Куссуль Н.Н., Соколов А.М.

Адаптивное обнаружение аномалий в поведении пользователей компьютерных систем при помощи марковских цепей переменного порядка

Часть I. Адаптивная марковская модель переменного порядка

В настоящее время работа многих учреждений, компаний и физических лиц в значительной мере зависит от надежности функционирования и защищенности компьютерных систем. Уровень ценности обрабатываемых ими данных варьируется от частной и коммерческой информации до военной и государственной тайны. Получение несанкционированного доступа в компьютерную систему, уничтожение, изменение или разглашение информации, нелегальное использование ресурсов или приведения системы в недееспособное положение будет иметь нежелательные последствия для владельцев информации или системы. За последнее десятилетие мир убедился, что даже самые надежные системы защиты, на построение которых было потрачено много времени и денег, не всегда способны защитить компьютерные системы коммерческих, государственных и военных учреждений от хакерских атак.

Поэтому актуальной является разработка механизмов обнаружения попыток несанкционированного доступа. Для решения этой задачи предназначены *системы обнаружения вторжений* (*intrusion detection systems*, или *IDS*).

В данной работе предлагается метод обнаружения аномалий в компьютерных системах, основанный на адаптивной модификации модели марковских цепей переменного порядка. В первой части статьи обосновывается адаптивная марковская модель переменного порядка. Во второй части предложены несколько способов обнаружения аномалий при помощи новой модели и метод борьбы с replay-атаками.

1. Анализ IDS. Системы IDS можно разделить на 2 класса — системы обнаружения злоупотреблений (*misuse detection systems*), которые реагируют на уже известные атаки, и системы обнаружения аномалий (*anomaly detection systems*), выявляющие отклонения процесса эволюции системы от нормального.

1.1. Системы обнаружения злоупотреблений

Атака почти всегда является многошаговым процессом, и осуществление этих шагов требует высокой квалификации хакера. Потому простейший путь взлома состоит в применении стандартных средств (*exploits*), то есть уже написанных модулей, которые реализуют все необходимые этапы атаки на определенные уязвимые места системы.

Работа систем обнаружения злоупотреблений базируется на составлении шаблонов, или «подписей» атак. Защитные системы этого типа эффективны на известных схемах атак, однако, в случаях новой неизвестной атаки или отклонений хода атаки от шаблона возникают серьезные проблемы. Поэтому приходится поддерживать большую базу данных, включающую каждую атаку и ее вариации, и организовывать непрерывное пополнение баз шаблонов.

1.2. Системы обнаружения аномалий

Системы обнаружения аномалий, в отличие от экспертных систем, являются более гибкими и позволяют обнаруживать неизвестные атаки. Системы обнаружения аномалий основаны на предположении, что все действия злоумышленника обязательно чем-то отличаются от поведения обычного пользователя. То есть такие действия являются аномалиями. Работе таких систем предшествует период накопления информации, когда строится концепция нормальной активности системы, процесса или пользователя. Она становится эталоном, по которому оцениваются последующие данные.

Построение шаблона нормального поведения часто осложняется отсутствием данных, которые содержат следы вторжений. Поэтому обучение приходится проводить только на положительных примерах, что значительно усложняет задачу. Часто по той же причине тестирование систем обнаружения

аномалий тоже проводится при отсутствии реальных данных с «содержанием» атак. В этом случае используют перекрестные тесты (*cross tests*), когда данные работы одного субъекта проверяются по профилю другого.

После составления модели нормального поведения система обнаружения аномалий, фиксируя необходимую часть параметров системы, проверяет их на предмет соответствия модели. При этом возможны два вида ошибок:

1. Нормальное поведение системы/пользователя ошибочно принимается за злонамеренное (*false positives*).
2. Попытка нелегального проникновения в систему воспринимается как нормальная деятельность (*false negatives*).

Хотя ни одна из этих ситуаций нежелательна, вторая — более опасна. Поэтому одной из основных задач построения систем обнаружения аномалий является четкое определение условий, при которых ситуация воспринимается как аномальная, чтобы ни одна из перечисленных ситуаций не возникала слишком часто.

1.3. Подход к обнаружению аномалий, основанный на марковских цепях переменного порядка

Пионерской работой в отрасли обнаружения аномалий была работа [1], в которой определены основные понятия и подходы к этой проблеме. Позднее, в литературе появилось много примеров построения систем обнаружения аномалий. Большинство из них представляет собой концептуальные модели, цель которых — проверить возможность применения математической модели или подхода.

