Куссуль Н.Н., Соколов А.М.

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

Часть II. Методы обнаружения аномалий и результаты экспериментов

В первой части нашей работы [1] предложена адаптивная модификация модели марковских цепей переменного порядка. В настоящей статье эта модель используется для выявления аномальной деятельности пользователей компьютерных систем. Предложены несколько способов обнаружения аномалий при помощи разработанной модели и новый метод борьбы с replay-атаками.

1. Среда и технология проведения экспериментов

С целью проверки подхода была проведена серия экспериментов. Из-за сложности получения или моделирования искусственных данных со следами вторжений и спорности непосредственной экстраполяции полученных таким путем результатов на реальные ситуации, было решено использовать реальные аудит-файлы сеансов реальных пользователей.

Эксперименты проводились на данных, полученных за год с UNIX сервера физико-технического факультета НТУ Украины «КПИ». Механизмами аудита операционной системы FreeBSD отслеживались все процессы, которые запускались от имени зарегистрированных в системе пользователей, в том числе и псевдо-пользователей (демонов). Таким образом, не было потребности

модифицировать ядро операционной системы, чтобы собрать необходимую статистику, как, например, в [2]. Всего было получено данных для более чем 800 пользователей.

Из доступной информации о процессе мы использовали только имя процесса, не принимая во внимание остальные параметры (время, ресурсные характеристики, наличие прав суперпользователя и тому подобное). Они не играют существенной роли для проверки нашего подхода, однако их учет является необходимым при построении полноценной системы обнаружения аномалий.

Среди множества всех команд (более 900 уникальных имен процессов), которые были использованы в течение периода наблюдения, мы оставили только те, интенсивность использования которых превышала 200 запусков за весь период (350 команд). Все другие редкие команды были заменены одной специальной командой **RARE**. Кроме того, в начале и в конце каждого сеанса работы были вставлены индикаторы **ALPHA** и **OMEGA** соответственно. Таким образом, обучение происходило на примерах вида

{ALPHA|tcsh|pine|who|tin|RARE}

и начиналось с пустого дерева с одним корневым узлом.

Интересно, что если бы мы использовали «полные» марковские цепи с фиксированным порядком, например $L = 5$, то для построения соответствующего автомата понадобилось бы $350^5 \approx 5 \cdot 10^{12}$ состояний, тогда как с использованием PST с таким же параметром L, среднее количество состояний эквивалентного автомата среди всех пользователей не превышает нескольких тысяч.

Для каждого сеансаработы пользователя вычислялась его вероятность по формуле (1) из [1]. Поскольку вероятность — это произведение чисел, меньших единицы, сеансы с меньшим числом команд будут иметь более высокую

вероятность. Поэтому в качестве коэффициента «нормальности» была выбрана величина, которая (при одинаковых множителях) не зависит от длины последовательности:

$$K = \sqrt[N]{P_T^N(r)},$$

где $P_T^N(r)$ — вероятность сеанса r длины N.

2. Примеры обработки сеансов

Для каждого пользователя по модифицированному алгоритму было построено отдельное PST. На рисунках изображен процесс эволюции значения коэффициента K в течение определенного времени (полгода). По оси абсцисс откладывается номер сеанса в порядке поступления, а по оси ординат — значения K. Для всех графиков обучение начиналось с пустого дерева, при $L = 5$ и $\alpha = 0.99$. Исключение составляют графики, показанные на рис.4 и 5, где эти параметры изменяются.

Поведение большинства пользователей имеет более или менее регулярный характер. На рис. 1 изображены некоторые примеры работы для обычных пользователей. Регулярность может объясняться использованием сценариев, которые выполняются от имени пользователя, когда его нет (для UID 7003 выполнялся сценарий создания архивов аудит-информации, которую использовали для этой работы), выполнением большого количества однотипных действий или преимущественным использованием своей учетной записи в системе для проверки поступления почты (UID 40103) и т.д.

