую информацию о соединении / потоке (например, время начала соединения; IP адрес источника атаки; порт источника атаки и т.п.), информативные признаки (например, длительность соединения, число переданных / принятых байт и т.п.), а также признаки, которые используются для описания атаки или нормального сетевого соединения (например, метка класса трафика, описание атаки, реакция антивирусного средства на соединение).
- Природа информативных признаков. В табл. 1 используются следующие обозначения:
  ПСС – признаки, характеризующие сетевое соединение (например, длительность сетевойисессии);
  ПНП – признаки, характеризующие направление передачи данных (например, число байт переданных в направлении сервера; среднее время между сетевыми пакетами в направлениииклиента);
  ППУ – признаки, характеризующие операции, выполняемые на прикладном уровне (например, успех операции удаленной аутентификации пользователя, число операций с файлами в данном соединении и т.п.);
- Инструмент, который был использован для выделения признаков из сетевого трафика.
- Типы сетевых атак в наборе данных.

Табл. 1. Описание наборов данных, предназначенных для обучения несигнатурных систем обнаружения компьютерных атак уровня сети
Table 1. Description of datasets designed for training non-signature network intrusion detection systems

№	Набор данных	Число признаков в наборе данных	Природа информативных признаков	Инструмент для выделения признаков	Типы сетевых атак в наборе данных
1	DARPA 1998 [9]	10	ПСС	Нет сведений	DoS, Remote to User, User-to-Root, Surveillance/probing attacks
2	KDD Cup 1999 [10]	42	ПСС, ППУ	MADAM ID	DoS, Remote to User, User-to-Root, Surveillance/probing attacks
3	Kyoto 2006+ [11]	24	ПСС	Нет сведений	Различные атаки на honeypots (backscatter, DoS, exploits, malware, port scans, shellcode)
4	NSL-KDD 2009 (создан на основе KDD Cup 1999) [12]	42	ПСС, ППУ	MADAM ID	DoS, Remote to User, User-to-Root, Surveillance/probing attacks
5	ISCX 2012 [13]	19	ПСС, ПНП	Не известно	Brute Force SSH, HTTP DoS, DDoS using an IRC Botnet, Infiltrating the network from inside
6	CTU-13 [14]	33	ПСС, ПНП	Argus	botnets (Menti, Murlo, Neris, NSIS, Rbot, Sogou, Virut)
7	UNSW-NB15 [15]	45	ПСС, ПНП, ППУ	Argus, Bro-IDS	Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconaissance, Shellcode, Worms
8	CIDDS-001 [16]	14	ПСС, ПНП	NetFlow	Port scanning, DoS, BruteForce, Ping Scan
9	UGR-16 [17]	132 (Feature as Counter), 13 (в CSV файле)	ПСС, ПНП	FCParser	low- and high-rate DoS, Port scanning, UDP port scanning, SSH scanning, Botnet, Spam
10	CICIDS 2017 [18]	85	ПСП, ПСС, ПНП	CICFlowMeter	DoS Hulk, PortScan, DDoS, DoS GoldenEye, FTP-Patator, SSH-Patator, DoS slowloris, DoS Slowhttptest, Bot, Infiltration, Heartbleed, Web Attack – Brute Force, Web Attack – XSS, Web Attack – SQL Injection
11	CICIDS 2018 и другие наборы данных, созданные в	80	ПСП, ПСС, ПНП	CICFlowMeter-v3	Brute Force, Heartbleed, Botnet, DoS, DDoS, Web attacks, Infiltration of the network from inside

	Canadian Institute for Cybersecurity [19]				

Анализ публикаций с результатами оценки качества различных классификаторов сетевого трафика, обучение которых осуществлялось на представленных выше наборах данных, показал следующее [6 , 8]:

- для построения систем обнаружения компьютерных атак уровня сети использование признаков, характеризующих операции, которые выполняются на прикладном уровне, невозможно в силу того, что в настоящее время практически весь сетевой трафик является зашифрованным (используются протоколы TLS / SSL и IPSEC);
- в большинстве приведенных выше наборов данных используются признаки, характеризующие только сетевое соединение в целом (адресная информация, длительность соединения, число переданных байт), а также дополнительные признаки, полученные при анализе данных прикладного уровня;
- для улучшения качества классификаторов сетевого трафика необходимо использование признакового пространства, которое включает в себя множество признаков, характеризующих сетевое соединение в целом, каждое направление в отдельности, а также особенности конкретного соединения (потока) на транспортном уровне.

В настоящем исследовании за основу взяты признаки, представленные в наборах данных CICIDS 2017 и CICIDS 2018.

4. Инструменты выделения признаков сетевых сессий

Качество формируемого набора данных непосредственно зависит от качества инструментов, используемых на этапе сбора и анализа сетевого трафика, выделения признаков сетевых сессий. В ходе исследования публичных наборов данных отдельное внимание уделялось анализу используемых авторами инструментов.

Инструменты выделения признаков сетевых сессий обычно решают ряд задач и предоставляют следующие возможности.

1) Анализ сетевого трафика, выделение сетевых сессий (обычно применяются настраиваемые таймауты ожидания, активности и др. протокола TCP, от которых зависит момент логического завершения сетевой сессии инструментом).
2) Обработка сетевых сессий и выделение признаков.
3) Обработка трафика реального времени и предварительно сохраненного трафика (PCAP - файлы и др.).
4) Сохранение сформированных наборов данных в одном из форматов экспорта (CSV, XML, TXT и др.).

Наиболее распространенными инструментами анализа трафика и выделения признаков сетевых сессий являются следующие (табл. 2).

- Argus (Audit Record Generation and Utilization System) [20] – свободно распространяемый инструмент сетевого аудита, разработан одним из первых в своем классе. Позволяет обрабатывать сетевой трафик с выделением широкого спектра признаков сетевых сессий (всего 125), анонимизировать трафик, обогащать вектора признаков сессий дополнительными данными (например, данными о геолокации и др.). Поддерживает платформы Mac OS X, Linux, Unix, Windows; протоколы SMTP, POP3, HTTP, NNTP, ICMP, SNMP, FTP, Telnet, SSH, Gopher, NFS, DNS, Radius, IAX2, SIP, SunRPC, Whois, Rwhois, LPD, NTP; поддерживает IPv4 и IPv6. Предоставляет возможность расширения функционала с помощью пользовательских скриптов. Использовался при сборе наборов данных CTU-13 и UNSW-NB15.

