Инженерный вестник Дона, №8 (2024)
ivdon.ru/ru/magazine/archive/n8y2024/9457

Выявление ложноположительных инцидентов кибербезопасности на основе искусственных нейронных сетей

А.А. Исхаков[1], *А.З. Махмутова*[1,2], *И.В. Аникин*[1]

[1]*Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ, Казань*
[2]*Государственное бюджетное учреждение «Безопасность дорожного движения», Казань*

Аннотация: Исследована возможность обнаружения ложноположительных инцидентов кибербезопасности с применением моделей глубокого обучения – GRU, Bidirectional LSTM (Bi-LSTM), LSTM. Полученные результаты демонстрируют эффективность решения поставленной задачи для сценариев Powershell. Наилучшие результаты классификации показала модель Bi-LSTM, продемонстрировав точность 98,50 % на тестовой выборке.
Ключевые слова: машинное обучение, классификация, кибербезопасность, глубокое обучение, Powershell.

Введение

Согласно ГОСТ Р 59709-2022 «Защита информации. Управление компьютерными инцидентами. Термины и определения», под компьютерным инцидентом понимается факт нарушения и/или прекращения функционирования информационного ресурса, сети, электросвязи, используемой для организации взаимодействия информационных ресурсов, и/или нарушения безопасности, обрабатываемой в информационном ресурсе информации, в том числе произошедшей в результате компьютерной атаки.

Автоматизация обнаружения инцидентов кибербезопасности является неотъемлемой составляющей работы современных SIEM-систем. Однако, при этом возможны ложноположительные события, при котором средства обнаружения идентифицируют допустимое, законное или нормальное поведение как потенциально вредоносное, в отсутствии реальной угрозы [1]. Причинами появления ложноположительных инцидентов являются высокая настройка чувствительности детектора, устаревшие сигнатуры и правила, не совсем корректно настроенные исключения. В настоящей статье решается

задача выявления ложноположительных инцидентов кибербезопасности с помощью их классификации на основе искусственных нейронных сетей.

Характеристика рассматриваемых инцидентов кибербезопасности

Будут рассматриваться ложноположительные инциденты, связанные с реализацией сценариев Powershell [2]. Примером подобного инцидента является событие, детектируемое SIEM-системой на основе правила корреляции «Powershell_Obfusctation». Сценарий срабатывает при появлении событий, содержащих PowerShell - командлеты (или их фрагменты), отвечающие за обфускацию (запутывание кода), а именно:

– исполнение PowerShell-кода с Event ID 4103 или Event ID 4104, где поля "ScriptBlockText", "ContextInfo" или "Payload" содержат искомые командлеты;

– события создания процессов с Event ID 4688 или Event ID 1, где в качестве запускаемого процесса указан "powershell.exe" или "pwsh.exe", а в параметрах командной строки передаётся код, содержащий искомые командлеты.

Контролируемыми ключевыми словами будут являться: EncodedCommand, join, char, Sptth, Righttoleft. Данные события могут указывать на попытку обхода защитных средств устройства и выполнения полезной нагрузки вредоносного программного обеспечения.

Поскольку Powershell можно законно использовать для решения задач администрирования системы, возможно появление ложноположительных инцидентов. Пример одной из компаний показывает, что ее специалистами по информационной безопасности ежедневно расследуется более 300 инцидентов, связанных с использованием вредоносного программного обеспечения, в частности сценариев Powershell, 250 из которых (~83%) являются ложноположительными. При этом, ручной разбор каждого

инцидента занимает значительное время, что актуализирует решение задачи автоматического выявления ложноположительных инцидентов.

Методология исследовании

В связи с тем, что сценарии на языке PowerShell представляют собой текстовую последовательность, для выявления ложноположительных инцидентов будем использовать методы классификации текстов [3], в частности, методы, основанные на нейросетевых моделях [4,5,6] LSTM, Bi-LSTM [7], GRU [8]. В ходе исследования также была рассмотрена архитектура сверточных нейронных сетей [9]. По итогам анализа для обнаружения признаков PowerShell-сценариев была выбрана модель сверточной нейронной сети. Для учета контекста кодовой последовательности данных при классификации была выбрана рекуррентная нейронная сеть.

