9 МЕТОДЫ И СИСТЕМЫ ЗАЩИТЫ ИНФОРМАЦИИ

УДК 004.056.53

ПРИМЕНЕНИЕ ДИНАМИЧЕСКОЙ БАЙЕСОВСКОЙ СЕТИ В СИСТЕМАХ ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ

С.А. Арустамов, В.Ю. Дайнеко

Сформулированы проблемы, встречающиеся при функционировании известных систем обнаружения вторжений. Дано описание структуры разработанной модели системы обнаружения вторжений. Приведены описания используемой динамической байесовской сети, алгоритмов обучения и тестовой доменной сети. Определены ошибки первого и второго рода работы при эксплуатации системы обнаружения вторжений. Представлены сравнительные результаты тестирования при использовании разработанной системы и системы Snort.

Ключевые слова: система обнаружения вторжений, динамическая байесовская сеть, байесовский вывод.

Введение

Возросшая роль компьютерных сетей в жизни не только отдельно взятого человека, но и всего общества и даже государства заставляет серьезно относиться к проблеме защиты компьютерных сетей. Для построения комплексной системы защиты наравне с остальными традиционно используемыми инструментами защиты все чаще применяют системы обнаружения вторжения (СОВ).

Существует два распространенных метода обнаружения вторжений, используемые в СОВ: основанные на сигнатурах и аномалиях. Сигнатурный метод позволяет точно обнаруживать вторжения, но не способен выявлять новые типы вторжений. Метод, основанный на аномалиях, позволяет выявить новые вторжения, но имеет низкую надежность, выраженную в высокой частоте ложных обнаружений. Для решения проблемы ложных срабатываний в СОВ авторами предлагается использовать алгоритмы байесовского вывода (БВ) для принятия решения о вторжениях. БВ является статистически корректным инструментом для прогнозирования, фильтрации и оценивания текущей модели, которая описывает наблюдаемые данные. Оценка выбранной модели позволяет ответить на вопрос о корректности описания наблюдаемых событий, произошедших в прошлом или настоящем. При этом БВ применяется в виде модели динамической байесовской сети (ДБС). ДБС предназначена для моделирования последовательности характеристик сетевого трафика. Отдельно авторы рассматривают проблему увеличения временного отрезка автономной работы СОВ без участия человека для обслуживания. Система обнаружения вторжения после ввода в эксплуатацию способна в течение некоторого промежутка времени функционировать эффективно. Однако с течением времени система может начать деградировать, что приводит к росту частоты ложных срабатываний и пропуску вторжений. Чтобы этого не происходило, требуется участие человека в обслуживании и настройки системы с течением времени.

Ранее уже производились попытки решить проблему ложных срабатываний в СОВ. Так, в [1] предложено использовать шаги БВ в переходной модели между выводами, чтобы оценить, содержит ли конкретный всплеск сетевого трафика атаку. В работе [2] была предложена модель, которая имитирует интеллектуального злоумышленника, используя байесовские методы, чтобы создать план целенаправленного действия. Это исследование также предлагает схему классификации событий на основе байесовских сетей (БС). Отмечается преимущество использования БС, выраженное в простоте включения дополненной информации в уже существующую модель.

В [3] считают, что байесовская система обеспечивает прочную основу для упрощения математики и решения тех проблем, которые СОВ не в состоянии решить. В работе [4] предлагается адаптивная СОВ с использованием БС, обученных на смеси реального и синтетического трафика с помощью *K*2 алгоритма, и использующая алгоритм *junction tree* для БВ. В [5] предлагается использовать скрытые марковские модели для обнаружения аномалий в системных вызовах к ядру операционной системы, что позволяет локализовать аномалию с точностью до системного вызова.

Цель настоящей работы заключается в разработке структуры модели системы обнаружения вторжений для повышения надежности выявления вторжения с использованием модели ДБС и повышения времени автономной работы системы.

Структура модели разработанной системы обнаружения вторжений

Предлагаемая авторами структура модели системы обнаружения вторжений представлена на рис. 1 и состоит из пяти модулей.

