Методы обнаружения вторжений и аномалий функционирования киберсистем

С. А. Петренко

Термин «обнаружение вторжений» впервые появился в работах американских ученых Д. Андерсона и Д. Деннинг в 1980-е гг. Несмотря на достаточно большое количество имеющихся публикаций, говорить о создании теории обнаружения вторжений и аномалий киберсистем пока рано. Вопросы аксиоматики, терминологии, методологии и связи теории обнаружения вторжений и аномалий с другими научными дисциплинами находятся в стадии становления.

Сравнительный анализ известных моделей и методов обнаружения вторжений и аномалий показал следующее.

Наибольшее влияние на свойства групп методов обнаружения вторжений и аномалий представляется вносимым двумя системами классификаций: по уровню обрабатываемых данных и по схеме принятия решения о наличии факта нарушения (алгоритму решающей схемы).

Классификация по уровню обрабатываемых данных подразделяет на методы анализирующие:

У.1) двоичное представление данных или кодов команд;

У.2) команды, операции, события и/или их параметры (безотносительно их физического представления в средствах вычислительной техники);

У.3) характеристики системы, прямо или опосредованно отражающие ее целевое назначение, например, статистику задействованных ресурсов, количество обработанных за единицу времени запросов, скорость и другие характеристики сетевого обмена и т. п.

Алгоритмы низкоуровневого (машинно-зависимого) анализа, как правило, гораздо более просты в реализации, обладают высоким быстродействием и наименее требовательны к ресурсам. С другой стороны,

анализ двух более высоких уровней снабжает решающие алгоритмы более целенаправленным потоком информации, что потенциально повышает качество принимаемых решений при тех же затратах вычислительных мощностей, а кроме того обладает определенной степенью платформо-независимости. Наиболее высокий уровень анализа состояния АС (класс У.3) обычно информирует об имеющихся отклонениях опосредованно, что зачастую требует привлечения эксперта с целью обнаружения истинной причины нештатного функционирования АС. Однако в некоторых случаях он может быть единственным источником сведений о проводящемся деструктивном информационно-техническим воздействии (например, при распределенной атаке вида «отказ в обслуживании» путем формирования большого количества корректных, но ресурсоемких запросов и схожих случаях).

Классификация по схеме принятия решения о факте деструктивного информационно-технического воздействия противника представляется наиболее адекватной в разрезе подхода с позиций теории распознавания образов, к которой в общем случае относится данная задача.

Р.1. *Структурные методы* распознавания формируют строгую модель либо заведомо корректного состояния или воздействия, либо заведомо злоумышленного воздействия. Иные варианты воздействий, в т. ч. возможно корректные либо вредоносные (но неизвестные на момент создания модели), не анализируются и приводят либо к ошибкам I-го рода, либо к ошибкам II-го рода в зависимости от выбранного алгоритма анализа. К преимуществам методов данного класса относится полное отсутствие ложных срабатываний в области, описываемой моделью, к недостаткам — принципиальная невозможность описания новых, неизвестных ранее, либо не укладывающихся в разработанную модель методов злоумышленных воздействий.

Р.1.1. *Контроль корректности состояния*.

Р.1.1.1. *Алгоритмы инспектирования* выполняют наиболее жесткий контроль над системой: проверка целостности файлов (реализованы в системах Tripwire, AIDE и аналогичных), областей памяти или более сложных структур данных (например, баз префиксов сетевых маршрутов, на основе тех или иных записей о заведомо корректном их состоянии: размер файлов, контрольные суммы, криптостойких хеш-суммы и т. п.

Р.1.1.2. *Алгоритмы контроля графа состояний / графа переходов модели системы или протокола* представляют собой наиболее широко исследуемый подкласс структурных методов. Анализу подвергаются значимые события, происходящие в системе, о которой известно ее текущее состояние. Описание тем или иным способом разрешенных для каждого состояния переходов позволяет генерировать события при отклонении поведения системы от разрешенного. Одной из первых работ в данном направлении являлись исследования P. Porras и K. Ilgun, реализованные в системах STAT и USTAT соответственно. Впоследствии внутри выделился подкласс методов, использующий для контроля за последовательностью событий сети Петри. В настоящее время ведутся исследования, направленные на повышение гибкости описания допустимого поведения системы (например, в диссертации Д. Ю. Гамаюнова) и на автоматизацию процесса построения графа разрешенных переходов.

Р.1.1.3. *Алгоритмы контроля политики штатных воздействий* представляют собой полное или частичное описание разрешенных воздействий на систему, тем самым формируя политику «запрещено всё, что не разрешено» (англ. «*default deny*»), любая попытка нарушения которой формирует информирующее событие. Наряду с алгоритмами инспектирования представляет подкласс, имеющий наибольшую историю в области обнаружения компьютерных атак. Различные варианты реализации данного подхода были внедрены во множество систем контроля доступа (в т. ч. в одну из первых функционально завершенных IDS — проект Haystack в 1988 г.).

