2018).
УДК 004.89 Дата подачи статьи: 14.12.18
DOI: 10.15827/0236-235X.126.268-272 2019. Т. 32. № 2. С. 268–272

Нейросетевой поведенческий анализ действий пользователя в целях обнаружения вторжений уровня узла

В.Н. Зуев [1], *зав. лабораторией, zvn_tver@mail.ru*
А.Ю. Ефимов [1], *зав. отделом, efimovay@cps.tver.ru*

[1] *НИИ «Центрпрограммсистем», г. Тверь, 170024, Россия*

В данной статье рассматривается применение машинного обучения для обнаружения аномалий в поведении пользователя.

С каждым годом количество известных атак стремительно растет. Для противостояния данной угрозе необходимо использовать эффективные средства защиты, такие как системы обнаружения вторжений. Этот вид средств защиты обычно использует сигнатурный анализ и требует регулярного обновления баз сигнатур вторжений, так как не способен обнаруживать атаки, сигнатуры которых отсутствуют в этих базах.

Более привлекательны методы, основанные на обнаружении аномалий, поскольку с их помощью можно выявлять неизвестные ранее атаки без необходимости предварительного создания сигнатур вторжений для каждой новой атаки. Одно из наиболее популярных направлений в обнаружении вторжений на уровне узла – анализ поведения пользователя.

В данной статье описывается метод обнаружения аномалий поведения пользователей, основанный на применении искусственных нейронных сетей. Метод использует информацию о командах пользователя, извлекаемую из системных log-файлов операционной системы и ПО. Данная информация о командах конвертируется во временной ряд, который потом используется для прогнозирования следующей команды пользователя. Количество ошибок прогнозирования команд пользователя определяет наличие аномалии в его поведении.

Экспериментальные результаты показали, что данный метод хорошо подходит для выявления аномалий в поведении пользователя и обладает низкой вероятностью ложных срабатываний.

Ключевые слова: *обнаружение вторжений, компьютерная атака, анализ событий, поведенческий анализ, обнаружение аномалий, нейронная сеть, машинное обучение, ПК «Ребус-СОВ».*

Компьютеры, которые функционируют в информационно-телекоммуникационных сетях, потенциально подвержены вторжениям (атакам) – несанкционированному доступу к информационным ресурсам со стороны нарушителей. Своевременное выявление вторжений достигается посредством *систем обнаружения вторжений* (СОВ).

По характеру работы эти системы делятся на два вида:

– СОВ уровня сети, осуществляющие перехват сетевого трафика, как правило, на границе сегмента сети (шлюз), и анализ перехваченного трафика с целью выявления атак;

– СОВ уровня узла, анализирующие события доступа к локальным файлам, системные вызовы, журналы *операционной системы* (ОС), а также осуществляющие перехват и анализ сетевого трафика узла.

В настоящей статье речь пойдет об обнаружении вторжений на уровне узла с использованием анализа событий ОС и ПО. Данное направление обнаружения атак незаслуженно обделено вниманием отечественных разработчиков СОВ. Российский рынок сертифицированных ФСТЭК-решений в основном представлен СОВ уровня сети. Сегодня полноценно обеспечивающим обнаружение вторжений на уровне узла можно считать лишь *программный комплекс* (ПК) обнаружения вторжений «Ребус-СОВ» [1], разработанный НИИ «Центрпрограммсистем» (г. Тверь). ПК «Ребус-СОВ» может использоваться на ЭВМ, объединенных в вычислительную сеть и функционирующих под управлением основных ОС, применяемых в Вооруженных Силах (Windows, МСВС и др.). Также ведутся доработки ПК «Ребус-СОВ» для обеспечения возможности функционирования на отечественных вычислительных средствах, созданных на основе микропроцессоров с архитектурой «Эльбрус», работающих под управлением ОС «Эльбрус-Д» и ОС СН «Astra Linux Special Edition» релиз «Ленинград».

