УДК 004.89

ИСПОЛЬЗОВАНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ГЕНЕРАЦИИ ПРАВИЛ ОБНАРУЖЕНИЯ ВРЕДОНОСНОГО ТРАФИКА В АВТОНОМНОМ ПРОГРАММНО-АППАРАТНОМ КОМПЛЕКСЕ

Шнайдер А.В., Казаков Ф.А.

ФГАОУ ВО «Сибирский федеральный университет», Красноярск, e-mail: office@sfu-kras.ru

В данной работе, в рамках создания автономного программно-аппаратного комплекса по выявлению вредоносных сетевых воздействий, рассматривается возможность применения методов обработки естественного языка (NLP) для генерации правил системы Snort. Оценивается возможность создания автоматического модуля генерации правил, что позволит говорить о возможности создания полностью автономного программно-аппаратного комплекса. Предлагается техника NLP для упрощения процесса составления правил и, таким образом, повышения скорости реакции на возникновение новых сетевых угроз. С помощью методов обработки естественного языка и методов машинного обучения новые правила генерируются на основе выделенного обучающего набора сетевого трафика (сетевых пакетов), который может формироваться как на основе уже известных сетевых угроз, так и на основе только что выделенных вредоносных пакетов. Для создания модуля генерации трафика выбран новый класс нейросетей – трансформеров, используемый для решения задач обработки естественного языка, учитывающий сильные зависимости. Выполнено сравнение стационарного типового набора правил выделения сетевых угроз с динамически сформированным набором с точки зрения обнаружения существующих типов вредоносного трафика и количества «ложноположительных» срабатываний, показано достаточно высокое качество полученного набора и минимальные показатели ошибок.

Ключевые слова: анализ сетевого трафика, машинное обучение, генерация правил, искусственная нейронная сеть, сигнатура

USING MACHINE LEARNING METHODS TO GENERATE RULES FOR DETECTING MALICIOUS TRAFFIC IN AN AUTONOMOUS HARDWARE AND SOFTWARE COMPLEX

Shnayder A.V., Kazakov F.A.

Siberian Federal University, Krasnoyarsk, e-mail: office@sfu-kras.ru

In this paper, within the framework of creating an autonomous hardware-software complex for detecting malicious network influences, the possibility of applying natural language processing (NLP) methods for generating Snort system rules is considered. The possibility of creating an automatic module for rule generation is assessed, which will allow talking about the possibility of creating a fully autonomous hardware and software complex. We propose an NLP technique to simplify the rule generation process and thus increase the speed of response to the emergence of new network threats. Using natural language processing and machine learning techniques, new rules are generated from a dedicated training set of network traffic (network packets), which can be generated from both already known network threats and newly identified malicious packets. To create a traffic generation module, a new class of neural networks – transformers – is selected, which is used to solve natural language processing problems, taking into account strong dependencies. The comparison of the stationary typical set of rules for detection of network threats and dynamically generated set-in terms of detection of existing types of malicious traffic and number of "false positives" is performed, it is shown that the obtained set is of high enough quality and minimal error rates.

Keywords: network traffic analysis, machine learning, rule generation, artificial neural network, signature

В рамках создания автономного программно-аппаратного комплекса для выявления вредоносного воздействия на сетевые ресурсы, возникли задачи автоматической генерации правил для проектируемой аппаратной системы контроля трафика. Задачи автоматической генерации вытекают на основе требований к автономному функционированию создаваемого комплекса.

Программная часть комплекса основана на системе Snort и использует аналогичный синтаксис набора правил для описания основных алгоритмов выделения вредоносного трафика. Для определения возможности автоматического расширения набора правил в данной работе обсуждаются проблемы использования алгоритмов машинного обучения для генерации синтаксически правильных правил. Предлагаемые решения проверялись на оригинальном ПО Snort, поэтому в тексте статьи упор делается именно на использование данного ПО.

Использование системы Snort с модулем искусственного интеллекта

Большинство работ по улучшению системы Snort с помощью использования методов искусственного интеллекта **(**ИИ), как правило, сосредоточены на образцах вредоносного трафика в их исполняемой форме или в форме набора данных (дампа трафика). Исследования показывают [1, 2], что технологии ИИ повышают скорость обнаружения вредоносного трафика. Для соз-

дания точных моделей классификации использовались различные характеристики образцов вредоносного трафика. Наконец, алгоритмы ИИ могут быть использованы для генерации правил на основе сигнатур вредоносного трафика, и именно этому посвящена работа.