Р.1.2. *Контроль* (*поиск*) *нештатных воздействий*.

Р.1.2.1. *Алгоритмы контроля политики нештатных воздействий* представляют описание перечня заведомо запрещенных воздействий на систему, формируя политику «разрешено всё, что не запрещено» (англ. «*default allow*»). В отличие от контроля политики

штатных воздействий, которая может быть выведена из протокола или некоторого формального описания желаемого поведения системы, формирование полного перечня запрещенных воздействий затруднительно, а во многих случаях и невозможно в силу сложности и многоуровневости информационных систем. Решающие правила данного класса лишены ошибок «ложного срабатывания», что дает им значительно преимущество при внедрении в системах без участия человека-оператора. Однако они не в состоянии обнаруживать новые, не учтенные в их базе знаний типы злоумышленных воздействий на систему, а следовательно, качество их работы во многом зависит от скорости актуализации модели злоумышленника.

Р.1.2.2. *Сигнатурные алгоритмы* выполняют поиск заранее известных шаблонов компьютерных вторжений и отличаются уровнем анализа (согласно приведенной выше классификации), а также различной степенью детализации/обобщения шаблонов. Алгоритмы этого класса используют антивирусные программные продукты, а также системы фильтрации сетевого трафика (в т. ч. почтового и веб-контента). Современные исследования в этом классе на уровне анализа команд/событий посвящены в первую очередь универсализации баз знаний с целью унифицированной актуализации сведений об атаках различной этимологии, уровней и интенсивности, а также вопросам масштабируемости систем на их основе. В подклассе методов, выполняющих поиск на байт-ориентированном уровне, исследования ведутся в области автоматической генерации сигнатур вторжений, а также в области поиска эффективных методов противодействия мимикрии и полиморфизму в атаках (например, путем анализа графа переходов в двоичном коде червя.

Р.2. *Корреляционные методы* вводят метрики отличия наблюдаемого вектора признаков либо более сложной (например, поведенческой) характеристики от заведомо корректного либо заведомо злоумыш-

ленного состояния. Характеризуются тем, что формируют определенные (положительные или отрицательные) значения для всего множества воздействий — в том числе это касается и чрезвычайно маловероятных состояний (хотя степень достоверности при принятии решения в них невелика). Преимуществом корреляционных методов является покрытие всего множества допустимых воздействий, что гипотетически позволяет принимать корректные решения и в отношении неизвестных ранее атак. Задача совокупного снижения уровня ошибок как I-го так и II-го рода является основной для данного класса алгоритмов. Применительно ко всем корреляционным методам возможны как реализации в режиме «обучения с учителем» так и в режиме самообучения (адаптивный режим).

Р.2.1. *Алгоритмы «без памяти»* рассматривают каждое событие (воздействие, переход системы из одного состояния в другое, либо один отсчет измерения какой либо характеристики системы) как отдельный элемент множества, в отношении которого необходимо принять решение. К данному классу также применим термин *методы пространства признаков*.

Р.2.1.1. *С одномерным вектором признаков*.

Р.2.1.1.1. *Пороговые алгоритмы* генерируют информационное события о факте обнаружения аномалии по превышению наблюдаемого значения некоторой граничной величины. Пороговые алгоритмы были первыми представителями класса корреляционных методов обнаружения вторжений, в частности, они описаны и в основополагающей для всей области IDS работе D. Denning в 1987 г. и в системе Haystack (1988). Наибольшее применение на практике получил контроль за объемом запрашиваемых в системе ресурсов и за частотами тех или иных событий в системе (например, для конкретного вида событий — в работах, агрегированно для статистики рассеивания вектора частот — в исследовании П. А. Баранова).

Р.2.1.2. *С многомерным вектором признаков*.

Р.2.1.2.1. *Алгоритмы линейной классификации* в многомерном пространстве признаков в настоящее время уступили позиции алгоритмам кластерного и нейросетевого анализа, как более гибким.

Р.2.1.2.2. *Кластерный анализ* как зарекомендовавший себя метод классификации получил широкое применение и в области обнаружения компьютерных атак. В настоящее время исследования проводятся как в направлении обнаружения без учителя (поиск значительных отклонений), например, в работах L. Portnoy, так и кластеризации с предварительным обучением на размеченных входных данных.

Р.2.1.2.3. *Нейросетевые методы* используют для принятия решения о наличии либо отсутствии злоумышленного воздействия решающую схему на базе нейронной сети. Первые работы в этом классе относятся к концу 1990-х гг. В настоящее время количество различных методов в данном классе достаточно велико, в т. ч. существуют и независимые отечественные исследования. В частности в работе В. В. Райха, И. Н. Синицы и С. М. Шарашкина предлагается использовать нейронные сети адаптивного резонанса, а в работе С. В. Васютина и С. С. Завьялова решение принимается нейронной сетью на основании вектора, содержащего частоты системных запросов и идентификатор состояния контролируемого вычислительного процесса.