Подготовка данных для обучения

Для обучения нейросетевых моделей был использован датасет из 3102 вредоносных powershell сценариев, каждый из которых был представлен отдельным файлом с расширением PS1. Тестовая выборка состояла из 1,000 вредоносных и 1,000 легитимных файлов сценария PowerShell.

Вредоносные скрипты обладают функциями, которые можно использовать для классификации:

– Шелл-код. Установлено, что вредоносные сценарии PowerShell часто смешиваются с шелл-кодом. Следовательно, наличие шелл-кода в сценариях PowerShell является важным показателем.

– Информационная энтропия - это мера непредсказуемости информационного содержимого, которая используется для анализа распределения различных символов [10]. Энтропия доброкачественного скрипта выше, чем запутанного скрипта.

– Длина строки. Установлено, что доброкачественные сценарии, как правило, используют более длинные строки.

– URL или IP. Вредоносные сценарии PowerShell часто загружают вредоносное ПО или вредоносный код с внешних веб-сайтов для дальнейшей атаки.

– Специальные имена переменных. Переменная во вредоносных скриптах PowerShell всегда называется «cmd», «Shell», «c», в связи с тем, что злоумышленники часто используют сценарии PowerShell для вызова командной строки при атаках.

Архитектура системы выявления ложноположительных инцидентов кибербезопасности

Архитектура системы выявления ложноположительных инцидентов кибербезопасности представлена на рис. 1.