Системы искусственного интеллекта для генерации сигнатур вредоносного трафика

Воздействие вредоносных программ проявляется на сетевом уровне, и для их идентификации и пометки создаются соответствующие правила, позволяющие классифицировать характер трафика на основе основных атрибутов, определенных строк и его типа [3]. Эта нетривиальная задача, может быть, и при создании правил в ручном режиме требует внимательного анализа большого объема данных для формирования совокупности признаков и выделения характерной строки (сигнатуры). Ручная обработка не исключает возникновения ошибок. Такими ошибками могут быть: необнаружение вредоносного трафика, «ложноотрицательные результаты» или чрезмерное количество обнаружений «ложноположительных результатов».

После обзора предметной области были сделаны выводы, что практически не проводилось исследований по генерации сигнатур для систем обнаружения вредоносного трафика с использованием выделенных классов вредоносного трафика в качестве входного набора данных. Цель автоматизации состоит в том, чтобы автоматически создаваемые правила точнее обнаруживали вредоносный трафик и оперативней обновлялись при выявлении новых классов атак. Для достижения этой цели была поставлена задача объединить методы обработки естественного языка с алгоритмами машинного обучения для генерации сигнатур и правил идентификации при использовании в автономном программно-аппаратном комплексе.

Система защиты от вторжений Snort

Система Snort разработана Мартином Роешем – это ведущий отраслевой инструмент, который используется в системах обнаружения и предотвращения вторжений для выявления определенных моделей сетевого трафика на основе правила [4]. Поскольку система Snort является инструментом с открытым исходным кодом и широко доступна, она имеет большое количество правил для обнаружения различных типов вредоносного трафика. Правила доступны в формате с открытым исходным кодом, это делает данную систему привлекательной для использования в программно-аппаратном комплексе. В большинстве отраслей система записи правил Snort считается стандартом де-факто и используется в других аналогичных решениях IDS и IPS. Язык генерации правил хорошо настраивается и способен работать в качестве сканера (т.е. позволяет в реальном времени обнаруживать определенные модели сетевого трафика), существует большое количество плагинов, способных оценить качество новых правил, что позволяет использовать систему Snort в автоматическом режиме.

Структура правил системы Snort и методы тестирования

Правило системы Snort (рис. 1) состоит из двух основных частей: заголовка и опций правила. Заголовок правила содержит «действия», протоколы, IP-адрес и маску источника/назначения, а также порты источника/назначения [5]. Опции правила содержат предупреждающие сообщения и спецификацию того, какая часть пакета должна быть проверена при принятии решения о действии правила.

Правила системы Snort создаются поэтапно и необходимы для обнаружения определенной вредоносной активности, происходящей в сети. Для тщательного определения и описания характерного поведения аномалий, которое должно быть обнаружено, обычно используют тестовые или неавторизованные локальные сети, также необходимо помнить, что правило для обнаружения отклоняющегося поведения не должно «обнаруживать» другие типы трафика, особенно доброкачественные или невредоносные типы трафика, поскольку это приведет к ложным срабатываниям.

После того как правило написано, его необходимо протестировать с помощью сетевых файлов, содержащих вредоносный трафик, на который направлено правило, а затем протестировать с другими типами трафика, включая невредоносный трафик. Как уже говорилось, помимо срабатывания на нужные типы вредоносного трафика целью является снижение количества ложных срабатываний, которые могут произойти при развертывании правила в реальной среде. Поэтому тестирование на срабатывание происходит следующим образом: проверяется, правильно ли система Snort загружает правило, проверяется, может ли система Snort вызвать предупреждение и, наконец, подтверждается, правильно ли регистрируются предупреждения [6]. Также правило должно быть протестировано на достаточно большой репрезентативной выборке доброкачественного трафика, чтобы убедиться, что оно не срабатывает ложно.