1. Модуль байесовского вывода производит решение о наличии вторжения на основании собранной с сенсоров информации о свойствах сетевых сессий с защищаемой сети и выбранной обученной модели ДБС. Под сеансом понимается промежуток времени между запросом на соединение и запросом на

разрыв соединения, в течение которого между двумя IP-адресатами посылаются потоки данных по определенному протоколу.

2. Модуль обучения отвечает за процесс обучения ДБС. Выбор критерия обучения производится модулем конфигурации СОВ.
3. Модуль конфигурации СОВ управляет работой СОВ и отвечает за выбор алгоритма обучения и алгоритма вывода.
4. Модуль базы моделей ДБС содержит обученные модели ДБС, применяемые в модуле вывода;
5. Модуль сбора и обработки информации представляет собой группу сенсоров, расположенных в контролируемых узлах сети. Сенсоры захватывают сетевые пакеты на контролируемом узле и формируют последовательности свойства сессий для модуля байесовского вывода. Сетевым сенсором является программа, использующая сетевую библиотеку WinPcap [6], которая позволяет захватывать и собирать информацию о сетевом трафике. Собранная информация о сессии обрабатывается в виде свойств сетевых сессий и передается на вход модуля байесовского вывода.

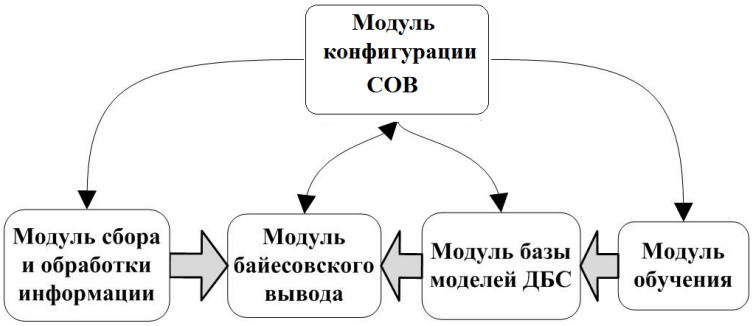

Рис. 1. Структура предлагаемой модели СОВ

С обучения ДБС начинается период инициализации работы СОВ. Используя один из алгоритмов обучения и обучающих данных, модуль обучения формирует обученную модель ДБС. Обучающие данные определяют ту априорную информацию, которая закладывается в модель ДБС. После обучения, обученная модель добавляется в базу моделей СОВ. В базе данных моделей каждая модель ранжируется по убыванию в зависимости от критерия оценки при обучении. Далее СОВ переходит в режим эксплуатации. Модуль байесовского вывода на основании лучшей модели ДБС, с точки зрения наблюдаемых свойств сетевых сеансов и выбранного алгоритма вывода, производит оценку БВ для определения аномалий в последовательностях. На этапе генерации модулем рабочей конфигурации СОВ может происходить оценка корректности описания наблюдаемых свойств сессий для текущей модели ДБС. При расхождении предсказания и наблюдения производится выбор следующей модели ДБС из базы моделей. Если перебор моделей ДБС не дал желаемых результатов, то модуль конфигурации СОВ производит обучение новой модели ДБС с помощью изменения алгоритма обучения. В случае обнаружения вторжений задача модуля конфигурации СОВ заключается в занесении обнаруженной последовательности в обучающие данные и обучении новой модели ДБС для получения лучшей модели описания вторжений. Кроме того, модуль конфигурации может взять на себя функцию по блокированию источников вторжений, например, путем добавления источника в черный список или выработки дополнительных правил для межсетевого экрана.

Применение динамических байесовских сетей

ДБС является обобщенной моделью в модели пространства состояний [7] для описания последовательностей, порожденных моделируемой динамической системой. ДБС состоит из двух байесовских сетей – исходной БС, определяющей структуру сети в каждом временном срезе, и транзитной БС, определяющей переходы между двумя ближайшими временными срезами. Структура БС остается неизменной во всех временных срезах. Срезом называют текущее состояние ДБС в дискретный момент времени без транзитивных связей.