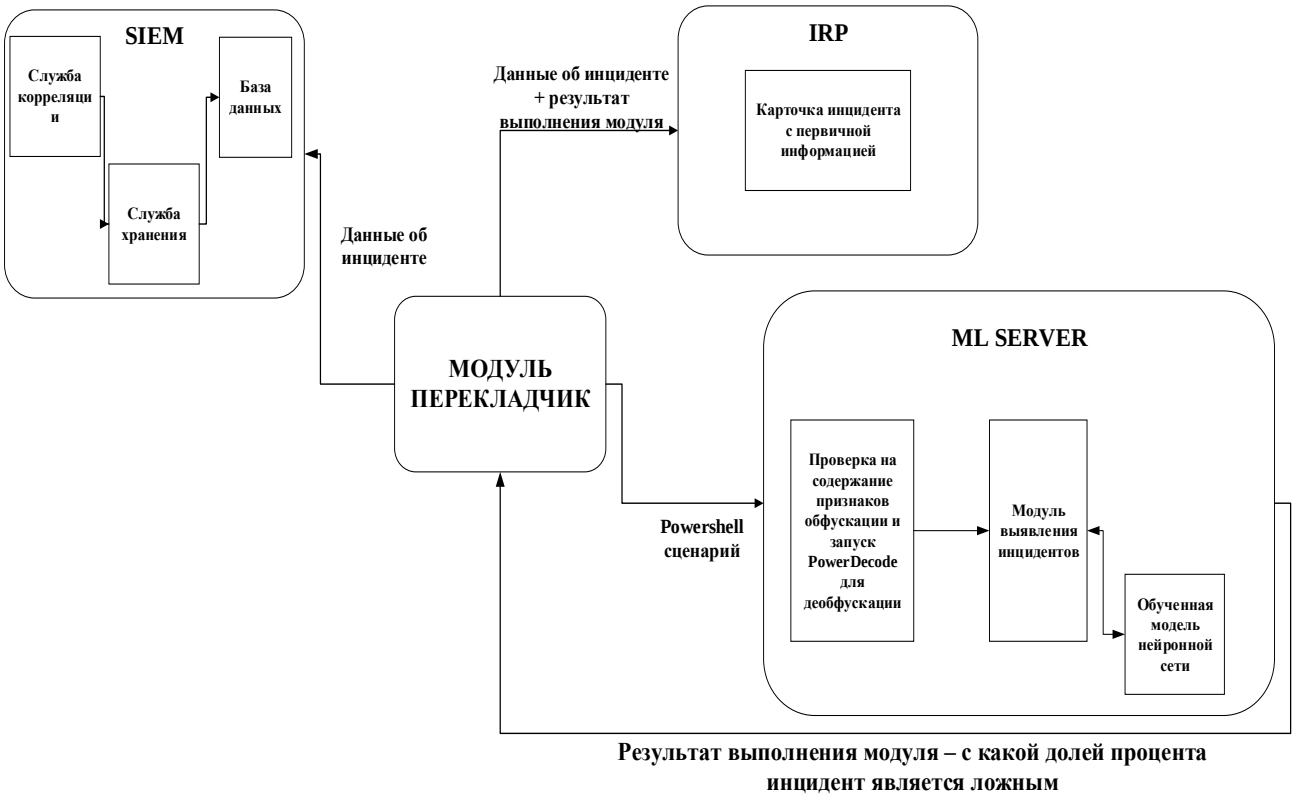

Рисунок 1 – Архитектура системы выявления ложноположительных инцидентов кибербезопасности

Первый блок представляет собой SIEM систему, детектирующую инциденты на основе правил корреляции. Модуль-перекладчик периодически обращается к базе данных для поиска информации о новых инцидентах. Если сработали правила корреляции, связанные с powershell сценариями, модуль перекладчик обращается к ML серверу для передачи информации о сценарии в утилиту PowerDecode для деобфускации. Далее, модуль выявления инцидентов подключается к обученной нейросетевой модели для классификации сценария. На заключительном этапе модуль перекладчик передает полученные данные в систему регистрации инцидентов IRP и заполняет карточку инцидента информацией из SIEM-системы и ML сервера.

Экспериментальная часть

На первом этапе PowerShell сценарий (текстовая последовательность) обрабатывается классической сверточной нейронной сетью. Результаты экспериментов представлены в Таблице 1.

Таблица № 1

Сравнительная таблица параметров для сверточных слоев НС

embedding-пространство	Количество фильтров	Размер ядра	Скорость обучения нейросетевой модели, мин		accuracy
			CPU	GPU	
16	256	32	160	18	84,15
	128	48	320	36	82,32
	64	64	480	54	81,18
	32	80	640	70	83,1
	16	96	800	88	83,26
32	512	128	1200	38	83,18
	256	96	1520	76	83,46
	128	64	1840	114	83,56
	64	160	2320	152	82,23
	32	192	2480	190	85,30
64	1024	128	1300	54	81,2
	512	192	1900	108	82,50
	256	256	2200	162	82,53
	128	320	2430	216	82,9

	64	384	2500	270	85,14

Архитектура наиболее эффективной сверточной нейронной сети включала в себя: embedding-пространство размерностью 32, 3 скрытых слоя (количество фильтров составило 512, 256, 128; размер ядра составил 128, 96, 64). Скорость обучения составила на CPU 1840 минут (31 час), на GPU 114 минут, нейронная сеть показала точность 83,56%.

Далее полученные последовательности обрабатывались различными моделями рекурентных нейронных сетей – GRU, Bi-LSTM, LSTM, в результате чего получены следующие результаты.

– Наилучшими параметрами при построении модели GRU являются: 2 скрытых слоя с количеством нейронов 32 и 64.

– Наилучшими параметрами при построении модели Bi-LSTM являются: 3 скрытых слоя с количеством нейронов 16, 16, 128.

– Наилучшими параметрами при построении модели LSTM являются: 2 скрытых слоя с количеством нейронов 16, 16.

Используя данные параметры, удалось достичь наивысшего показателя F-меры и скорости обучения моделей нейронной сети.

Таблица 2 содержит итоговые результаты экспериментов.

Таблица № 2

Точность распознавания инцидентов кибербезопасности на тестовой выборке и скорость обработки инцидентов

Модель нейронной сети	Тестовая выборка, %	Скорость обработки инцидента кибербезопасности, сек
GRU	96,79	4
Bidirectional LSTM	98,50	8
LSTM	91,42	10

Выводы

Результаты проведенных исследований демонстрируют эффективность применения глубоких нейронных сетей для выявления ложноположительных инцидентов кибербезопасности.

Модель Bi-LSTM показала наиболее высокую точность распознавания тестовой выборки. Точность распознавания составила 98,50 %. Наименьшую точность распознавания 91,42% показала модель LSTM. Наибольшую скорость обучения показала модель GRU, 24 минуты на конфигурации с GPU и 1440 минут на CPU.