Статистика угроз, ежегодно публикуемая

ведущими мировыми компаниями в области обнаружения вторжений, показывает, что защита уровня узла приобретает все большую актуальность. Согласно отчету компании Check-Point, до 2014 года основной целью атак были серверы [2], атаки на них составляли 68 % от общего количества. Ситуация изменилась в 2014 году, по итогам которого 40 % атак пришлось на серверы и 60 % – на рабочие станции пользователей, и такая тенденция продолжается [3]. Кроме того, в отчетах приводится статистика по росту количества APT-атак.

Термин APT (advanced persistent threat) появился в 2013 году после сообщения в «Нью-Йорк таймс» об атаке на серверы данного издания [4]. APT-атака всегда направлена на конкретную организацию и, как правило, на конкретный компьютер. В зоне риска оказываются объекты критически важной инфраструктуры. Как следствие, методы выполнения отдельных этапов таких атак зачастую адаптированы под конкретные цели.

В APT-атаках часто используются методы социальной инженерии, а также уязвимости нулевого дня, которые не могут быть выявлены сигнатурными методами. Согласно отчету компании Symantec, уязвимости нулевого дня появляются в среднем раз в неделю, а время на их устранение может составлять от одного дня до нескольких месяцев [5].

Если атака была осуществлена, ставится задача своевременного обнаружения ее следов и локализации. APT-атака, известная как Darkhotel, действовала через сети Wi-Fi в элитных отелях и похищала данные посетителей через их телефоны в течение 7 лет [6].

Эффективным способом противодействия описанным угрозам является анализ событий ОС и ПО. Традиционным для СОВ методом обнаружения вторжений является сигнатурный анализ, в котором события анализируются с помощью шаблонов атак из базы решающих правил [7]; подобный механизм присутствует и в ПК «Ребус-СОВ». Однако, учитывая высокую вариативность используемых злоумышленниками методов, можно сделать вывод о недостаточности использования только сигнатурных методов обнаружения вторжений.

Популярным направлением выявления вторжений является анализ поведения пользователей. При данном подходе осуществляется сравнение поведения пользователя в течение сеанса работы с некоторым эталонным поведением. Существует множество подходов к сравнению, в их основе лежит предположение, что каждый сотрудник ежедневно выполняет примерно одни и те же действия. Объем событий, генерируемых в течение одной пользовательской сессии, может достигать огромных значений, а сама задача выявления аномального поведения пользователя плохо формализуется, поэтому для решения подобных задач применяются методы машинного обучения, в частности, искусственные нейронные сети.

Нейронная сеть настраивается на обучающем множестве, характеризующем поведение пользователя. После обучения нейронная сеть анализирует поведение пользователя и принимает решение о наличии аномалий и возможном нарушении безопасности.

В работе [8] была предложена нейросетевая модель, обрабатывающая такие параметры работы пользователя, как время начала и окончания сессии, используемые сетевые ресурсы, тип операций и т.д. Данные параметры нормировались и подавались на вход нейронной сети. В качестве выхода сети выступал один нейрон, выдающий коэффициент нормальности: 0 – нормальное поведение, 1 – аномалия в поведении. Недостатками такого подхода являются использование схемы обучения «с учителем» (подразумевающей наличие при обучении меток как для нормальных, так и для аномальных данных) и, как следствие, необходимость искусственного создания условий аномального поведения, а учитывая многообразие возможных вариантов, нейронная сеть не может адекватно обучиться для функционирования в реальных условиях.

В работе [9] авторы предлагают нейросетевой детектор атак, идентифицирующий поведение пользователя на основе количества запусков различных команд в течение дня. В данной модели учитывается только количество команд и не учитывается их последовательность. Кроме того, количество подаваемых на вход нейронной сети команд ограничено (100 команд), хотя в реальных условиях оно может быть значительно выше.