ДБС может быть представлена в виде графической модели направленного ациклического графа, каждой из вершин которого соответствует случайная наблюдаемая или скрытая переменная, а дуги задают условную вероятность переходов между вершинами. Вершина в сети может иметь родителя только в своем временном срезе или в непосредственно предшествовавшем временном срезе, другими словами, ДБС определяется как марковский процесс первого порядка. В отличие от статических БС, ДБС имеют механизм для использования временного аспекта данных временных рядов и позволяют моделировать циклические явления, что дает возможность описывать сложные последовательности. Однако необходимость обучения исходной и транзитивной БС увеличивает вычислительную сложность. ДБС в виде гра-

фической модели приведена на рис. 2, где показана построенная авторами транзитная БС между двумя срезами. ДБС состоит из 9 вершин, которые, в свою очередь, представлены восьмью свойствами сетевых сеансов и одной переменной типа вторжения: protocol_type, service, src_bytes, count, srv_count, same_srv_rate, diff_srv_rate, dst_host_same_src_port_rate и type.

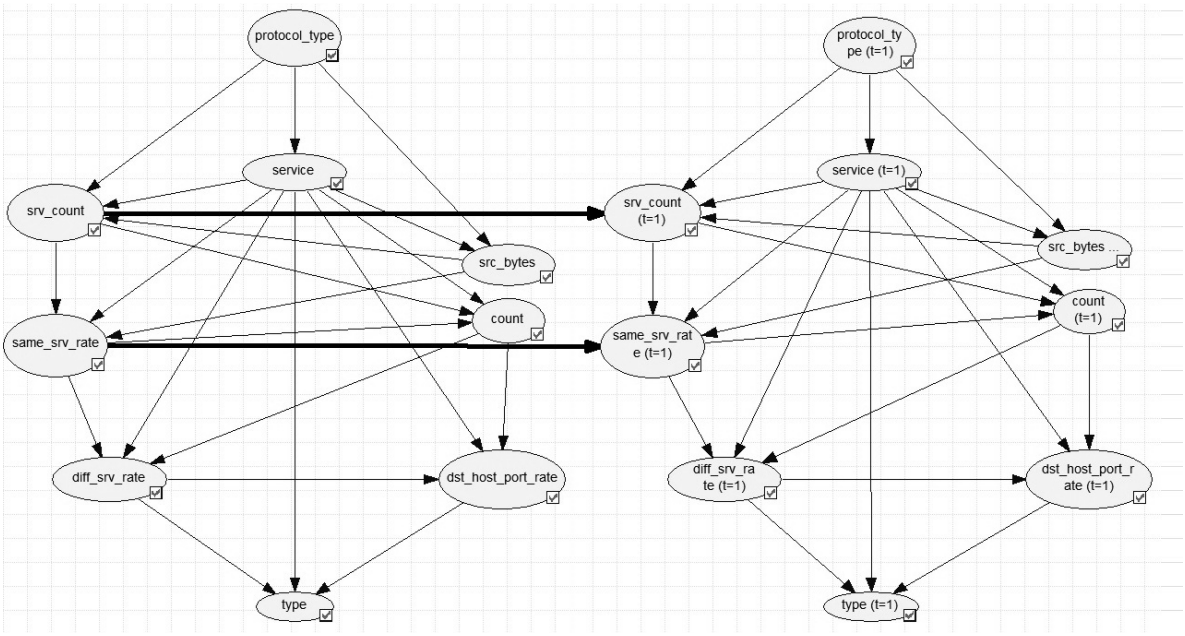

Рис. 2. Графическая модель транзитной ДБС

В модуле байесовского вывода СОВ используются алгоритмы БВ. Задачи, которые решаются с помощью БВ, представлены в табл. 1. Уравнение БВ для дискретного случая имеет вид

$$P(x_i \mid Y) = \frac{P(x_i) \cdot P(Y \mid x_i)}{P(Y)},\ x_i \in X,$$

где x_i – i-е скрытое состояние из всех возможных; X – пространство скрытых состояний; Y – данные наблюдений; $P(x_i \mid Y)$ – апостериорная вероятность скрытого состояния; $P(Y \mid x_i)$ – правдоподобие данных наблюдения; $P(x_i)$ – априорная вероятность скрытого состояния; $P(Y)$ – априорная вероятность наблюдаемых данных. В разработанной СОВ для решения задач предсказания, задачи Витерби, сглаживания на шаг применяются алгоритмы, сведенные в табл. 1.