- CICFlowMeter (первая версия называлась ISCXFlowMeter) [21] – свободно распространяемый анализатор сетевого трафика, который позволяет выделить сетевые сессии и для каждой сессии сформировать вектор признаков в формате CSV (всего 80). Разработан на языках программирования Java/C. Использовался при сборе наборов данных CICIDS 2017, CICAAGM 2017, CICAndMal 2017, CICIDS 2018, CICDDoS 2019 и др. Разработан в Canadian Institute for Cybersecurity.
- NFStream [22] – свободно распространяемый Python-фреймворк, предназначенный для высокопроизводительного анализа сетевого трафика. Расширяется с помощью системы плагинов, что позволяет добавлять функционал выделения новых признаков и встраивать модели машинного обучения в общий тракт обработки трафика.
- FCParser [23] – парсер потоков данных, реализующий методологию конструирования признаков «feature as a counter» (FaaC): каждый признак представляет собой счётчик числа наблюдений определённого события в заданный промежуток времени. Использовался при сборе набора данных UGR-16. Разработан на языке Python.
- MADAM ID (Mining Audit Data for Automated Models for Intrusion Detection) – сетевая система обнаружения атак, использующая интеллектуальный анализ данных для обнаружения аномалий. Устаревший инструмент, использовался при сборе одного из первых наборов данных – KDD Cup 1999.

Для проверки корректности разметки набора данных CICIDS 2017 авторами настоящего исследования были воспроизведены эксперименты [2, 24]. По результатам обработки исходных PCAP-файлов с захваченными пакетами набора данных CICIDS 2017 собственным инструментом выделения признаков сетевых сессий были обнаружены расхождения полученных данных и данных набора CICIDS 2017. Дальнейшие исследования показали наличие следующих ошибок в исходном коде инструмента CICFlowMeter, который использовался при сборе и формировании набора данных CICIDS 2017.

1) Ошибки при расчетах значений признаков «Packet Length Mean» (средняя длина полезной нагрузки в пакетах всего потока), «Packet Length Std», «Packet Length Variance» и «Average Packet Size». Ошибка связана с двойным учетом первого пакета в структуре данных со статистикой длин пакетов сетевой сессии.
2) Некорректное завершение сессий – при первом появлении в сессии пакета с флагом FIN. Возможные следующие пакеты с флагами FIN ACK и ACK, фактически относящиеся к первой незавершенной сессии, попадают во вторую сессию. Это приводит к появлению в наборе данных большого количества сессий, состоящих из одного-двух пакетов, с нулевой длиной полезной нагрузки.
3) Дублирование признака «Fwd Header Length».
4) Ошибка при расчете длины TCP пакета – длина дополнения (padding) фрейма Ethernet прибавляется к длине TCP пакета.
5) Признаки «Packet Length Mean» и «Average Packet Size» должны иметь одинаковое значение, однако по причине логической ошибки имеют различные значения. Ошибка состоит в том, что при завершении сессии по таймеру граничный пакет попадает в статистику длин пакетов, а счетчик количества пакетов (знаменатель в выражении для расчета значения признаков «Packet Length Mean» и «Average Packet Size») увеличивается только для одного из признаков.

Отчет об ошибках отправлен авторам набора данных в марте 2021 года, в том числе в виде issue в репозиторий с исходным кодом инструмента CICFlowMeter (https://github.com/ahlashkari/CICFlowMeter/issues/111), однако на момент подготовки публикации настоящего исследования (ноябрь 2021 года) ошибки не исправлены.

Указанные обстоятельства в отношении одного из наиболее цитируемых в мире наборов данных подтверждают необходимость как обязательной верификации используемых данных

для обучения моделей машинного обучения, так и предъявления соответствующего требования к создаваемым общедоступным наборам данных: возможность верификации публикуемых данных.

Наличие ошибок в общедоступных инструментах выделения признаков является причиной возможных скрытых ошибок в создаваемых с их помощью наборах данных. В таких условиях оправданными являются разработка своих собственных инструментов с последующим сравнением результатов их работы на публичных наборах данных с результатами общедоступных инструментов.

Табл. 2. Общедоступные инструменты выделения признаков сетевых сессий
Table 2. Public generators and analyzers of network traffic flows

№	Название инструмента, лицензия	Поддерживаемые платформы	Язык программирования	Количество выделяемых признаков	Наборы данных, сформированные с использованием инструмента
1	Argus, GPLv2	Linux, Solaris, BSD, OS X, IRIX, AIX, Windows, OpenWrt	C	125	CTU-13, UNSW-NB15
2	CICFlowMeter, производная от MIT	Сведения отсутствуют	Java/C	80	CICIDS 2017, CICAAGM 2017, CICAndMal 2017, CICIDS 2018, CICDDoS 2019 и др.
3	NFStreams, LGPL-3.0	Linux, MacOS, ARM	Python	48	Сведения отсутствуют
4	FCParser, сведения отсутствуют, исходный код открыт	Unix	Python	Переменное количество (методология FaaC)	UGR-16
5	MADAM ID, сведения отсутствуют	Сведения отсутствуют	Сведения отсутствуют	Сведения отсутствуют	KDD Cup 1999

5. Методика сбора обучающего набора данных

5.1 Требования к наборам данных

Большинство публичных наборов данных для обучения систем обнаружения компьютерных атак были разработаны с главной целью – предоставить исследователям возможность сравнения различных методов обнаружения в одинаковых условиях. На практике часто возникает и другая задача: оценить качество синтезированной модели машинного обучения на нескольких наборах данных. И в первом, и во втором случае исследователю предстоит обосновать выбор используемых при обучении данных. При сравнении характеристик различных наборов данных важным является вопрос формализации единых требований к ним.

Основополагающими требованиями к публикуемым наборам данных можно считать перечисленные в исследовании [25]:

- возможность однозначной идентификации – набор данных должен быть уникальным, содержать подробное описание, быть проиндексированным в соответствующих поисковых системах;
- доступность – должен быть предоставлен свободный доступ к набору данных по его идентификатору;
- возможность сравнения метаданных – наборы данных должны использовать единые словари метаданных;
- многократное использование – данные должны быть точно описаны совокупностью релевантных атрибутов и соответствовать отраслевым стандартам, должны быть указаны происхождение данных и лицензионные условия их использования.