Еще один подход был предложен в [10]. Аномальное поведение пользователя выявляется на основе последовательности выполняемых им команд. Суть метода заключается в том, что на основе выполняемых пользователем команд осуществляется прогноз следующей команды. По количеству отклонений за день делается вывод о наличии аномалии. Таким образом, задача идентификации в данном подходе сведена к прогнозированию времен-

ного ряда. Команды берутся из системных журналов ОС, кодируются и подаются на вход нейронной сети. На вход сети подаются m идущих подряд команд, в качестве ожидаемого выходного значения используется команда $m + 1$. Обучение выполняется по схеме «с частичным привлечением учителя» (то есть с наличием только нормальных помеченных данных) на всех командах за один день с использованием скользящего окна размером m. Нейронная сеть, обученная по данным аудита одного дня, используется для прогнозирования действий пользователя в последующие дни. Для каждого пользователя нейронная сеть обучается отдельно. Преимущества данного подхода:

– независимость от количества пользователей в системе;

– учет закономерностей поведения пользователя на основе информации не только о статистике команд, но и о последовательности их выполнения;

– отсутствие необходимости искусственно генерировать аномальные данные для обучения.

Однако в разные дни порядок действий пользователя и вид этих действий могут отличаться, например, для совещаний по понедельникам требуется распечатка документов. Количество этих действий также может варьироваться, например, при обработке электронной почты число входящих писем является случайной величиной. В результате операция, выполняемая после утренней проверки почты, почти всегда будет спрогнозирована с ошибкой. Также со временем задачи и обязанности пользователя могут постепенно меняться, что тоже должно учитываться. На основании данных фактов можно сформулировать два основных недостатка описанного подхода, которые приводят к большому количеству ложных срабатываний:

– обучение нейронной сети по результатам одной сессии не охватывает всего спектра возможных операций и их порядка;

– сравнение спрогнозированного значения со значением, непосредственно следующим за скользящим окном, приводит к большому количеству ошибок.

Учитывая названные недостатки, авторы данной статьи разработали новый метод. Его первое отличие от описанного выше метода заключается в используемых для обучения данных. Нейронная сеть обучается по данным аудита, сгенерированным по результатам работы нескольких сессий пользователя. Экспериментальным методом был выбран отрезок времени в один месяц. Нейронная сеть обучается по данным всех сессий за этот период и постоянно переобучается по мере поступления новых данных.

Вторым отличием является метод сравнения спрогнозированного и реального значений при сопоставлении поведения пользователя с подготовленной нейросетевой моделью. Очевидно, что жестко заданная позиция команды не подходит для реальных условий. Поэтому в разработанном подходе было принято решение сравнивать спрогнозированное значение с диапазоном фактических значений размером n, который в проводимом исследовании варьировался от 2 до 20. Если спрогнозированное значение соответствует одному из кодов действий в диапазоне фактических значений, считается, что прогноз соответствует фактическому значению.

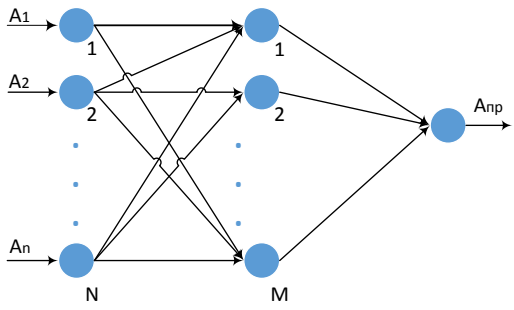
Согласно результатам проведенных исследований, структура сети для данной задачи представляет собой однонаправленную многослойную нейронную сеть, состоящую из нейронов сигмоидального типа (многослойный персептрон). Передача сигнала в этой сети осуществляется только в одном направлении – от входа к выходу.

Кодирование команд выполняется путем нумерации типа команды. Код каждой последующей команды инкрементируется. Нумерация действует в рамках всех сессий за рассматриваемый период. Число нейронов во входном слое равно размеру скользящего окна m, то есть на вход каждого нейрона поступает кодированное значение одной команды. Скрытый слой содержит число нейронов, равное $m \times 10$, а в выходном слое находится один нейрон, выдающий прогнозное значение следующей команды пользователя. Структура нейронной сети для прогнозирования команд пользователя приведена на рисунке.