Задача	Алгоритмы
Предсказание	$P(x(t+dt) \mid y(1:t))$ Алгоритм экстраполяции распределения вероятностей для будущих состояний ДБС
Фильтрация	$P(x(t) \mid y(1:t))$ Алгоритм оценки текущего состояния модели
Сглаживание	$P(x(1:t) \mid y(1:t))$ Алгоритм оценки всех наблюдаемых состояний в прошлом с учетом всех доказательств до текущего времени
Сглаживание на шаг	$P(x(t-dt) \mid y(1:t))$ Алгоритм оценки состояния для некоторого прошлого момента с учетом всех доказательств до текущего времени
Витерби	$\max_{x(1:t)} P(x(1:t) \mid y(1:t))$ Алгоритм для вычисления наиболее возможных последовательностей скрытых состояний с учетом полученных данных

Таблица 1. Алгоритмы вывода (t – время)

Алгоритм обучения

Для модуля обучения ДБС была выбрана комбинация применяемых алгоритмов обучения. Задача обучения ДБС заключается в поиске такой структуры из всего набора, которая обладала бы лучшей оценкой корректного описания данных обучения. В нашей работе используются:

- алгоритм поиска структуры сети *hill-clumbing*, реализованный в Probabilistic Network Library [8];
- критерий минимальной длины описания;

– критерий взаимной информации.

Эти критерии основаны на теории информации и реализованы в библиотеках GlobalMIT [9] и BnFinder [10] соответственно.

Апробация системы

Для тестирования работы СОВ была построена тестовая доменная сеть (рис. 3). Тестовая доменная сеть состоит из контроллера домена (Windows 2003 R2) и пяти клиентских машин (Windows XP SP3), межсетевого экрана и двух маршрутизаторов. Сетевые сенсоры СОВ установлены на каждую клиентскую машину тестовой сети и контроллер домена. На контроллере домена развернута служба Active Directory, хранящая критическую информацию, и запущены следующие серверы: DNS-сервер доменных имен, сервер службы принтеров, базы данных (Microsoft SQL).