В результате для распознавания ложноположительных инцидентов кибербезопасности рекомендуется использование модели Bi-LSTM.

Литература

1. Ho C.Y., Lai Y. C., Chen I-W., Wang F.Y., Tai W.H. Statistical Analysis of False Positives and False Negatives from Real Traffic with Intrusion Detection/Prevention Systems // IEEE Communications Magazine. 2012. 50. pp. 146-154.

2. Pleskonjic D., Krakovic D., Matkovic N., Milutinovic V., Omerovic S., Tomazic S. Reduction of False Positive Intrusions by using Neural Nets // Proceedings of Telecommunications in Modern Satellite, Cable and Broadcasting Services. 2007. pp. 7-10.

3. Люгер Д.Ф. Искусственный интеллект: стратегии и методы решения сложных проблем. М. Вильямс, 2002. 864 с.

4. Collobert R., Weston J. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning // Proceedings of International Conference on Machine Learning. 2008. pp. 160-167.

5. Gers F.A., Schmidhuber J., Cummins F. Learning to forget: Continual prediction with LSTM // Neural Computation. 2016. 28. pp. 2385-2399.

6. Дудкин Д.М., Кузнецов М.А., Авдосев Н.Г., Шабаловский В.А., Егунов В.А. Применение языковых нейросетевых моделей для обнаружения вредоносного программного обеспечения // Инженерный вестник Дона, 2024, №7 URL: ivdon.ru/ru/magazine/archive/n7y2024/9332.

7. Гудфеллоу Я. Глубокое обучение. М: ДМК Пресс, 2018. 652 с.

8. Николенко С, Кадурин А, Архангельская Е. Глубокое обучение. СПб.: Питер, 2018. 480 с.

9. Аникин И.В., Ягина А.В. Интеллектуальное обнаружение стеганографического преобразования изображений, основанное на классификации контейнеров // Инженерный вестник Дона, 2023, №8. URL: ivdon.ru/ru/magazine/archive/n8y2023/8618.

10. Jacovi A., Sar S., Goldberg Y. Understanding Convolutional Neural Networks for Text Classification // Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. 2018. pp. 56-65.

11. Морозов Д. И. Энтропийный метод анализа аномалий сетевого трафика в IP-сетях // Известия ЮФУ. Технические науки. 2006. №7. С. 120-124.

References

1. Ho C.Y., Lai Y. C., Chen I-W., Wang F.Y., Tai W.H. IEEE Communications Magazine. 2012. 50. pp. 146-154.

2. Pleskonjic D., Krakovic D., Matkovic N., Milutinovic V., Omerovic S., Tomazic S. Proceedings of Telecommunications in Modern Satellite, Cable and Broadcasting Services. 2007. pp. 7-10.

3. Lyuger D.F. Iskusstvennyy intellekt: strategii i metody resheniya slozhnykh problem [Artificial intelligence: strategy and methods for decision of compelx problems]. M.: Vil′yams, 2002. 864 p.

4. Collobert R., Weston J. Proceedings of International Conference on Machine Learning. 2008. pp. 160-167.

5. Gers F.A., Schmidhuber J., Cummins F. Neural Computation. 2016. 28. pp. 2385-2399.

6. Dudkin D.M., Kuznetsov M.A., Avdosev N.G., Shabalovskiy V.A., Yegunov V.A. Inzhenernyj vestnik Dona, 2024, №7. URL: ivdon.ru/ru/magazine/archive/n7y2024/9332.

7. Gudfelllou Y.A. Glubokoye obucheniye [Deep learning]. M: DMK Press, 2018. 652 p.

8. Nikolenko S., Kadurin A., Arkhangel′skaya E. Glubokoye obucheniye [Deep learning]. SPb.: Piter, 2018. 480 p.

9. Anikin I.V., Yagina A.V. Inzhenernyj vestnik Dona, 2023, №8. URL: ivdon.ru/ru/magazine/archive/n8y2023/8618.

10. Jacovi A., Sar S., Goldberg Y. Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. 2018. pp. 56-65.

11. Morozov D.I. Izvestiya YUFU. Tekhnicheskiye nauki. 2006. №7. pp. 120-124.

Дата поступления: 29.06.2024

Дата публикации: 8.08.2024