Практически все описанные в литературе модели обнаружения аномалий можно разделить на такие категории:

1. базирующиеся на перечислении и хранении примеров поведения [2];
2. частотные [1];
3. нейросетевые [3];
4. основанные на построении конечных автоматов [4–6].

Преимущественное большинство данных, доступное системам аудита в компьютерных системах, имеет последовательную и недетерминированную природу. Кроме того, для многих источников, которые генерируют эти последовательности, характерно то, что вероятность следующего символа или сигнала зависит от предыдущих. Часто они зависят только от небольшого количества предыдущих. О таких источниках говорят, что они имеют «короткую» память. Это наталкивает на мысль моделировать такие последовательные данные при помощи марковских цепей [1, 7–9] или скрытых марковских моделей [4]. Хотя эти статистические модели позволяют описывать широкий класс последовательностей, они имеют серьезные недостатки: размер марковских цепей экспоненциально увеличивается с ростом их порядка, то есть количество состояний соответствующего автомата ведет себя как $O(|\Sigma|^L)$, где $|\Sigma|$ — размер алфавита символов, L — порядок цепи. Следовательно, практическую ценность имеют только модели с очень небольшим порядком, а они могут плохо аппроксимировать последовательности, которые нас интересуют. Что касается скрытых марковских моделей, то помимо теоретически доказанной сложности их обучения, попытки их практической реализации показали необходимость весьма больших затрат на их настройку.

Как свидетельствует опыт, следующий символ или сигнал редко определяется предыдущим контекстом постоянной длины. Если рассматривать последовательности команд пользователей, то вероятность следующей команды определяется контекстом переменной длины. Значит нет потребности учитывать все возможные контексты — достаточно только тех, от которых зависит следующая команда. Совершенная система обнаружения аномалий должна принимать это во внимание. Поэтому была выбрана модель, которая позволяет получать марковские цепи переменного порядка по сериям примеров работы определенного класса автоматов [10].

Кроме того, большинство описанных в литературе подходов к моделированию поведения пользователей, программ или систем является

неадаптивными. В лучшем случае предлагается периодическая перенастройка модели с целью приспособления к изменчивому поведению. Это приводит к затратам времени и уменьшению надежности моделей со временем, которое проходит от очередного переобучения. Единственной известной нам работой, в которой применяется похожий на наш метод адаптивного изменения вероятностей, является [7]. Однако там за основу взята простейшая марковская модель первого порядка, а потому не учитывается контекст, длина которого превышает одну команду.

Для поддержки реальной адаптивности (когда для получения скорректированной модели текущая модель модифицируется с учетом поступившей информации) предлагается процедура изменения параметров модели таким образом, что наиболее весомой является более свежая информация. При этом сохраняются полезные аппроксимирующие свойства исходной модели.

2. Исходная модель

2.1. Базовые понятия

Рассматривается конечный алфавит символов (в нашем случае команд) Σ, конечное множество состояний автомата Q, где каждое состояние $q \in Q$ имеет строку-метку $s \in \Sigma^*$. Пустую строку пометим как $\boldsymbol{e}$.

Определение 1. Суффиксным вероятностным автоматом PSA (Probabilistic Suffix Automaton) *называется пятерка* $(Q, \Sigma, \tau, \gamma, \pi)$ *где* $\tau : Q \times \Sigma \to Q$ — *функция переходов между состояниями,* $\gamma : Q \times \Sigma \to [0,1]$ — *вероятность следующего символа,* $\pi : Q \to [0,1]$ — *начальное распределение состояний. Для пары состояний* $q_1, q_2 \in Q$, *и для каждого символа* $\sigma \in \Sigma$, *если* $\tau(q^1, \sigma) = q^2$ *и* q^1 *имеет метку* s^1, *то* q^2 *имеет метку* s^2, *которая является суффиксом* $s^1\sigma$.

Кроме того, для каждого $q \in Q$ *должно выполняться* $\sum_{\sigma \in \Sigma} \gamma(q,\sigma) = 1$ *и* $\sum_{q \in Q} \pi(q) = 1$.