В статье [26] рассматриваются следующие требования к синтетическим наборам данных для обучения моделей обнаружения атак.

- Полная конфигурация сети. В моделируемой сети должны быть представлены различные устройства: модемы, брандмауэры, коммутаторы, маршрутизаторы, с различными операционными системами.
- Полный трафик. Набор данных должен включать и «чистый» трафик, и трафик компьютерных атак.
- Наличие разметки.
- Полное взаимодействие. При сборе набора данных необходимо моделировать взаимодействие внутри конкретной локальной сети, между несколькими локальными сетями и связь через Интернет.
- Полный захват трафика и сохранение.
- Разнообразие сетевых протоколов. Набор должен включать данные взаимодействия по различным протоколам: HTTP, HTTPS, FTP, SSH, протоколам электронной почты и др.
- Разнообразие атак. При сборе трафика атак должны моделироваться наиболее распространенные атаки, такие как веб-атаки, bruteforce, DoS, DDoS, попытки проникновения, активность ботнетов, сканирование и др.
- Разнородность анализируемых данных. Должен осуществляться захват сетевого трафика и анализ дампа памяти и системных вызовов со всех машин-жертв во время выполнения атак.
- Представление данных. Собранные и размеченные данные следует публиковать в одном из общепринятых форматов представления данных.

В работе [3] предложены дополнительные требования к создаваемым наборам данных для обучения моделей обнаружения компьютерных атак: актуальности и разнородности представленных атак, наличия «чистого» пользовательского трафика и в нём – трафика полезной нагрузки, корректности разметки. Кроме того, при создании наборов данных должен охватываться значительный временной интервал для включения в набор достаточного количества записей, соответствующих атакам и «чистому» трафику. Авторы отмечают, что в метаданных набора должны указываться сведения о его разбиении на отдельные логические блоки данных (если есть), а также сведения о балансе классов объектов выборки.

В условиях наличия возможных скрытых ошибок в разметке данных необходимо предъявлять дополнительное требование к публикуемым наборам данных: возможность их верификации.

5.2 Выбор объекта защиты и класса атак

Качество модели выявления атак напрямую зависит от качества обучающего набора данных, которое тем выше, чем ближе моделируемый объект к реальному. В роли объекта защиты может рассматриваться как корпоративная сеть в целом, так и ее отдельные сервисы. Вместе с тем правильным выглядит подход, при котором в качестве объекта защиты рассматриваются именно конкретные продукты информационных технологий,

функционирующие в исследуемой сети. Например, в роли объектов защиты могут выступать SSH, FTP, TELNET и другие сетевые службы, веб-приложения, включая веб-консоли управления различных устройств сети, служба удаленных рабочих столов и т. д. Сбор обучающего набора данных для выделенных объектов защиты должен осуществляться в реальной сети, в которой они функционируют, или (при отсутствии такой возможности) на близком по структуре, составу и настройкам сетевых устройств макете сети. При этом для каждого объекта защиты перечень реализуемых в отношении них классов атак будет индивидуальным и определяется исходя из его особенностей. Также необходимо отметить, что некоторые классы атак будут характерны только для определенных типов объектов защиты, например, класс веб-атак характерен только для такого типа объектов защиты как веб-приложения.

В представленном исследовании вопрос формирования обучающего набора данных рассматривался на примере класса веб-атак.

5.3 Особенности генерации веб-атак в наборе данных CICIDS 2017

В ходе анализа информации об особенностях генерации веб-атак для набора данных CICIDS 2017 [2] было установлено, что в качестве атакуемого приложения в стенде использовался проект Damn Vulnerable Web Application (DVWA) – «чертовски» уязвимое веб-приложение, в отношении которого моделировалось три вида атак. Информация о количестве веб-атак каждого вида, продолжительности их генерации и используемых для этого средствах представлена в табл. 3.

Табл. 3. Информация о веб-атаках в CICIDS 2017
Table 3. Number of attacks in the CICIDS 2017 dataset

№	Тип атаки	Количество атак	Продолжительность генерации атак	Средство генерации
1	Web Attack – Brute Force	1507	40 мин.	Patator
2	Web Attack – XSS	652	20 мин.	Selenium
3	Web Attack – SQL Injection	21	2 мин.	Selenium

Итоговый набор данных CICIDS 2017 содержит 2180 записей, касающихся веб-атак, что составляет 0,07% от всего датасета. При этом продолжительность генерации атак говорит о том, что в ходе этого процесса не учитывались возможные временные параметры моделирования атак, т.е. задержки, а также другие возможные настройки средств генерации. Кроме того, было установлено, что при моделировании веб-атак использовались два субъекта – источник атаки (205.174.165.73/172.16.0.1) и атакуемое веб-приложение (192.168.10.50/205.174.165.68). То есть в данном датасете моделировалась ограниченная схема реализации XSS-атак (без участия третьей стороны). В сведениях о наборе данных не указано, по какому протоколу осуществлялся доступ к веб-приложению (только на основе анализа PCAP файла можно установить, что использовался протокол HTTP). Также отсутствует информация о типе и настройках используемого веб-сервера.

Проведенные исследования показали, что параметры сетевых соединений при работе с веб-приложениями существенно зависят от типа протокола (HTTP/HTTPS) и от типа веб-сервера (например, Apache/Nginx) и его настроек (например, параметр timeout). В частности, в ходе экспериментов установлено следующее:

- модели, обученные на HTTP трафике, не способны качественно работать на HTTPS трафике и наоборот;
- модели, обученные на трафике с веб-сервером Apache (KeepAliveTimeout 5), не способны качественно работать на трафике с веб-сервером Nginx (keepalive_timeout 65) и наоборот.

Также в ходе проведенных экспериментов было установлено, что при реализации веб-атак с использованием браузера в сетевом потоке будут содержаться как сессии, которые содержат элементы атаки, например, GET/POST-запросы со зловредной нагрузкой, так и сессии, которые относятся к фоновому трафику, например, штатная процедура авторизации или переходы по ссылкам. В этой связи такой трафик должен подвергаться тщательному анализу и разметке. Данная процедура не является сложной и может быть реализована с использованием механизма регулярных выражений. Вместе с тем для HTTPS трафика такая процедура потребует проведения расшифровки трафика, для чего необходимо наличие соответствующих SSL ключей. Такой вопрос в CICIDS 2017 не учитывался.