Р.2.1.2.4. *Иммунные методы* предпринимают попытку распространить принципы обнаружения и противодействия иммунной системы живых существ чужеродным вирусам. Система включает в себя централизованную «библиотеку генов» формирующую ограничен-

ный набор векторов, характеризующих потенциально чужеродные события, и распределенную систему датчиков, выполняющих собственно детектирование воздействий, и обладающих обратной связью с «библиотекой генов». Методы характеризуются нетребовательностью к ресурсам, однако, в некоторых условиях формируют высокий поток ложных событий.

Р.2.2. *Алгоритмы «с памятью»* анализируют события с учетом некоторой предыстории, а также, возможно, истинного или предполагаемого состояния системы.

Р.2.2.1. *Детерминированные алгоритмы контроля поведения* генерируют события по любому факту отклонения поведения системы от профиля, *созданного* на этапе обучения, и являются некоторым аналогом инспектирующих алгоритмов в классе структурных методов. В случае неудачного выбора объекта защиты или перечня контролируемых событий могут генерировать высокий поток ложных срабатываний. В основном вытеснены нечеткими алгоритмами как более гибкими.

Р.2.2.2. *Нечеткие алгоритмы контроля поведения* вычисляют в ходе анализа последовательности событий тем или иным образом вектор вероятностных характеристик и генерируют событие только по превышению ими некоторых *пороговых* значений. Анализ возможен как на уровне байт, например анализируются параметры системных запросов, так и на уровне команд/событий.

Заключение

К основным тенденциям развития современных методов обнаружения вторжений и аномалий киберсистем относятся:

- повышение достоверности и точности методов обнаружения вторжений и аномалий (снижение уровней ошибок I-го и II-го рода,

особенно в отношении ранее не наблюдавшихся информационно-технических воздействий);

- увеличение доли корректирующих процессов, не требующих участия человека эксперта, что снижает уровень эвристического принятия решения и позволяет перевести время реакции на злоумышленное воздействие на качественно новый уровень (например, при автоматической генерации сигнатур для нового вредоносного кода через несколько минут после подтверждения факта его аномально быстрого распространения по сети);
- противодействие новым технологиям, используемым злоумышленниками с целью: сокрытия факта вредоносного воздействия, например, с помощью полиморфных кодировщиков исполняемого кода и данных или техники мимикрии («растворения» либо маскировки в нормальном трафике) атак; активного воздействия на саму систему обнаружения атак путем формирования условий отказа в обслуживании либо генерации чрезмерного потока ложных срабатываний, что делает её применение невозможным.

Литература

1. *Anderson, James P.* Computer Security Threat Monitoring and Surveillance. Fort Washington, PA: James P. Anderson Co., 1980.
2. *Denning, Dorothy E.* (SRI International). «An Inrusion Detection Model.» IEEE Transactions on Software Engineering (SE-13), 2 (February 1987). P. 222–232.
3. *Debar, H., et al.* (IBM Zurich). Towards a Taxonomy of Intrusion-Detection Systems. Zurich, Switzerland: IBM Research Division, 1999.
4. *Axelsson S.* Intrusion detection systems: A taxonomy and survey. Technical Report 99–15, Dept of Computer Engineering, Chalmers University of Technology, Goteborg, 2000.
5. *Almgren M.* Consolidation and Evaluation of IDS Taxonomies. // Proceedings of the Eight Nordic Workshop on Secure IT Systems, NordSec 2003.
6. *Alessandri D., et al.* (IBM Zurich). Towards a Taxonomy of Intrusion-Detection Systems and Attacks. Zurich, Switzerland: IBM Research Division, 2001.
7. *RTO Techical Report* 49. Intrusion Detection: Generics and State-of-the-Art. RTO/NATO, Neuilly-sur-Seine CEDEX, France, 2002.
8. *Gene H. Kim, Eugene H. Spafford*. The Design and Implementation of Tripwire: A File System Integrity Checker. University of Purdue, 1993.

9. *Rami Lehti.* AIDE manual v0.13 http://www.cs.tut.fi/ ~rammer/aide/manual.html
10. *B. Gassend et al.* Caches and Hash Trees for Efficient Memory Integrity Verification // The 9th International Symposium on High Performance Computer Architecture (HPCA9), February 2003.
11. *M.Lad et al.* PHAS: A Prefix Hijack Alert System. // 15th USENIX Security Symposium/ 2006.
12. *Porras P. A., Kemmerer R. A.* Penetration State Transition Analysis — A Rule-Based Intrusion Detection Approach. // 8th Annual Computer Security Applications Conference, IEEE Computer Society Press, 1992. P. 220–229.
13. *Ilgun K.* USTAT: A real-Time Intrusion Detection System for UNIX. Computer Science Dept., Univ. of California, Santa Barbara, 1992.
14. *Kumar S., Spafford E. H.* An Application of Pattern Matching in Intrusion Detection. USA, Purdue University, 1994.