Вероятность того, что PSA сгенерирует последовательность символов $r = r_1 r_2 \ldots r_N$, составляет

$$P_M^N = \sum_{q^0 \in Q} \pi(q^0) \prod_{i=1}^{N} \gamma(q^{i-1}, r_i),$$

где $q^{i+1} = \tau(q^i, r_i)$.

Автомат PSA, имеющий узлы с длиной меток не более *L*, обозначается *L-PSA*. В случае, когда множество меток состояний *Q* совпадает с Σ^L для некоторого *L* (то есть, присутствуют все возможные состояния с метками длины *L)*, то такой *L-PSA* может рассматриваться как марковская цепь порядка *L*. Поскольку в этом случае состояния легко идентифицируются по своим меткам, обучение автомата сводится к аппроксимации переходных вероятностей γ. Для общего случая, когда имеются только некоторые состояния, обучение значительно осложняется, поскольку мы должны правильно идентифицировать текущее состояние автомата.

Для решения задачи построения автомата по примерам работы некоторого PSA *M*, в качестве класса гипотез выбирается подкласс вероятностных машин — вероятностные суффиксные деревья PST (*Prediction Suffix Trees*).

Определение 2. Вероятностным суффиксным деревом *T, над алфавитом* Σ, *называется дерево степени* $|\Sigma|$, *где каждое ребро дерева помечено одним символом* $\sigma \in \Sigma$ *так, что из каждого внутреннего узла выходит одно ребро с таким символом. С каждым узлом дерева связана пара* (s, γ_s), *где s — строка, ассоциированная со «спуском», начиная из корня дерева в данный узел в обратном порядке символов в строке, которая помечает этот узел (рис. 1), а*

$\gamma_s : \Sigma \to [0,1]$ — *связанная с s функция вероятностей следующего символа. Требуется, чтобы* $\sum_{\sigma \in \Sigma} \gamma_s(\sigma) = 1$ *для каждой s, помечающей узел в дереве.*

Вероятность того, что PST сгенерирует последовательность символов $r = r_1 r_2 \ldots r_N$, составляет

$$P_T^N(r) = \prod_{i=1}^{N} \gamma_{s^{i-1}}(r_i), \qquad (1)$$

где $s^0 = e$ и для каждого 1<j<(N-1), s^j — строка, помечающая самый глубокий узел, в который мы попадаем, сделав проход по дереву в соответствии с $r_j r_{j-1} \ldots r_1$, начиная от корня дерева *T*. Например, для дерева на рис. 1 вероятность строки 00101 составляет 0.5*0.5*0.25*0.5*0.6, а метки узлов, в которых были последовательно сгенерированы символы, имеют вид $s^0 = e, s^1 = 0, s^2 = 00, s^3 = 1, s^4 = 010$.

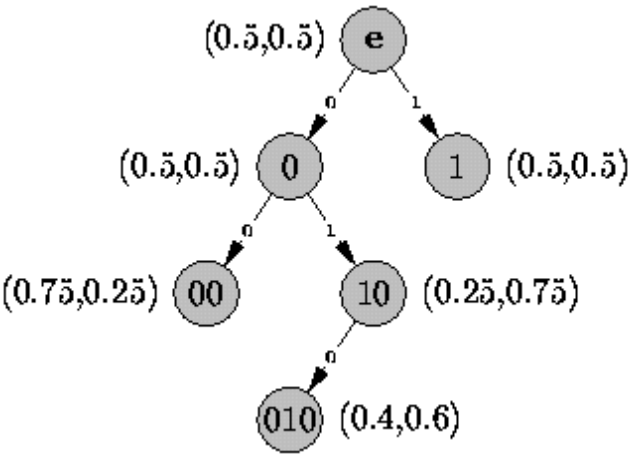

Рис. 1. Пример PST над алфавитом $\Sigma = \{0,1\}$.

Справедлива следующая теорема [10].

Теорема 1. *Для каждого L-PSA M можно построить эквивалентное ему по статистическим характеристикам PST T_M с максимальной глубиной L и не более чем $L \times |Q|$ узлами.*

Для оценки построенной гипотезы-дерева $\hat{T}$ нужно определить понятие качества гипотезы. Требуется, чтобы расстояние между вероятностными распределениями P_M и P_T строк, которые ими генерируются, не превышало заданного числа $\varepsilon > 0$ с большой вероятностью. Более формально

Определение 3. *PST $\hat{T}$ называется* ε -хорошей гипотезой *относительно PSA M, если для любого натурального N,*

$$\frac{1}{N} D_{KL}[P_M^N \| P_T^N] \leq \varepsilon,$$

где

$$D_{KL}[P_M^N \| P_T^N] = \sum_{r \in \Sigma^N} P_M^N(r) \log \frac{P_M^N(r)}{P_T^N(r)}$$

— это относительная энтропия.