Фоновый трафик в CICIDS 2017 моделировался на основе использования статистических профилей работы пользователей (B-Profile) [27].

На основе проведенного анализа были сформулированы следующие предложения по улучшению качества формируемого обучающего набора данных для выявления веб-атак:

- расширить перечень атак и используемых средств генерации трафика;
- использовать при моделировании трафика возможности применения временных задержек и других параметров (опций) средств генерации;
- осуществлять разметку трафика, явно содержащего как атаки, так и элементы легитимных действий пользователей;
- расширить объем набора данных;
- учитывать тип используемого протокола реального объекта;
- учитывать тип и настройки веб-сервера реального объекта.

5.4 Испытательный стенд для моделирования веб-атак

В качестве тестового веб-приложения для моделирования атак и фонового трафика было принято решение об использовании DVWA. Для моделирования был определен список из шести видов веб-атак:

- Cross-Site Scripting (XSS);
- Cross Site Request Forgery (CSRF);
- SQL Injection (SQLi);
- Upload a Web Shell to a Web Server (Malware);
- Password Brute Forcing (Brute);
- OS Command Injection (COMMi).

Далее был проведен анализ программных средств, которые могут использоваться для реализации этих атак, а также генерации фонового трафика, и выбраны наиболее подходящие (табл . 4).

Табл. 4. Базовые инструментальные средства моделирования веб-трафика
Table 4. Basic web traffic modeling tools

Тип веб-трафика	Базовые инструментальные средства моделирования
Фоновый трафик	Браузер; Selenium IDE;
Атака: Cross-Site Scripting (XSS)	Браузер; Selenium IDE; Hackapp; Xsser
Атака: Cross Site Request Forgery (CSRF)	Браузер; Selenium IDE; Hackapp
Атака: SQL Injection	Браузер; Selenium IDE; Sqlmap
Атака: Upload a Web Shell to a Web Server	Браузер; Selenium IDE; Weevely
Атака: Password Brute Forcing	Patator
Атака: OS Command Injection	Браузер; Selenium IDE; Commix

Итоговый список используемых для генерации обучающего трафика программных средств включает следующие: Браузер, Selenium IDE, Xsser, Hackapp, Sqlmap, Weevely, Patator, Commix. Все перечисленные средства, за исключением Hackapp, входят в состав дистрибутива ОС Kali Linux. Hackapp – это веб-приложение, которое было специально разработано для моделирования XSS (Reflected) и CSRF атак. Оно включает страницы со скрытыми формами, при посещении которых отправляются запросы от имени авторизованного пользователя атакуемого веб-приложения и тем самым осуществляются атаки соответствующих классов. В итоге, в состав стенда моделирования веб-атак были включены три компонента (рис. 1):

- Attaker – хост генерации и реализации атак;
- Web – хост, на котором установлено атакуемое веб-приложение (DVWA);
- Victim – хост пользователя, в отношении которого реализуются XSS (Reflected) и CSRF атаки.

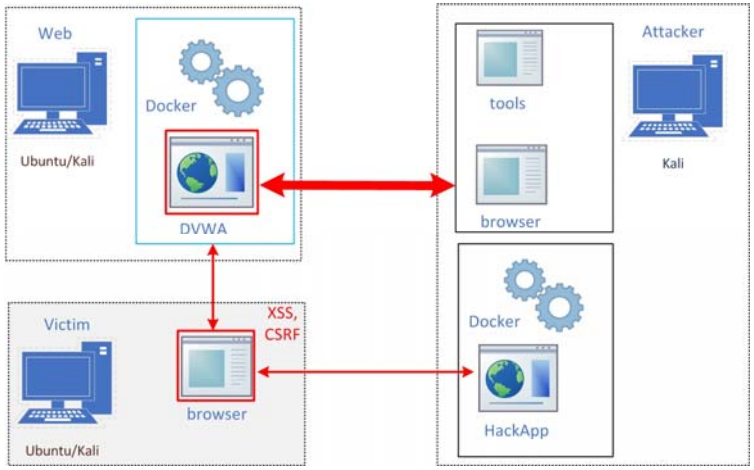

Рис. 1. Обобщенная схема моделирования веб-атак и фонового трафика
Fig. 1. A generalized modeling scheme for web attacks and background traffic

Для ускорения процесса развертывания веб-приложений использовалась технология контейнеризации Docker.

5.4 Сценарии генерации фонового веб-трафика и атак

С учетом сформированного перечня инструментальных средств, анализа их возможностей были подготовлены сценарии генерации десяти типов трафика (табл. 5).

Фоновый трафик, а также некоторые виды атак моделировались с использованием плагина для браузера Selenium IDE (SeIDE), позволяющего создавать и воспроизводить тесты, содержащие последовательность действий пользователя в браузере. Общий алгоритм моделирования трафика на основе SeIDE состоит в следующем.

1) Создается базовый тест моделирования действий пользователя.

2) Базовый тест тиражируется с одновременной вставкой секций пауз между секциями операций теста. Паузы заполняются случайным образом, а при необходимости конкретизации вводимых пользователем данных используется заранее подготовленный пул полезной нагрузки (например, для моделирования атак XSS, SQLi, COMMi).

3) Осуществляется последовательный запуск всех тиражированных тестов. При этом браузер для воспроизведения очередного теста выбирается случайным образом (в исследовании использовались браузеры Chrome и Firefox).