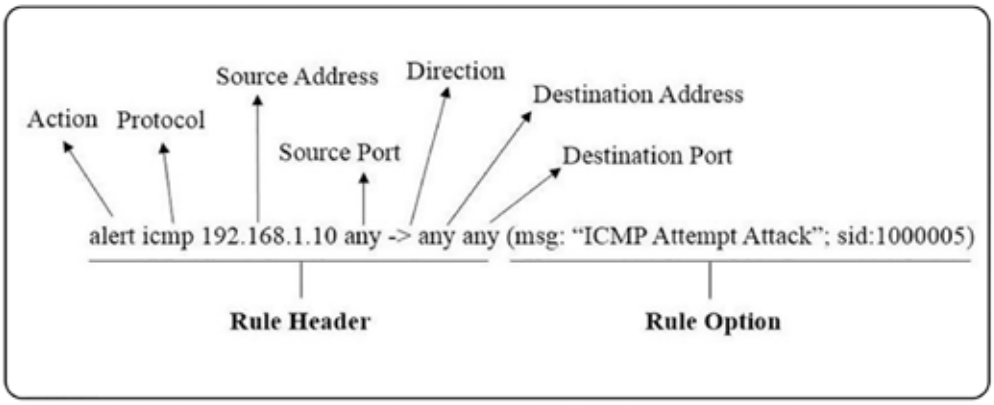

Рис. 1. Структура правила системы Snort

В реальных системах последствия ложных срабатываний могут варьироваться от переполнения журнала до ухудшения обнаружения или предотвращения.

Обработка естественного/искусственного языка и глубокое обучение

В лингвистике естественный язык – это любой язык, который естественным образом развился у людей в результате многократного использования и без сознательного планирования [7]. Это определение включает в себя все различные языки, на которых общаются люди. Искусственный язык – это любой язык, который был специально разработан для конкретных целей и обычно используется в вычислительных средах. Некоторые примеры искусственного языка: языки программирования и языки разметки. Поскольку языки этих классов имеют свои технические требования к формированию слов и предложений, они могут использоваться в различных лингвистических задачах, к таким задачам относятся: лексический анализ, статистический анализ, семантический анализ, вопросы и ответы, генерация текста и т.д.

Обработка естественного языка (Natural Language Processing, NLP) – это область лингвистики, информатики и искусственного интеллекта. NLP относится к взаимодействию между компьютерами и человеческим языком. Очень популярное в настоящее время использование NLP – это недавний набор интеллектуальных помощников, таких как Apple Siri, Google Assistant и Amazon Alexa [8]. NLP использует статистические методы и часто нейронные сети для выполнения различных задач: обработку текста в речь, генерацию текста, системы ответов на вопросы и т.д., а также растущее число других задач. Некоторые распространенные приложения NLP включают: информационный поиск, распознавание именованных сущностей и маркировку частей речи.

Использование глубокого обучения для NLP

Глубокое обучение (Deep Learning, DL) – это подобласть машинного обучения, которая связана с алгоритмами, основанными на структурах и функциях головного мозга, по этой причине их часто называют искусственными нейронными сетями. Искусственные нейронные сети – это машинное представление процессов и модели человеческого мозга (т.е. образование связей между нейронами).

Существует несколько типов искусственных нейронных сетей. К популярным относятся Конволюционные нейронные сети или Сверточные нейронные сети (CNNs), Рекуррентные нейронные сети (RNNs), Рекуррентные нейронные сети с управлением (GRUs), Сети с долговременной памятью (LSTM), Трансформеры и т.д. [9]. Эти типы нейронных сетей в настоящее время используются для выполнения различных задач, таких как распознавание изображений, финансовое прогнозирование, генерация текста и нарастающий список других задач.

Техники генерации текста NLP и Трансформеры

Генерация текста, также называемая генерацией естественного языка (Natural Language Generation NLG), является популярным подтипом NLP. NLG – это процесс обучения системы генерировать связный и логичный текст, который может быть использован для решения различных задач, таких как написание газетных статей, резюмирование сложных документов и т.д.

Трансформеры – это новая архитектура, разработанная для решения задач, свя-

занных с обработкой таких последовательностей, как текст на естественном языке, при одновременной обработке больших зависимостей. Под большими зависимостями понимается способность нейронной сети запоминать контекст в данной последовательности [10]. Трансформеры используют контекст (т.е. окружающие слова, которые связаны друг с другом) для удаления лишнего содержимого и для улучшения качества генерируемого текста, они также могут сделать полученный текст более кратким или осмысленным. Задача преобразования последовательностей – это задача, поставленная перед нейронной сетью по изменению последовательности в другую текстовую последовательность. Цель состоит в том, чтобы понять входной текст (входную последовательность) и создать менее размерное представление этой последовательности, далее это представление используется для генерации выходной последовательности. Контекст очень важен в NLG главным образом потому, что контекст является основным требованием при формировании новых предложений. Этот процесс называется внимание (в некоторых случаях самовнимание). Модуль внимание работает путем сравнения каждого слова входной последовательности со всеми другими словами, включая само слово, создается модель, которая учитывает контекстуальную релевантность и значение.