По теореме 1 для любого PSA существует эквивалентное ему PST. Наоборот, можно доказать, что для каждого PST можно построить «почти» эквивалентный PSA (на первых L символах вероятности следующих символов могут отличаться). Следовательно, построенная в процессе работы алгоритма PST-гипотеза $\hat{T}$ может быть трансформирована в соответствующую гипотезу PSA $\hat{M}$ [10].

2.2. Алгоритм построения PST

Используя примеры работы целевого автомата *M*, определим *эмпирическую* вероятность $\widetilde{P}(.)$. Для данной подпоследовательности символов s $\widetilde{P}(s)$ — это относительное количество вхождений s в эту последовательность, $\widetilde{P}(\sigma|s)$ — относительное количество вхождений σ в строку после s. То есть, если m — это длина строки r, а L — максимальная длина s, то, определяя $\chi_j(s)$ как 1, когда $r_{j-|s|+1}...r_j = s$ и 0 в противном случае, имеем:

$$\widetilde{P}(s) = \frac{1}{m - L + 1}\sum_{j=L}^{m-1}\chi_j(s), \qquad \widetilde{P}(\sigma \mid s) = \frac{\sum_{j=L}^{m-1}\chi_{j+1}(s\sigma)}{\sum_{j=L}^{m-1}\chi_j(s)} .$$

Если всего имеется m' строк длиной $l \geq L+1$, то

$$\widetilde{P}(s) = \frac{1}{m'(l - L + 1)}\sum_{i=1}^{m'}\sum_{j=L}^{m-1}\chi_j(s), \qquad \widetilde{P}(\sigma \mid s) = \frac{\sum_{i=1}^{m'}\sum_{j=L}^{m-1}\chi_{j+1}(s\sigma)}{\sum_{i=1}^{m'}\sum_{j=L}^{m-1}\chi_j(s)} . \qquad (2)$$

Работа алгоритма построения PST зависит от таких параметров: L — максимальная длина меток состояний, n — верхний предел количества состояний в целевом PSA *M*, а также параметров $\varepsilon > 0$ и $\widetilde{P}(.)$. Требуется, чтобы алгоритм выдавал PST-гипотезу $\hat{T}$, которая бы с вероятностью как минимум $1-\delta$ была ε -хорошей гипотезой относительно PSA *M*.

Алгоритм начинает работу с дерева с одним корневым узлом ***e***. Потом в дерево добавляются следующие узлы, которые, по нашему мнению, должны ему принадлежать. Так, узел с меткой s становится листом дерева, если

эмпирическая вероятность $\widetilde{P}(s)$ не является пренебрежимой, и для некоторого символа σ эмпирическая вероятность $\widetilde{P}(\sigma \mid s)$ существенно отличается от эмпирической вероятности получить его после суффикса s $\widetilde{P}(\sigma \mid suffix(s))$, то есть s является определяющим контекстом для σ. Алгоритм завершает свою работу, если больше не существует листа, для которого справедливы приведенные условия или достигнуто ограничение на максимальную глубину дерева L.

В работе алгоритма также используется вспомогательный набор величин: $\varepsilon_0, \varepsilon_1, \varepsilon_2, \varepsilon_3$ и $\gamma_{\min}$. Они являются простыми функциями от $\varepsilon, \delta, n, L$ и $|\Sigma|$, конкретный вид которых устанавливается в ходе анализа алгоритма (подробнее см. [10]).

Алгоритм

1. Инициализировать $\hat{T}$ одним узлом ***e*** и множество $\overline{S}$

$$\overline{S} = \{\sigma \mid \sigma \in \Sigma, \widetilde{P}(\sigma) \geq (1 - \varepsilon_1)\varepsilon_0\}$$

2. Пока $\overline{S}$ не пусто, выполнять: выбрать какую-либо $s \in \overline{S}$ и

(a) удалить s из $\overline{S}$.