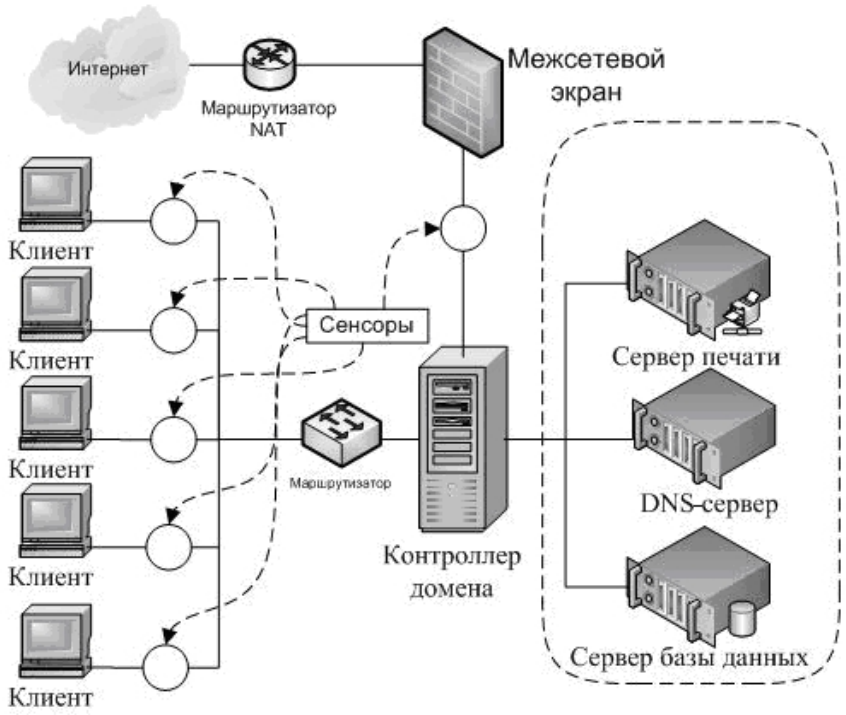

Рис. 3. Схема тестовой доменной сети

Атаки на доменную сеть, как правило, проводят в три этапа:

1. первоначальное сканирование сети;
2. проведение атак на найденные уязвимости для получения доступа к узлам сети;
3. поднятие своих прав до уровня администратора домена.

Авторами были реализованы два вида сканирований – сканирования сети для определения наличия сервера доменных имен (порт 53) и ARP-пинг. ARP-пинг – широковещательный запрос MAC-адреса компьютера с IP-адресом по протоколу ARP. Применялись следующие атаки на домен: использование уязвимости MS10-061 – повышение своих прав через службу печати; делегирование захваченных токенов, принадлежащих администратору; MS08-067 – выполнение произвольного кода из-за ошибки в проверке входных данных; ARP-spoofing – атака «человек посередине» с использованием протокола ARP. Атаки MS10-015, использующая доступ к 16-битным приложениям из 32-битной системы, и SMB relay, эксплуатирующая протокол NetBIOS, неизвестны для тестируемых СОВ.

Для создания обучающих данных авторами были проведены описанные выше сканирования сети и атаки на защищаемую доменную сеть. Вторжения выполнялись в ручном режиме. К собранным данным с сенсоров добавлялась информация о типе производимого вторжения.

После обучения и включения СОВ в режиме эксплуатации, с компрометирующего ресурса запускался диспетчер атак, в задачу которого входило проведение атак. Диспетчер атак запускал в случайной последовательности сканирования и атаки, направленные на тестовую доменную сеть. Атаки воспроизводились через каждые 5–10 минут в случайном порядке в течение 24 часов. Информация о времени запуска и о типе атаки сохранялось в файле истории.

Для сравнения эффективности работы разработанной СОВ в качестве аналога была выбрана система Snort [11]. Snort – свободная и открытая СОВ, производящая анализ трафика и использующая пра-

вила для обнаружений вторжений. Snort и предложенная СОВ в процессе своей работы сохраняли историю работы, выводя время обнаружения вторжения и его тип. По окончании тестирования файлы историй атак и историй обнаружения вторжений были проанализированы. Для определения эффективности работы СОВ использовались следующие оценки [12]:

- False Positive (FP) – вероятность обнаружения вторжения в случае, если его не было, определяемая как отношение количества ложных обнаруженных вторжений к общему числу сессий, содержащих вторжения. Такую ситуацию будем рассматривать как ошибку первого рода.
- False Negative (FN) – вероятность необнаружения вторжения в случае, если оно имело место, определяемая как отношение необнаруженных вторжений к общему числу сессий, содержащих вторжения. Такую ситуацию будем рассматривать как ошибку второго рода.

Всего было проведено 355 атак в течение 9846 нормальных сессий. В табл. 2 представлены результаты сравнения работы систем. Система Snort не смогла обнаружить новые типы вторжения, так как правила для нее задаются извне и не формируются в процессе работы. Разработанная СОВ показала высокую вероятность обнаружения вторжения и меньшую, чем Snort, вероятность ложных срабатываний.

Кроме того, обнаружение 20 новых вторжений во время эксперимента произошло без вмешательства специалиста, что подтверждает тезис об увеличении времени эффективной автономной работы.