Табл. 5. Типы генерируемого веб-трафика
Table 5. Types of generated web traffic

Класс трафика		Обозначение	Средство генерации	Тип трафика
Benign	Фоновый трафик	Benign	SeIDE	Трафик на основе тестов SeIDE, осуществляющих выполнение в DVWA общих действий, не содержащих атак
		Benignxss	SeIDE	Трафик на основе тестов SeIDE, осуществляющих выполнение на уязвимой странице DVWA, содержащей уязвимость XSS, действий, не содержащих атак
		Benignsqli	SeIDE	Трафик на основе тестов SeIDE, осуществляющих выполнение на уязвимой странице DVWA, содержащей уязвимость SQLi, действий, не содержащих атак
Attack	Cross-Site Scripting (XSS)	XSS	Xsser	Трафик, генерируемый с использованием утилиты Xsser, осуществляющей поиск XSS-уязвимостей в DVWA
			SeIDE	Трафик на основе тестов SeIDE, осуществляющих реализацию атак (внедрение зловредной полезной нагрузки) на уязвимой странице DVWA, содержащей уязвимость XSS
			HackApp	Трафик на основе тестов SeIDE, осуществляющих посещение недоверенного сайта (приложение HackApp), содержащего зловредный код, реализующий XSS-атаки в отношении DVWA
	SQL Injection	SQLi	Sqlmap	Трафик, генерируемый с использованием утилиты Sqlmap, осуществляющей поиск SQLi-уязвимостей в DVWA
			SeIDE	Трафик на основе тестов SeIDE, осуществляющих реализацию атак (внедрение зловредной полезной нагрузки) на уязвимой странице DVWA, содержащей уязвимость SQLi
	Password Brute Forcing	Brute	Patator	Трафик, генерируемый с использованием утилиты Patator, осуществляющей подбор пароля DVWA
	Cross Site Request Forgery (CSRF)	CSRF	HackApp	Трафик на основе тестов SeIDE, осуществляющих посещение недоверенного сайта (приложение HackApp), содержащего зловредный код, реализующий CSRF-атаки в отношении DVWA
	Upload a Web Shell to a Web Server	Malware	Weevely	Трафик, генерируемый с использованием утилиты Weevely, устанавливающей соединение с внедренным в DVWA Shell-кодом (BackDoor.php).

	OS Command Injection	COMMi	Commix	Трафик, генерируемый с использованием утилиты Commix, осуществляющей эксплуатацию уязвимости OS Command Injection в DVWA
			SeIDE	Трафик на основе тестов SeIDE, осуществляющих реализацию атак (внедрение зловредной полезной нагрузки) на уязвимой странице DVWA, содержащей уязвимость OS Command Injection

Для успешной работы специализированных программных средств генерации веб-атак требуется знание авторизационной cookie (для DVWA это PHPSESSID), которая может быть извлечена из хранилища браузера или из информационного обмена с веб-приложением. В представленном исследовании cookie PHPSESSID извлекалась по второму варианту и передавалась в качестве одного из входных параметров для работы скриптов генерации, запускающих соответствующие программные средства (Sqlmap, Xsser, Patator, Weevwly, Commix). Кроме того, при генерации атак применялись соответствующие опции данных программ, определяющие значения задержек, потоков и других параметров, позволяющих вариативно реализовывать зловредные действия.

5.5 Комплекс скриптов генерации веб-трафика и формирования на его основе обучающего набора данных

Для написания всех скриптов использовался язык Bash.

В процессе проведения исследования были выработаны требования к модулям генерации фонового веб-трафика и трафика веб-атак, позволяющие автоматизировать данный процесс.

1) Скрипты генерации фонового веб-трафика и трафика веб-атак должны позволять:
   - генерировать отдельный трафик по выбору;
   - генерировать весь обучающий трафик сразу.
2) Трафик должен генерироваться одним скриптом (TrafficGen.sh), в котором будут определены все задачи по формированию обучающего трафика.
3) Выполнение задач по генерации конкретного вида трафика осуществляется с использованием разработанных скриптов и тестов генерации веб-атак и фонового веб-трафика, запуск которых прописывается в основном скрипте (TrafficGen.sh).
4) Скрипты генерации трафика должны использовать параметры командной строки, позволяющие передавать в них все необходимые для их работы параметры, что обеспечит отсутствие необходимости внесения изменений в сами скрипты.
5) Для скриптов генерации трафика должна быть предусмотрена возможность определения параметров или их части в конфигурационных файлах.

В результате был разработан комплекс скриптов, позволяющих полностью автоматизировать процесс генерации, сбора и обработки заданного обучающего веб-трафика. Обобщенная схема работы комплекса представлена на рис. 2.

Перед запуском основного модуля генерации трафика TrafficGen.sh необходимо:

- определить значения общих параметров, требуемых для работы скриптов (IP адреса, пути к каталогам с ресурсами, настройки SSH соединений, опции выполняемых действий и т.д.), которые прописываются в едином конфигурационном файле TrafficGen.conf;
- уточнить в основном модуле генерации трафика TrafficGen.sh параметры запускаемых вспомогательных модулей (осуществляют подготовительные действия, заключающиеся в настройке приложения Hackapp, выделении PHPSESSID, тиражировании тестов и вставке в них пауз) и модулей генерации трафика (выполняют основной функционал по формированию обучающего набора данных).

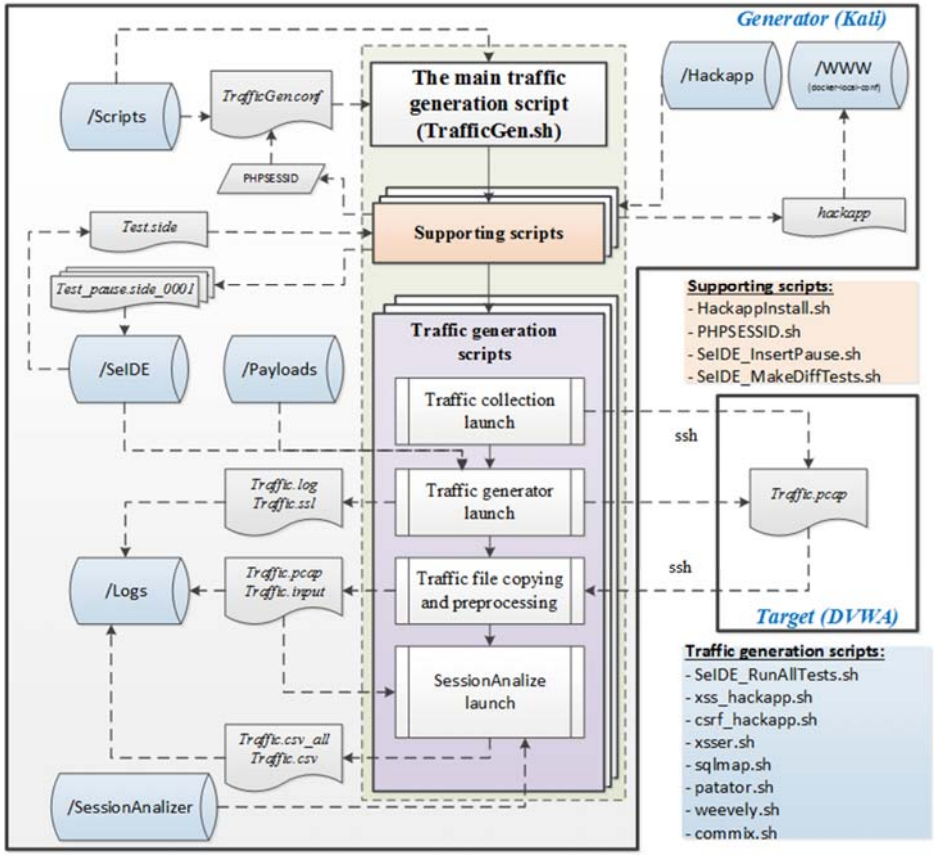

Рис. 2. Обобщенная схема работы комплекса скриптов генерации веб-трафика и формирования на его основе обучающего набора данных