Трансформеры – это сети без повторений или петель, в которых используется механизм внимания, это техника, которая пытается имитировать когнитивные процессы, проходящие в человеческом мозгу. Цель механизма внимания – усилить или сфокусироваться на важной части данных и уменьшить важность остальных. Рекуррентные нейронные сети, в частности трансформеры, исторически было трудно реализовать в параллельных процессорных архитектурах, и они были менее эффективны при обучении текста с длительными зависимостями в последовательности входных и выходных данных. Реализация была осложнена требованием вычислений, требующих значительного объема памяти. Термин «длительные зависимости» относится к способности нейронной сети запоминать или обучаться на контексте слов. Длительные зависимости подразумевают, что система использует большую часть данных и пытается извлечь из них смысл и понимание, трансформеры могут улучшить общую производительность генерации текста при разумном использовании контекста.

Благодаря такому акценту на контексте, трансформер также может хорошо моделировать долгосрочные зависимости. Трансформеры используют несколько блоков для обработки данных и несколько входных последовательностей для создания общей выходной последовательности. При множественном распределении внимания несколько блоков работают одновременно, что позволяет им параллельно обслуживать различные части входной последовательности. Это значительно увеличивает скорость обучения нейронных сетей, делая их более практичными для многих новых моделей.

Архитектура трансформеров

Архитектура трансформеров была представлена Саи Сатьянараян [11]. Трансформер состоит из двух основных частей (рис. 2): кодера и декодера, обозначенных на схеме Nx, кодер на схеме изображен с левой стороны.

Кодер состоит из одного слоя многомодульного внимания, за которым сразу же следует еще один слой нейронной сети с прямой передачей. Декодер имеет ту же структуру, только с добавлением дополнительного скрытого слоя многомодульного внимания, это делается для того, чтобы сделать обучение более параллельным, что позволяет модели обучаться быстрее. В задачах NLG можно использовать различные конструкции нейронных сетей. В данной работе мы использовали модель Generative Pretrained Transformer 2 (GPT-2) от OpenAI. Эта модель является преемником модели GPT, выпущенной в открытый доступ в 2018 г. Архитектура GPT-2 не является абсолютно новой и похожа на трансформер, основное отличие заключается в том, что он обучается с использованием очень большого набора параметров.

Практическое использование SNORT и NLP с глубоким обучением

Основная задача использования NLP при построении модели противодействия вторжения в телекоммуникационную сеть – создание новых правил для системы Snort, которые будут обладать всеми требованиями к системам информационной безопасности, в частности увеличивать процент найденных аномалий и уменьшить количество ложноположительных срабатываний. В качестве исходных данных для проверки модуля NLP на разработанном программно-аппаратном комплексе использовался набор данных для тестирования IPS/IDS совместного проекта Службы безопасности связи (Communications Security Establishment) и Канадского института кибербезопасности (Canadian Institute for Cybersecurity) CSE-CIC-IDS2020, при этом были полученные следующие результаты (таблица).