	Обнаруженные вторжения	Пропущено вторжений (ошибка II рода)	Сессии, признанные нормальными	Ложные срабатывания (ошибка I рода)	Новые обнаруженные вторжения
Snort	281 (79,15%)	74 (20,85%)	9256 (94%)	590 (6%)	0
Разработанная СОВ	326 (91,84%)	29 (8,16%)	9561 (97,1%)	285 (2,9%)	20

Таблица 2. Сравнение результатов работы разработанной СОВ и Snort

Заключение

Авторами отмечены недостатки известных из литературы систем обнаружения вторжений – высокая вероятность ложных срабатываний и частое вмешательство человека в работу системы. Благодаря способности самонастройки алгоритмов своей работы разработанная система обнаружения вторжения способна увеличить время автономной работы без вмешательства специалиста. Это свойство сохраняется и для случаев, когда текущая модель динамической байесовской сети перестает корректно описывать наблюдаемые свойства сеансов. Проведенные эксперименты показали большую эффективность предложенной системы в сравнении с системой Snort для исследованных типов атак с точки зрения способности обнаруживать новые вторжения и уменьшения ошибок первого и второго рода.

Литература

1. Axelsson S. The base-rate fallacy and the difficulty of intrusion detection // ACM Transaction of Information System Security. – 2000. – V. 3. – P. 186–205.
2. Kruegel C., Mutz D., Robertson W., Valeur F. Bayesian event classification for intrusion detection // Proceedings of the 19th Annual Computer Security Applications Conference. – 2003. – P. 14–23.
3. Johansen K., Lee S. Network Security: Bayesian Network Intrusion Detection (BINDS) [Электронный ресурс]. – Режим доступа: http://www.cs.jhu.edu/~fabian/courses/CS600.424/course_papers/samples/Bayesian.pdf, свободный. Яз. англ. (дата обращения 02.10.2011).
4. Cemerlic A., Yang L., Kizza J.M. Network Intrusion Detection Based on Bayesian Networks // In Proceedings of SEKE. – 2008. – P. 791–794.
5. Аникеев М.В. Метод обнаружения аномалий на основе скрытых марковских моделей с поиском оптимального числа состояний // Материалы VII Международной научно-практической конференции «Информационная безопасность». – Таганрог: ТРТУ, 2005. – С. 58–60.
6. The WinPcap Team. WinPcap Documentation [Электронный ресурс]. – Режим доступа: http://www.winpcap.org/docs/docs_412/html/main.html, свободный. Яз. англ. (дата обращения 10.10.2011).
7. Murphy K.P. Dynamic bayesian networks: representation, inference and learning. The dissertation. – University of California Berkeley, 2002 [Электронный ресурс]. – Режим доступа: http://www.cs.ubc.ca/~murphyk/Thesis/thesis.pdf, свободный. Яз. англ. (дата обращения 02.11.2011).
8. Intel Corporation. Probabilistic Network Library – User guide and reference manual [Электронный ресурс]. – Режим доступа: http://www.sourceforge.net/projects/openpnl, свободный. Яз. англ. (дата обращения 15.12.2011).

9. Vinh N.X., Chetty M., Coppel R., Wangikar P.P. GlobalMIT: Learning Globally Optimal Dynamic Bayesian Network with the Mutual Information Test (MIT) Criterion // Oxford Journals: Bioinformatics. – 2011. – V. 27. – P. 2765–2766.
10. Wilczynski B., Dojer N. BNfinder: Exact and efficient method for learning Bayesian networks Supplementary Methods Documentation [Электронный ресурс]. – Режим доступа: http://bioputer.mimuw.edu.pl/software/bnf/supp.pdf, свободный. Яз. англ. (дата обращения 12.01.2012).
11. Sourcefire. Inc. Snort Users Manual 2.9.2 [Электронный ресурс]. – Режим доступа: http://www.snort.org/assets/166/snort_manual.pdf, свободный. Яз. англ. (дата обращения 12.01.2012).
12. Mattord H., Whitman M. Principles of Information Security // Course Technology. – 2008. – P. 290–301.

Арустамов Сергей Аркадьевич – Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, доктор технических наук, профессор, sergey.arustamov@gmail.com

Дайнеко Вячислав Юрьевич – Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, аспирант, daynekovy@yandex.ru