Fig. 2. Web traffic generation scripts and training dataset generation

Общий алгоритм работы модулей генерации трафика состоит в следующем (рис. 2):

1) Осуществляется SSH-подключение к мишени, где запускается процедура сбора трафика с помощью программного средства Tshark. Для минимизации объема собираемых данных регистрация трафика может выполняться на интерфейсе docker0.
2) Осуществляется запуск генератора трафика (запуск соответствующего программного средства, которое моделирует трафик определенного вида). При этом ведется лог работы, в котором регистрируется выполняемое действие и время его начала и завершения. Кроме того, для HTTPS трафика в хранилище сохраняются SSL-ключи. По окончании работы генератора сбор трафика прекращается.
3) Выполняется копирование PCAP-файла трафика в центральное хранилище и его предобработка. В частности, для трафика, содержащего веб-атаки с элементами легитимных действий пользователей, выделяются номера пакетов, в которых содержится зловредная полезная нагрузка (для HTTPS трафика перед этим с помощью программного средства Tshark выполняется процедура расшифровки на основе сохраненных ранее SSL-ключей).

4) Осуществляется обработка PCAP -файла трафика на предмет выделения сессий и расчета их параметров с помощью разработанного в ходе исследования программного модуля SessionAnalizer. В случае трафика, содержащего веб-атаки с элементами легитимных действий пользователей, на вход данного модуля также подается файл с номерами пакетов, содержащих зловредную полезную нагрузку, что позволяет пометить сессии, содержащие данные пакеты. Результатом работы модуля SessionAnalizer является файл в формате CSV, где каждая строка соответствует выделенной сессии, а столбец соответствует параметру сессии (признаку). За основу взято признаковое пространство набора данных CICIDS 2017. При этом в обучающий набор данных включается только тот трафик, который генерируется с IP адреса хоста-генератора на заданный порт (80/443), по заданному протоколу (HTTP/HTTPS), а также содержащий интересующие пакеты. Последним столбцом результирующего файла включается метка, которая определяет принадлежность сессии к фоновому трафику или к одной из моделируемых веб-атак. Данная метка предопределена заранее и соответствует типу трафика, который моделируется скриптом. Результатом работы каждого модуля генерации трафика будет являться фрагмент обучающего набора данных, принадлежащий только одному классу.

5.6 Развертывание испытательного стенда и сбор данных

Как отмечалось выше, сбор данных при моделировании атак и фонового трафика для выбранного объекта защиты должен осуществляться в реальной сети. В этой связи было осуществлено развертывание элементов предложенного испытательного стенда в реальной инфраструктуре в виде соответствующих виртуальных машин:

- Web – виртуальная машина с операционной системой Ubuntu 20.04 с развернутыми в виде docker-контейнеров мишенями DVWA, использующими разные типы веб-серверов (DVWA_apache и DVWA_nginx);
- Attacker, Victim – виртуальная машина с операционной системой Kali Linux с установленными средствами генерации (моделирования) веб-трафика (Chrome, FireFox, Selenium IDE, Xsser, HackApp, Sqlmap, Weevely, Patator, Commix).

Результаты сбора обучающего набора данных для протокола HTTPS, полученные на развернутом испытательном стенде, представлены в табл. 6.

Табл. 6. Результаты сбора обучающего набора данных для протокола https
Table 6. The results of collecting the training dataset for the https protocol

Класс	Тип трафика	Обозначение	Инструмент	Кол-во	Всего по типу трафика	Всего по классам	Кол-во	Всего по типу трафика	Всего по классам
				Apache			Nginx		
Benign	Фоновый веб-трафик	benign	SeIDE	19901	29807	29807	4752	8498	8498
		benignsqli	SeIDE	5023			1861		
		benignxss	SeIDE	4883			1885		
Web Attack	CAPEC-63: Cross-Site Scripting (XSS)	xss	SeIDE	1286	13925	28223	944	11353	36027
			xsser	11637			9022		
			hackapp	1002			1387		
	CAPEC-62: Cross Site	csrf	hackapp	931	931		1422	1422	

	Request Forgery								
	CAPEC-66: SQL Injection	sqli	SeIDE	537	1455		521	1431	
			sqlmap	918			910		
	CAPEC-49: Password Brute Forcing	brute	patator	2693	2693		49	49	
	CAPEC-650: Upload a Web Shell to a Web Server	malware	weevely	6452	6452		6748	6748	
	CAPEC-88: OS Command Injection	commi	SeIDE	101	2767		100	15024	
			commix	2666			14924		
Итого:						58030			44525

6. Тестирование и апробация методики

На основе сформированного набора данных была синтезирована модель выявления веб-атак, в основе которой использовался хорошо зарекомендовавший себя классификатор RandomForestClassifier. При построении модели осуществлялся подбор гиперпараметров модели.

Модель строилась в предположении того, что веб-приложение функционирует по протоколу HTTPS на веб-сервере Apache, т.е. для ее обучения использовался соответствующий набор данных (табл. 6). Полученная модель (табл. 7, эксперимент № 4) по метрикам качества превосходит модели, представленные в [8], которые использовали при обучении набор данных CICIDS 2017 (табл. 7, эксперимент № 1-2), а также набор данных, собранный на реальной сети, но в ограниченных условиях (табл. 7, эксперимент № 3).