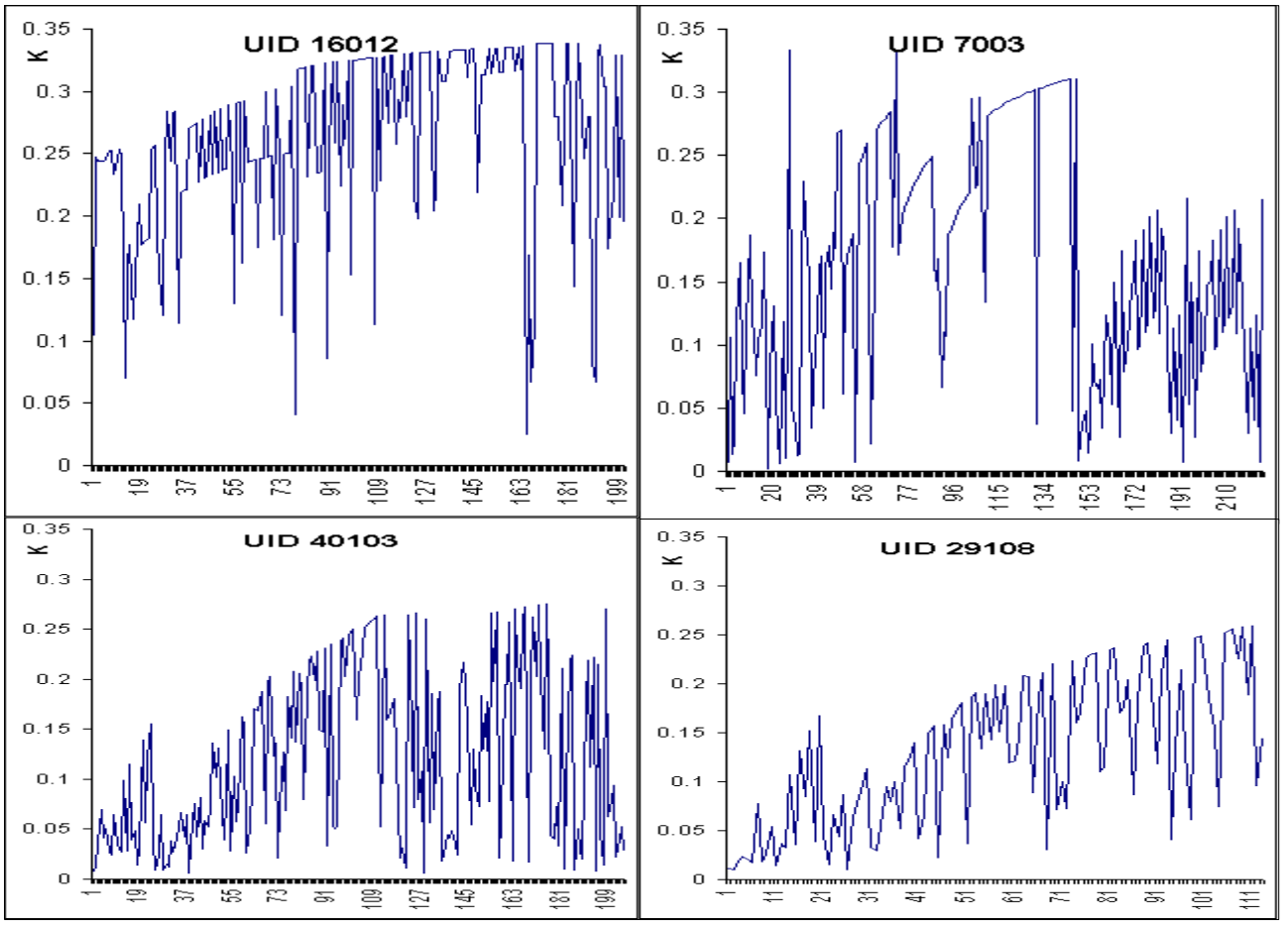

Рис. 1. Эволюция значения коэффициента K для некоторых пользователей

3. Выявление вторжений

3.1. Подмена пользователя

Целью работы системы обнаружения аномалий на уровне пользователей является получение ответа на вопрос: сгенерирован ли текущий сеанс пользователем, от которого получены данные аудита сеанса, или от его имени выступает другое лицо. Распространенным примером вторжений такого вида является ситуация с украденным паролем. На рис. 2 изображены результаты обработки сеанса с умышленно вставленными частями сеансов других пользователей.