(b) если существует $\sigma \in \Sigma$ такого, что

$$\widetilde{P}(\sigma \mid s) \geq (1 + \varepsilon_2)\gamma_{\min} \tag{3}$$

и одновременно

$$\frac{\widetilde{P}(\sigma \mid s)}{\widetilde{P}(\sigma \mid suffix(s))} \geq 1 + 3\varepsilon_2 \tag{4}$$

то добавить узел с меткой s в дерево и (возможно) во все узлы на пути от самого глубокого узла в $\hat{T}$, являющегося суффиксом s, до узла с меткой s.

(c) если $s<L$, то для каждого $\sigma' \in \Sigma$, если

$$\widetilde{P}(\sigma' s) \geq (1-\varepsilon_1)\varepsilon_0$$

добавить строку $\sigma' s$ в $\overline{S}$.

В работе [10] доказывается следующая основная

Теорема 2. *Для любых PSA M и параметров* $\varepsilon > 0 \quad 0 < \delta < 1$ *алгоритм строит PST* $\hat{T}$ *такое, что с вероятностью как минимум* $1-\delta$, *дерево* $\hat{T}$ — ε *-хорошая гипотеза относительно M. Если алгоритм имеет доступ к источнику независимо сгенерированных строк, то время его работы ограниченно полиномом от* $L, n, |\Sigma|, 1/\varepsilon, 1/\delta$.

Для доказательства этой теоремы существенно используется вспомогательная

Лемма 1. *Существует полином* m_0 *такой, что набор примеров из* $m > m_0(L, n, |\Sigma|, 1/\delta, \varepsilon_0, \varepsilon_1, \varepsilon_2)$ *строк, длиной более (L+1) каждая, является* типичным *с вероятностью, не меньшей* $1-\delta$.

Неформально говоря, набор строк считается типичным, если для каждой сгенерированной *M* подстроки символов ее эмпирическая вероятность и вероятность следующего символа при условии этой подстроки, близки к своим математическим ожиданиям. Или формально,

Определение 4. *Набор строк называется* типичным, *если для каждого* $s \in \Sigma^{\leq L}$ *выполняется следующее:*

1. Если $s \in Q$, *то* $|\widetilde{P}(s) - P(s)| \leq \varepsilon_1\varepsilon_2$

2. Если $P(s) \geq (1-\varepsilon_1)\varepsilon_0$, *то* $\forall \sigma \in \Sigma, |\widetilde{P}(\sigma | s) - P(\sigma | s)| \leq \gamma_{\min}\varepsilon_2$

Надо отметить, что в определении эмпирических вероятностей и, соответственно, в определении 4 вероятность $\widetilde{P}(s)$ в случае нескольких строк является суммой независимых ограниченных случайных величин. Для определения $\widetilde{P}(s)$ берется их арифметическое среднее, то есть все слагаемые добавляются с одинаковыми весовыми коэффициентами $\frac{1}{m'(l - L + 1)}$. Это дает возможность применить известные неравенства для сумм независимых случайных величин.

Так, для доказательства леммы 1 используется вариант неравенства Хеффдинга [11], которое дает оценку сверху вероятности того, что сумма независимых ограниченных случайных величин не превышает своего математического ожидания. А именно,

Теорема 3. *Пусть* $X_1, X_2, ..., X_n$ — *независимые ограниченные случайные величины,* $S = \sum_{i=1}^{n} X_i$. *Тогда для любого* $\varepsilon > 0$

$$P\{| S - MS | \leq n\varepsilon \} \geq 1 - 2e^{2n\varepsilon^2}$$

3. Модификация модели и алгоритм построения PST

3.1. Модификация модели

Особенность нашей задачи состоит в необходимости адаптивного построения и модификации суффиксного дерева. Чтобы снабдить модель определенной адаптивностью, предлагается модифицировать алгоритм вычисления эмпирических вероятностей таким образом, чтобы вклад более ранних примеров в совокупную эмпирическую вероятность с каждым шагом

уменьшался. Тогда более поздние примеры будет учитываться с большими весовыми коэффициентами, и можно будет смоделировать «забывание» ранних примеров. Соответственно необходимо изменить и алгоритм построения дерева.

В качестве класса гипотез выберем сами вероятностные суффиксные деревья.