Табл. 7. Протокол эксперимента
Tabl. 7. Experiment protocol

Эксперимент / Характеристика	**Эксперимент №1**	**Эксперимент №2**	**Эксперимент №3**	**Эксперимент №4**
Этап обучения модели				
Используемый набор данных	Сбалансированная и предобработанная подвыборка веб-атак WebAttacks набора данных CICIDS 2017. 7267 записей, из них 5087 экземпляров класса «нет атаки» и 2180 экземпляров класса «есть атака».		Сформированный набор данных, соответствующих реальному сетевому трафику	Сформированный набор данных, соответствующих реальному сетевому трафику
Обучающая выборка	70% записей используемого набора данных		70% записей используемого набора данных	70% записей используемого набора данных

Модель машинного обучения	RandomForestClassifier (max_depth=17, max_features=10, min_samples_leaf=3, n_estimators=50)		RandomForestClassifier (max_depth=None, max_features='auto', min_samples_leaf=1, n_estimators=250)	RandomForestClassifier (max_depth=17, max_features=10, min_samples_leaf=3, n_estimators=50)
Наиболее значимые признаки	1. Average Packet Size 2. Flow Bytes/s 3. Max Packet Length 4. Fwd Packet Length Mean 5. Fwd IAT Min 6. Total Length of Fwd Packets 7. Fwd IAT Std 8. Flow IAT Mean 9. Fwd Packet Length Max 10. Fwd Header Length		Flow Packets/s Flow IAT Max Bwd Packet Length Min Flow Duration Flow IAT Mean Flow IAT Std Average Packet Size Fwd Packet Length Max Total Packets Fwd Header Length	Fwd IAT Total Fwd Packet Length Max Fwd Packet Length Std Bwd IAT Total Bwd Packet Length Std Total Length of Fwd Packets Flow IAT Max Bwd Packet Length Max Bwd Packet Length Mean Total Length of Bwd Packets
Этап тестирования модели				
Тестовая выборка	30% записей используемого набора данных. Тестовая и обучающая выборка не имеют пересечений	100% записей сформированного набора данных, соответствующих реальному сетевому трафику	30% записей используемого набора данных. Тестовая и обучающая выборка не имеют пересечений.	30% записей используемого набора данных. Тестовая и обучающая выборка не имеют пересечений
Значения метрик качества классификации				
Accuracy	0.983	0.456	0.858	0.983
Precision	0.982	0.812	0.812	0.998
Recall	0.961	0.033	0.966	0.961
F1	0.971	0.064	0.882	0.979

Полученные результаты свидетельствуют о необходимости обучения модели обнаружения атак на наборе данных, собранном на основе анализа сетевого трафика в отношении конкретного защищаемого объекта реальной сети. При этом перед процедурой формирования обучающего набора данных должна быть проведена всесторонняя оценка объекта защиты. По ее результатам должны быть определены:

- условия, в которых функционирует объект защиты;
- особенности конфигурации объекта защиты;
- перечень актуальных для объекта защиты атак;
- перечень инструментальных средств, используемых для моделирования актуальных атак в отношении объекта защиты;
- перечень инструментальных средств, используемых для моделирования фонового трафика объекта защиты;
- мишень, в качестве которой может выступать сам объект защиты или его аналог;
- генератор(ы), с помощью которых осуществляется моделирование фонового трафика и атак;
- типы генерируемого трафика и порядок его разметки;

- структура испытательного стенда, позволяющая моделировать фоновый трафик и все атаки из сформированного перечня (все типы трафика), а также обеспечивающая возможность развертывания стенда в реальной инфраструктуре;
- пути автоматизации процедуры формирования обучающего набора данных.

7. Заключение

В предшествующем исследовании [8] для оценки применимости методов машинного обучения в системах обнаружения компьютерных атак был проведен эксперимент с настройкой модели «случайный лес», обучением на публичном наборе данных CICIDS 2017 и тестированием в реальных условиях. Настройка параметров выбранного классификатора позволила на валидационной выборке получить оценку F1-меры 0.971 для набора данных CICIDS 2017. При этом была подчеркнута невозможность применения предобученной модели на тестовой выборке, полученной на основе анализа сетевого трафика в реальной компьютерной сети (F1-мера 0.064, неудовлетворительное качество). Для получения удовлетворительного качества обнаружения потребовалось выполнить переобучение модели на сформированном собственном наборе данных, что позволило получить значение F1-меры 0.882.

В настоящем исследовании предложена методика сбора такого обучающего набора данных, позволяющая синтезировать адекватную модель обнаружения компьютерных атак в отношении заранее известного объекта защиты. Основа методики заключается в том, что сбор обучающего набора данных ведется в реальной сети в отношении конкретного объекта защиты или его модели, максимально учитывающей все его характеристики. При этом методика предполагает использование при моделировании трафика широкого круга программных средств, запускаемых с применением временных задержек и других параметров (опций), позволяющих вариативно реализовывать негативные действия, а также требует осуществлять разметку трафика, явно содержащего как атаки, так и элементы легитимных действий пользователей, в том числе и при использовании протокола HTTPS. Это позволяет повысить точность выявления атак: результаты апробации предлагаемых решений демонстрируют возможности дальнейшего повышения качества обнаружения атак, на тестовой выборке получено значение F1-меры 0.979 (точность значения F1-меры ограничивается, в том числе, объемом выборки).

Для поддержания набора данных в актуальном состоянии необходимо включать в него данные по всем существующим на текущий момент актуальным генераторам атак и постоянно дополнять по мере появления новых средств реализации атак.

Кроме того, построенная модель обнаружения компьютерных атак должна дообучаться по мере расширения набора данных, а также апробироваться на атаках, реализуемых из разных точек расположения сетевой инфраструктуры.

Направлениями дальнейших исследований являются: возможная автоматизация процедуры обучения для новых объектов защиты; глубокий анализ признакового пространства с целью определения признаков, независящих от физической структуры сети, настроек используемого оборудования, используемых программных средств; всесторонняя оценка полученных результатов в сравнении с существующими средствами защиты информации.

Список литературы / References

[1]. Sarker I.H., Furhad M.H., Nowrozy R. AI-Driven Cybersecurity: An Overview, Security Intelligence Modeling and Research Directions. SN Computer Science, vol. 2, issue 3, 2021, article no: 173.
[2]. Sharafaldin I., Lashkari A.H., Ghorbani Ali A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proc. of the 4th International Conference on Information Systems Security and Privacy (ICISSP), 2018, pp. 108-116.
[3]. Ring M., Wunderlich S. et al. Computers & Security, vol. 86, 2019, pp. 147-167.