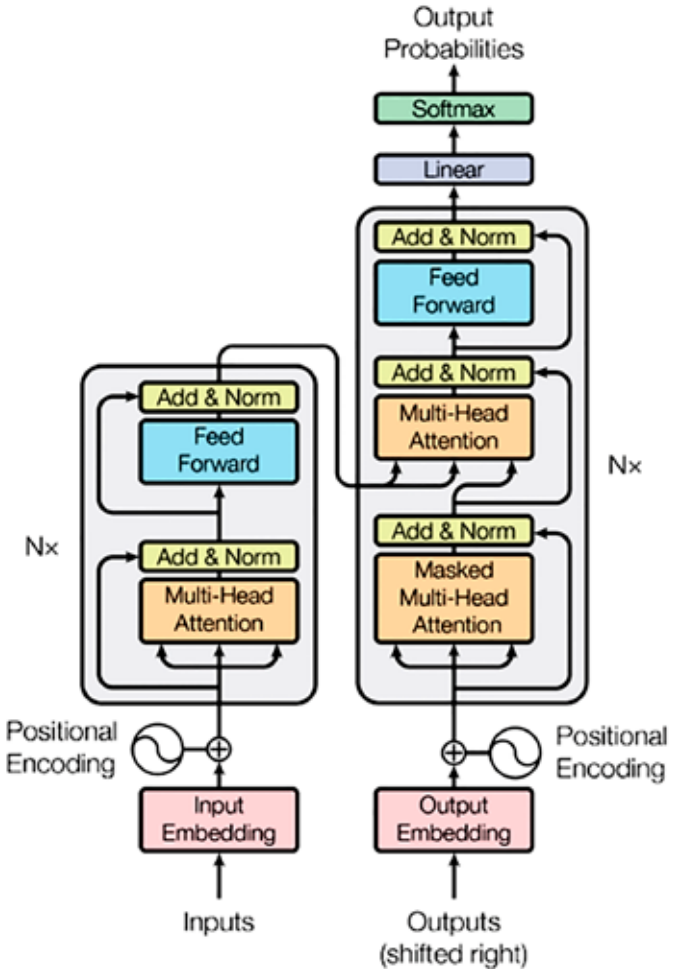

Рис. 2. Архитектура трансформера

Результаты работы систем Snort с модулем NLP

	Система Snort	Система Snort с модулем NLP
Количество правил, шт.	7 600	7 800
Найденные аномалии, %	100	95
Выделенные типы атак, шт.	120	120
Ложные срабатывания, %	0	0.132

Заключение

Набор данных CSE-CIC-IDS2020 представляет собой два дампа трафика, вредоносного и тестового (с примесями вредоносного и «обычного» трафика) с описанием каждого типа аномалий и набор правил для борьбы с ними. Для тестирования было подготовлено две системы Snort, на первую систему были загружены правила из набора данных CSE-CIC-IDS2020, предоставленные для их обнаружения, дамп вредоносного трафика был направлен во вторую систему с модулем NLP для автоматической генерации правил, которые в дальнейшем были загружены в «чистую» систему Snort. После на системы Snort был направлен дамп тестового трафика для оценки работы модуля NLP.

Как видно по результатам эксперимента (таблица), количество автоматически созданных правил сравнимо с набором созданных вручную группой экспертов. Качество созданного набора правил незначительно ниже эталонного, но все равно является достаточно высоким. Учитывая возможность дальнейшего оперативного обновления рабочего набора правил, использование автоматической генерации вполне применимо в автономных системах.

Список литературы

1. Гафаров Ф.М., Галимянов А.Ф. Искусственные нейронные сети и приложения: учебное пособие. Казань: Издательство Казанского университета, 2018. 121 с.

2. Kashyap N. Providing Cyber Security using Artificial Intelligence. A survey, in 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC). Mar. 2019. P. 717–720. DOI: 10.1109/ICCMC.2019.8819719.

3. Scarfone K.A., Mell P.M. Guide to Intrusion Detection and Prevention Systems (IDPS). National Institute of Standards and Technology, 2007. P. 28. P. 800-894. DOI: 10.6028/NIST.S.

4. Understanding and Configuring Snort Rules. Rapid7 Blog, Rapid7, Dec. 09. 2019. [Электронный ресурс]. URL: https://www.rapid7.com/blog/post/2016/12/09/understanding-and-configuring- snort-rules/ (дата обращения: 15.11.2022).

5. Peter J.B., Richard A.D. Introduction to Time Series and Forecasting – 2nd ed. Springer-Verlag New York, Inc. 2020. 449 с.

6. Abanda A., Mori U., Lozano J. A review on distance-based time series classification. Data Mining and Knowledge Discovery. 2019. Vol. 33. No. 2. P. 378–412.

7. Top NLP-Based Personal Assistant Apps Used In 2019. [Электронный ресурс]. URL: https://analyticsindiamag.com/top-nlp-based-personal-assistant-apps-used-in-2019/ (дата обращения: 19.11.2022).

8. Natural Language Processing (NLP): What Is It & How Does it Work? MonkeyLearn. [Электронный ресурс]. URL: https://monkeylearn.com/natural-language-processing/ (дата обращения: 16.11.2022).

9. Range Dependency – an overview // ScienceDirect Topics. [Электронный ресурс]. URL: https://www.sciencedirect.com/topics/computer-science/range-dependency (дата обращения: 16.11.2022).

10. Attention Is All You Need // Vaswani A. Dec. 2017. [Электронный ресурс]. URL: http://arxiv.org/abs/1706.03762 (дата обращения: 16.11.2022).

11. Sathyanarayan V.S., Kohli P., Bruhadeshwar B. Signature Generation and Detection of Malware Families. in Information Security and Privacy, Berlin, Heidelberg, 2008. P. 336–349. DOI: 10.1007/978-3-540-70500-0_25.