Зададимся определенным коэффициентом обучения $0 < \alpha < 1$, и положим эмпирическую вероятность состояния с меткой *s* в момент времени *t*

$$\widetilde{P}_1(s) = P_1'(s)$$

$$\widetilde{P}_t(s) = \alpha \widetilde{P}_{t-1}(s) + (1-\alpha)P_t'(s) = \alpha^{t-1}P_1'(s) + (1-\alpha)\sum_{\tau=2}^{t} \alpha^{t-\tau} P_\tau'(s)$$

где $P'_t(s)$ — эмпирическая вероятность состояния в строке, которая поступила в момент времени *t*.

Вычисление вероятностей по таким правилам дает возможность уменьшить влияние более ранних примеров и одновременно учесть с большим весом более свежие. Величина коэффициента α регулирует скорость забывания: значения, близкие к нулю приведут к быстрому забыванию, а значения, близкие к единице — к медленному.

В нашем случае, $\widetilde{P}_t(s)$ — сумма независимых ограниченных случайных величин $P'_t(s)$ с переменными коэффициентами.

Предположим, что в момент времени *t=1* вероятностное суффиксное дерево, по примерам работы которого мы стараемся построить аппроксимацию, изменилось. То есть скачкообразно изменилось распределение вероятностей состояний. Будем считать, что в нашей предметной области смоделирована ситуация, когда пользователь (процесс) сам изменяет свое поведение или мы начали наблюдать поведение другого субъекта системы, который идентифицирован как легитимный пользователь (процесс).

Модифицированный алгоритм должен приспособиться к новому поведению и обеспечить необходимую адаптивность модели, оставляя верными результаты теоремы 2.

Будем обозначать эмпирические вероятности на шаге *t* через X_t. Пусть

$$S_t = \alpha^t X_0 + (1-\alpha)\sum_{\tau=1}^{t} \alpha^{t-\tau} X_\tau \quad .$$

Справедлива следующая теорема.

Теорема 4. *Для любых* $\varepsilon > 0$ *и* $0 < \delta < 1$ *существуют такие* $0 < \alpha < 1$ *и* $t \geq 1$, *что в случае скачкообразного характера изменения распределения величин X*

$$P\{|S_t - MX_t| \leq \varepsilon\} \geq 1-\delta \quad . \qquad (6)$$

Доказательство в основном повторяет общую схему доказательства теоремы 3 из [11].

Из неравенства Чебышева для любого $h \geq 0$, учитывая то, что экспонента является монотонной и неотрицательной функцией, имеем:

$$P\{S_t - MX_t \geq \varepsilon\} \leq \frac{Me^{h(S_t - MX_t)}}{e^{h\varepsilon}}. \qquad (7)$$

Поскольку $MX_t \equiv \alpha^t MX_t + (1-\alpha)\sum_{\tau=1}^{t} \alpha^{t-\tau} MX_t$, то

$$S_t - MX_t = \alpha^t (X_0 - MX_t) + (1-\alpha)\sum_{\tau=1}^{t} \alpha^{t-\tau} (X_\tau - MX_t) \qquad (8)$$

Поскольку e^{hx} — выпуклая функция, то для ограниченной случайной величины *X*, $0 \leq X \leq b$

$$e^{hx} \le \frac{b - X}{b} + \frac{X}{b} e^{hb}.$$

Потому

$$Me^{hx} \le \frac{b - MX}{b} + \frac{MX}{b} e^{hb}. \qquad (9)$$

Тогда из (8) и (9), учитывая, что $b=1$, имеем:

$$Me^{h(S_t - MX_t)} = e^{h\alpha^t (X_0 - MX_t)} \prod_{\tau=1}^{t} e^{h(1-\alpha)\alpha^{t-\tau}(X_\tau - MX_t)} \le$$

$$e^{-h\alpha^t MX_t}(1 - MX_0 + MX_0 e^{h\alpha^t}) \prod_{\tau=1}^{t} e^{-h(1-\alpha)\alpha^{t-\tau} MX_t}(1 - MX_\tau + MX_\tau e^{h\alpha^{t-\tau}(1-\alpha)}).$$

Обозначив MX_i через $\mu_i, i = 0,...,t$ и $b_0 = \alpha^t, b_\tau = (1-\alpha)\alpha^{t-\tau}, \tau = 1,...,t$ получим

$$Me^{h(S_t - \mu_t)} \le \lambda \prod_{\tau=0}^{t} e^{hb_\tau \mu_t}(1 - \mu_\tau + \mu_\tau e^{hb_\tau}). \qquad (10)$$

Рассмотрим одно слагаемое из произведения (10). Пусть

$$L(h_\tau) = -h_\tau \mu_t + \ln(1 - \mu_\tau + \mu_\tau e^{h_\tau}),$$

где $h_\tau = hb_\tau$.