[4]. Гетьман А.И., Иконникова М.К. Обзор методов классификации сетевого трафика с использованием машинного обучения. Труды ИСП РАН, том 32, вып. 6, 2020 г., стр. 137-154 / Getman A.I., Ikonnikova M.K. A survey of network traffic classification methods using machine learning. Trudy ISP RAN/Proc. ISP RAS, vol. 32, issue 6, 2020, pp. 137-154 (in Russian). DOI: 10.15514/ISPRAS–2020–32(6)–11.
[5]. Khatouni A.S., Heywood N.Z. How much training data is enough to move a ML-based classifier to a different network? Procedia Computer Science, vol. 155, 2019, pp. 378-385.
[6]. Ghurab M., Gaphari G. et al. A Detailed Analysis of Benchmark Datasets for Network Intrusion Detection System. Asian Journal of Research in Computer Science, 2021, vol. 7, issue 4, pp. 14-33.
[7]. Magán-Carrión R., Urda D. et al. Towards a Reliable Comparison and Evaluation of Network Intrusion Detection Systems Based on Machine Learning Approaches. Applied Sciences, 2020, vol. 10, issue 5.
[8]. Горюнов М.Н., Мацкевич А.Г., Рыболовлев Д.А. Синтез модели машинного обучения для обнаружения компьютерных атак на основе набора данных CICIDS2017. Труды ИСП РАН, том 32, вып. 5, 2020 г., стр. 81-94 / Goryunov M.N., Matskevich A.G., Rybolovlev D.A. Synthesis of a machine learning model for detecting computer attacks based on the CICIDS2017 dataset. Trudy ISP RAN/Proc. ISP RAS, vol. 32, issue 5, 2020, pp. 81-94 (in Russian). DOI: 10.15514/ISPRAS–2020–32(5)–6.
[9]. 1998 DARPA Intrusion Detection Evaluation Dataset. URL: https://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset, accessed 24.10.2021.
[10]. KDD Cup 1999 Data. URL: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, accessed 24.10.2021.
[11]. Traffic Data from Kyoto University's Honeypots. URL: http://www.takakura.com/Kyoto_data/, accessed 24.10.2021.
[12]. NSL-KDD dataset. URL: https://www.unb.ca/cic/datasets/nsl.html, accessed 24.10.2021.
[13]. Intrusion detection evaluation dataset (ISCXIDS2012). URL: https://www.unb.ca/cic/datasets/ids.html, accessed 24.10.2021.
[14]. CTU-13 Dataset. URL: https://mcfp.felk.cvut.cz/publicDatasets/CTU-13-Dataset/, accessed 24.10.2021.
[15]. UNSW-NB15 Dataset. URL: https://ieee-dataport.org/documents/unswnb15-dataset#files, accessed 24.10.2021.
[16]. CIDDS-001 Coburg Intrusion Detection Data Set. URL: https://www.hs-coburg.de/fileadmin/hscoburg/WISENT-CIDDS-001.zip, accessed 24.10.2021.
[17]. UGR'16: A New Dataset for the Evaluation of Cyclostationarity-Based Network IDSs. URL: https://nesg.ugr.es/nesg-ugr16/index.php, accessed 24.10.2021.
[18]. Intrusion Detection Evaluation Dataset (CIC-IDS2017). URL: https://www.unb.ca/cic/datasets/ids-2017.html, accessed 24.10.2021.
[19]. Canadian Institute for Cybersecurity datasets. URL: https://www.unb.ca/cic/datasets/index.html, accessed 24.10.2021.
[20]. Argus. URL: https://openargus.org/, accessed 24.10.2021.
[21]. CICFlowMeter. URL: https://github.com/CanadianInstituteForCybersecurity/CICFlowMeter, accessed 24.10.2021.
[22]. NFStream: a Flexible Network Data Analysis Framework. URL: https://github.com/nfstream/nfstream, accessed 24.10.2021.
[23]. FCParser: Feature as a Counter Parser for Networkmetrics. URL: https://github.com/josecamachop/FCParser, accessed 24.10.2021.
[24]. Kostas K. Anomaly Detection in Networks Using Machine Learning. Master's Thesis. University of Essex, 2018, 70 p.
[25]. Wilkinson M., Dumontier M. et al. The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data, vol. 3, 2016, article number 160018.
[26]. Gharib A., Sharafaldin I. et al. An Evaluation Framework for Intrusion Detection Dataset. In Proc. of the International Conference on Information Science and Security (ICISS), 2016, pp. 1-6.
[27]. Sharafaldin I., Gharib A. et al. Towards a reliable intrusion detection benchmark dataset. Software Networking, issue 1, 2017, pp. 177–200.

Информация об авторах / Information about authors

Александр Игоревич ГЕТЬМАН – кандидат физико-математических наук, старший научный сотрудник ИСП РАН, доцент ВШЭ. Сфера научных интересов: анализ бинарного кода, восстановление форматов данных, анализ и классификация сетевого трафика.

Aleksandr Igorevich GETMAN – PhD in physical and mathematical sciences, senior researcher at ISP RAS, associate professor at HSE. Research interests: binary code analysis, data format recovery, network traffic analysis and classification.

Максим Николаевич ГОРЮНОВ – кандидат технических наук. Сфера научных интересов: информационная безопасность, системы обнаружения вторжений, системы анализа защищенности, машинное обучение, безопасная разработка программного обеспечения.

Maxim Nikolaevich GORYUNOV – Ph.D. Research interests: information security, intrusion detection systems, security analysis systems, machine learning.

Андрей Георгиевич МАЦКЕВИЧ – кандидат технических наук, доцент. Сфера научных интересов: информационная безопасность, системы обнаружения вторжений, системы антивирусной защиты, машинное обучение, криптографические методы защиты информации.

Andrey Georgievich MATSKEVICH – Ph.D., associate professor. Research interests: information security, intrusion detection systems, anti-virus protection systems, machine learning, cryptographic methods for protecting information.

Дмитрий Александрович РЫБОЛОВЛЕВ – кандидат технических наук. Сфера научных интересов: информационная безопасность, системы обнаружения вторжений, машинное обучение, криптографические методы защиты информации.

Dmitry Aleksandrovich RYBOLOVLEV – Ph.D. Research interests: information security, intrusion detection systems, machine learning, cryptographic methods for protecting information.