Функция активации выходного слоя является линейной, а скрытых слоев – сигмоидальной: logsig: $\psi(x) = \frac{1}{1+e^{-x}}$.

Прогнозирование осуществляется на одну точку вперед. Качество прогнозирования оценивается с помощью показателя абсолютной процентной погрешности (MAPE):

$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|A_n - \hat{A}_{\text{пр}}|}{A_n}\times 100\,\%$, где $\hat{A}_{\text{пр}}$ – спрогнозированный код команды; A_n – фактическое значение команды; n – количество прогнозируемых значений [11].



Структура нейронной сети для прогнозирования команд пользователя

A neural network structure for predicting user commands

Данный подход полностью не исключает возможность ложных срабатываний, и каждое обнаруженное отклонение должно анализироваться администратором безопасности. Тем не менее, разработанная модель может оценивать тысячи событий в автоматическом режиме, что невозможно сделать в ручном режиме. Метод продолжает развиваться и проходит апробацию в ПК «Ребус-СОВ».

Таким образом, в данной работе предложен новый подход к построению модели поведения пользователя с целью выявления аномальной активности, основанный на применении нейронных сетей. Проведенные исследования свидетельствуют об эффективности разработанного метода. Его использование в СОВ совместно с сигнатурным анализом событий повысит эффективность обнаружения вторжений, в том числе принципиально новых и модифицированных существующих видов.

Для повышения степени эффективности обнаружения вторжений необходимо продолжить исследование, основной задачей которого является уменьшение количества ложных срабатываний, все еще происходящих при функционировании на реальных объектах.

Литература

1. Программный комплекс обнаружения вторжений «Ребус-СОВ». URL: https://rebus-sov.ru/ (дата обращения: 10.12.2018).

2. Check Point Security Report 2015. URL: https://blog.checkpoint.com/2015/06/16/check-point-2015-security-report-paints-a-picture-of-the-threat-landscape-and-its-not-pretty/ (дата обращения: 10.12.2018).

3. Check Point Security Report 2018. URL: https://blog.checkpoint.com/2018/04/16/2018-security-report-97-companies-unprepared-cyber-attacks/ (дата обращения: 10.12.2018).

4. Inside the Targeted Attack on The New York Times. URL: https://threatpost.com/inside-targeted-attack-new-york-times-013113/77477/ (дата обращения: 10.12.2018).

5. Symantec Internet Security Threat Report. 2016, vol. 21. URL: https://www.symantec.com/content/dam/symantec/docs/reports/istr-21-2016-en.pdf (дата обращения: 10.12.2018).

6. The Darkhotel APT. A Story of Unusual Hospitality. URL: https://media.kasperskycontenthub.com/wp-content/uploads/sites/58/2014/11/21181939/darkhotel_kl_07.11.pdf (дата обращения: 10.12.2018).

7. Лукацкий А. Обнаружение атак. СПб: БХВ-Петербург, 2001. 624 с.

8. Yuan F., Cao Y., Shang Y., Liu Y., Tan J., Fang B. Insider threat detection with deep neural network. Proc. ICCS 2018. LNCS, 2018, vol. 10860. DOI: 10.1007/978-3-319-93713-7.

9. Manoranjan P., Sateesh K., Sudhir K. Anomaly detection using artificial neural network. IJSET, 2012, vol. 2, iss. 1, pp. 29–36.

10. Zheng G., Srikumar V. DeepLog: anomaly detection and diagnosis from system logs through deep learning. Proc. 2017 ACM SIGSAC CCS'17, 2017, pp. 1285–1298. DOI: 10.1145/3133956.3134015.

11. Оссовский С. Нейронные сети для обработки информации; [пер. с польск. И.Д. Рудинского]. М.: Финансы и статистика, 2004. 344 с.