Вычислим две первые производные от $L(h_\tau)$:

$$L'(h_\tau) = -\mu_\tau + \frac{\mu_\tau e^{h_\tau}}{1 - \mu_\tau + \mu_\tau e^{h_\tau}},$$

$$L'(0) = -\mu_t + \mu_\tau ,$$

$$L'(h_\tau) = \frac{\mu_\tau (1-\mu_\tau)e^{h_\tau}}{(1-\mu_\tau+\mu_\tau e^{h_\tau})^2} \le 1/4$$

Тогда из разложения функции $L(h_\tau)$ в ряд Тейлора имеем:

$$L(h_\tau) = L(0) + L'(0)h_\tau + \frac{L''(\xi)h_\tau^2}{2} \le h_\tau(\mu_\tau - \mu_t) + \frac{h_\tau^2}{8}, \qquad (11)$$

где $\xi \in [0, h_\tau]$.

Следовательно, учитывая (11) и неравенства (7) и (10), получаем, что

$$P\{S_t - MX_t \ge \varepsilon\} \le e^{-h_t} \prod_{\tau=0}^{t} e^{hb_\tau(\mu_\tau - \mu_t + \frac{hb_\tau}{9})} .$$

Правая часть (10) достигает минимума по h при $h = \frac{4(\varepsilon - \Delta\mu_t)}{\sum_{\tau=0}^{t} b_\tau^2}$, где $\Delta\mu_t = \mu_t - \sum_{\tau=0}^{t} b_\tau\mu_\tau$. Поэтому

$$P\{S_t - MX_t \ge \varepsilon\} \le \exp\left(-\frac{2(\varepsilon - \Delta\mu_t)^2}{\sum_{\tau=0}^{t} b_\tau^2}\right)$$

$$= \exp\left(-h\varepsilon - h\left(\mu_t - \sum_{\tau=0}^{t} b_\tau\mu_\tau\right) + \frac{1}{8}h^2\sum_{\tau=0}^{t} b_\tau^2\right) . \qquad (12)$$

Из того факта, что $P\{|S_t - MX_t| \ge \varepsilon\} = P\{S_t - MX_t \ge \varepsilon\} + P\{-S_t + MX_t \ge \varepsilon\}$ вытекает оценка

$$P\{|S_t - MX_t| \geq \varepsilon\} \leq \exp\left(-\frac{2(\varepsilon - \Delta\mu_t)^2}{\sum_{\tau=0}^{t} b_\tau^2}\right) + \exp\left(-\frac{2(\varepsilon + \Delta\mu_t)^2}{\sum_{\tau=0}^{t} b_\tau^2}\right).$$

Вычислив сумму $\sum_{\tau=0}^{t} b_\tau^2$, которая равна $\frac{1-\alpha+2\alpha^{2t+1}}{1+\alpha}$, получим

$$P\{|S_t - MX_t| \geq \varepsilon\} \leq \exp\left(-\frac{2(1+\alpha)(\varepsilon - \alpha^t(\mu_0 - \mu))^2}{1-\alpha+2\alpha^{2t+1}}\right) + \exp\left(-\frac{2(1+\alpha)(\varepsilon + \alpha^t(\mu_0 - \mu))^2}{1-\alpha+2\alpha^{2t+1}}\right). \quad (13)$$

Для каждого из двух слагаемых в (13), при условии $\alpha \geq 1 - \frac{\varepsilon^2}{\ln\frac{2}{\delta}}$ (этого достаточно), существуют такие t_1, t_2, что при $t > \max(t_1, t_2)$, каждое из слагаемых ограничено сверху значением $\frac{\delta}{2}$. Тогда их сумма — меньше δ, откуда получаем (6). Теорема доказана.

3.2. Модифицированный алгоритм построения PST

Из-за измененного правила обновления эмпирических вероятностей (5) необходимо изменить алгоритм построения PST таким образом, чтобы он отражал эти изменения.