Software & Systems
DOI: 10.15827/0236-235X.126.268-272

Received 14.12.18
2019, vol. 32, no. 2, pp. 268–272

Neural network user behavior analysis for detecting host-level intrusion

V.N. Zuev [1], *Head of Laboratory, zvn_tver@mail.ru*
A.Yu. Efimov [1], *Head of Department, efimovay@cps.tver.ru*

[1] *R&D Institute Centerprogramsystem, Tver, 170024, Russian Federation*

Abstract. The paper focuses on applying machine learning for detecting anomalies in user behavior.

The number of known attacks is rapidly increasing every year. In order to resist that treatment, there is a need for effective security systems, such as Intrusion Detection Systems (IDS). This type of systems usually uses signature analysis and requires signature updates. Such systems are not capable of detecting unknown attacks.

The methods based on anomaly detection are more attractive as they can identify previously unknown attacks without preliminary creating of intrusion signatures for every possible attack. One of the most popular directions for host-based IDS in anomaly detection is user behavior analysis.

The paper describes a method of detecting user behavior analysis anomalies based on artificial neural networks. For detecting user behavior anomalies, the method uses the information on user's commands extracted from system log files and software. This information is converted into time series that is used to forecast next user's commands. The number of forecasting errors determine the presence of an anomaly in user behavior.

The experimental results demonstrate that the proposed method is good for detecting anomalies in user behavior, and has low probability of false positive.

Keywords: : intrusion detection, log file analysis, user behavior, anomaly detection, neural networks, machine learning.

References

1. *Intrusion Detection System "Rebus-SOV"*. Available at: https://rebus-sov.ru/ (accessed December 10, 2018).
2. *Check Point Security Report 2015*. Available at: https://blog.checkpoint.com/2015/06/16/check-point-2015-security-report-paints-a-picture-of-the-threat-landscape-and-its-not-pretty/ (accessed December 10, 2018).
3. *Check Point Security Report 2018*. Available at: https://blog.checkpoint.com/2018/04/16/2018-security-report-97-companies-unprepared-cyber-attacks/ (accessed December 10, 2018).
4. *Inside the Targeted Attack on The New York Times*. Available at: https://threatpost.com/inside-targeted-attack-new-york-times-013113/77477/ (accessed December 10, 2018).
5. *Symantec Internet Security Threat Report*. 2016, vol. 21. Available at: https://www.symantec.com/content/dam/symantec/docs/reports/istr-21-2016-en.pdf (accessed December 10, 2018).
6. *The Darkhotel APT. A Story of Unusual Hospitality*. Available at: https://media.kasperskycontenthub.com/wp-content/uploads/sites/58/2014/11/21181939/darkhotel_kl_07.11.pdf (accessed December 10, 2018).
7. Lukatsky A. *Intrusion Detection*. St. Petersburg, BHV-Peterburg Publ., 2001, 624 p.
8. Yuan F., Cao Y., Shang Y., Liu Y., Tan J., Fang B. Insider threat detection with deep neural network. *Proc. ICCS 2018. LNCS*. 2018, vol. 10860. DOI: 10.1007/978-3-319-93713-7.
9. Manoranjan P., Sateesh K., Sudhir K. Anomaly detection using artificial neural network. *IJSET*. 2012, vol. 2, iss. 1, pp. 29–36.
10. Zheng G., Srikumar V. DeepLog: anomaly detection and diagnosis from system logs through deep learning. *Proc. 2017 ACM SIGSAC CCS'17*. 2017, pp. 1285–1298. DOI: 10.1145/3133956.3134015.
11. Osowsky S. *Neural Networks for Information Processing*. Moscow, Finansy i statistika Publ., 2004, 344 p.

Для цитирования

Зуев В.Н., Ефимов А.Ю. Нейросетевой поведенческий анализ действий пользователя в целях обнаружения вторжений уровня узла // Программные

For citation

Zuev V.N., Efimov A.Yu. Neural network user behavior analysis for detecting host-level intrusion. *Software & Systems*. 2019,