Модифицированный алгоритм

1. Получить $\hat{T}_{t-1}$ и текущую пример-сессию *r*.

2. Обновить эмпирические вероятности всех узлов дерева согласно правилу (5).

3. Рекурсивно удалить все листы из дерева меток *s,* для которых

$$\widetilde{P}_t(\sigma) < (1 - \varepsilon_1)\varepsilon_0 \ . \tag{14}$$

4. Инициализировать множество $\overline{S}$.

$$\overline{S} = \{s \mid s \in \Sigma^*, suffix(s) \in L(\hat{T}_{t-1}), \widetilde{P}_t(\sigma) \geq (1 - \varepsilon_1)\varepsilon_0\},$$

где $L(\hat{T}_{t-1})$ — множество листьев дерева $\hat{T}_{t-1}$.

5. Пока $\overline{S}$ не пусто, выполнять: выбрать любую строку $s \in \overline{S}$ и

(a) удалить *s* из $\overline{S}$

(b) если существует $\sigma \in \Sigma$, такое что

$$\widetilde{P}_t(\sigma \mid s) \geq (1 + \varepsilon_2)\gamma_{\min} \tag{15}$$

и одновременно

$$\frac{\widetilde{P}_t(\sigma \mid s)}{\widetilde{P}_t(\sigma \mid suffix(s))} > 1 + 3\varepsilon_2, \tag{16}$$

тогда добавить узел соответствующий *s* в дерево и (возможно) все узлы на пути от самого глубокого узла в $\overline{S}$, который является суффиксом *s*, до узла с меткой *s*.

(c) Если $|s| < L$, то для каждого $\sigma' \in \Sigma$, если

$$\widetilde{P}(\sigma' s) \geq (1 - \varepsilon_1)\varepsilon_0,$$

добавить строку с меткой $\sigma' s$ в $\overline{S}$.

Модифицированный алгоритм адаптивно перестраивает дерево согласно изменившимся вероятностями состояний. При этом состояния, которые по обновленным значениями вероятностей стали пренебрежимыми, (см. (14)) удаляются, а состояния, для которых условия (15) и (16) начали выполняться — добавляются.

Литература

1. D.E. Denning. An intrusion-detection model. In *Proc. IEEE Symposium on Security and Privacy*, p. 118-131, 1986.
2. S. Forrest, S.A. Hofmeyr, A. Somayaji, and T.A. Longstaff. A sense of self for Unix processes. In *Proceedings of the 1996 IEEE Symposium on Research in Security and Privacy*, p. 120-128. IEEE Computer Society Press, 1996.
3. *А М. Резник, Н.Н. Куссуль и А.М. Соколов*. Нейросетевая идентификация поведения пользователей компьютерных систем. *Кибернетика и вычислительная техника*, (123): С. 70-79, 1999.
4. T. Lane. Hidden markov models for human/computer interface modeling. In *IJCAI-99 Workshop on Learning About Users*, p. 35-44, 1999.
5. C.C. Michael and A. Ghosh. Two state-based approaches to program-based anomaly detection. *ACM Transactions on Information and System Security*, 5 (2), 2002.
6. D. Wagner and R. Dean. Intrusion detection via static analysis. In F. M. Titsworth, editor, *Proceedings of the 2001 IEEE Symposium on Security and Privacy*, p. 156-169, Los Alamitos, CA, May 14-16 2001. IEEE Computer Society.
7. B.D. Davison and H. Hirsh. Predicting sequences of user actions. In *Predicting the Future: AI Approaches to Time-Series Problems*, p. 5-12, Madison, WI, July 1998. AAAI Press. Proceedings of AAAI-98/ICML-98 Workshop, published as Technical Report WS-98-07.
8. E. Eskin. Anomaly detection over noisy data using learned probability distributions. In *Proc. 17 International Conf. on Machine Learning*, p. 255-262. Morgan Kaufmann, San Francisco, CA, 2000.
9. N. Ye. A markov chain model of temporal behavior for anomaly detection. In *Proceedings of the 2000 IEEE Systems, Man, and Cybernetics, Information Assurance and Security Workshop*, 2000.

10. D. Ron, Y. Singer, and N. Tishby. The power of amnesia: Learning probabilistic automata with variable memory length. *Machine Learning*, 25 (2-3): 117-149, 1996.