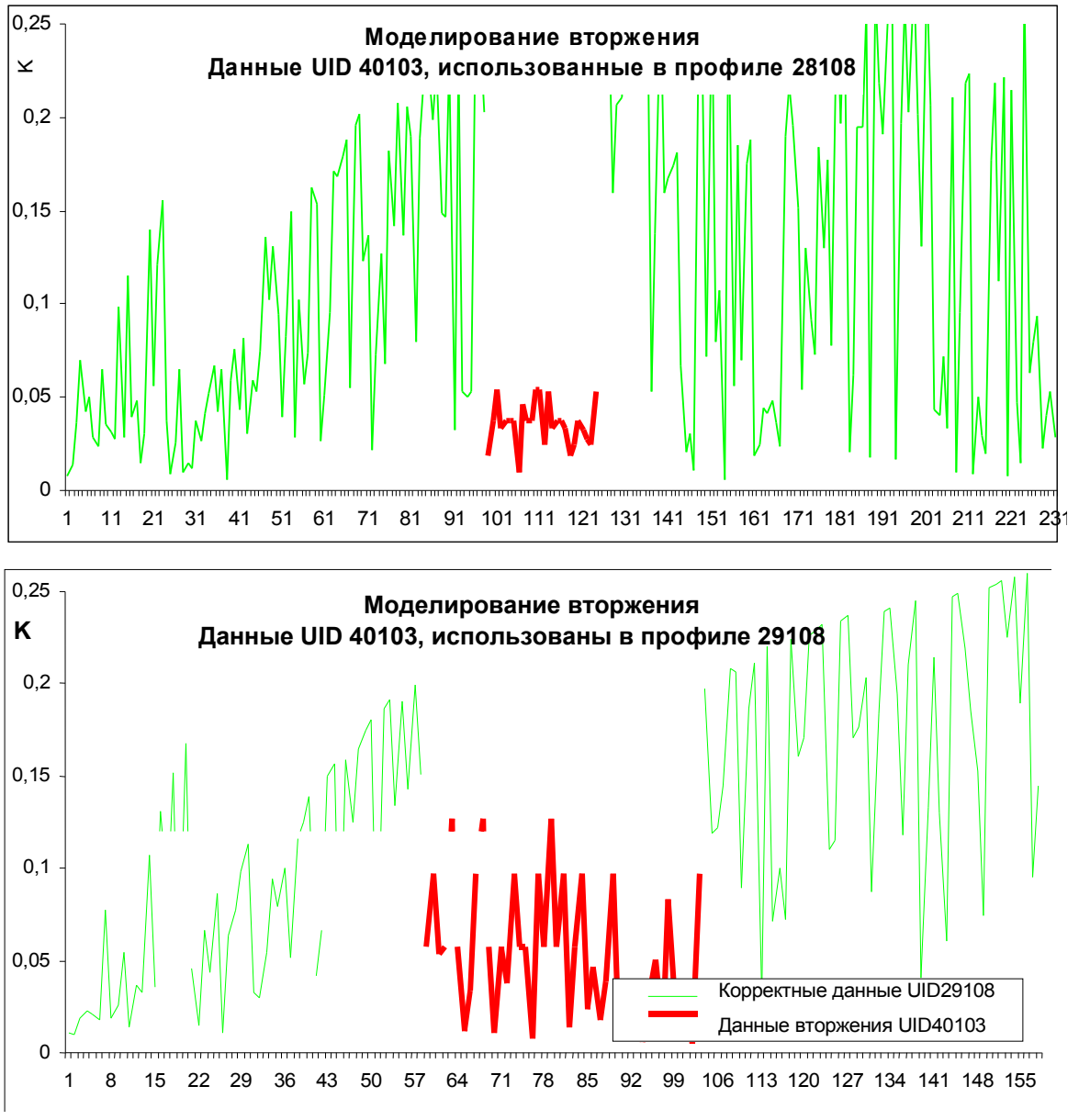

Рис. 2. Примеры подмены пользователя. Моделирование ситуации с украденным паролем

Хорошо видно, что «чужие» сеансы значительно отличаются по значениям *K*. Поэтому можно ввести пороговую классификацию на нормальную или аномальную сеанса. Если значение коэффициента *K* превышает этот порог, сеанс помечается как нормальный, в противном случае — как аномальный. Однако если оба пользователя принадлежат одному «классу», то такая классификация может быть не очень надежной (см. последний график на рис.3; где оба пользователя — студенты с похожими привычками).

3.2. Replay-атаки

Одном из недостатков систем обнаружения аномалий является уязвимость по отношению к replay-атакам. Они характеризуются тем, что злоумышленник, получив доступ к аудит-файлам легального пользователя, «повторяет» его сеанса, заменив незначительную их часть или вставив нужные ему команды. Обычные IDS пропустят такой сеанс как нормальный, поскольку количество добавляемых команд пренебрежимо мало. Одним из подходов к проблеме является внедрение верхнего порога для вероятности текущего сеанса. Считается, что если сеанс «слишком нормален», то он является аномальным [4]. Это не совсем корректно, поскольку, например, системные псевдопользователи (демоны) обычно демонстрируют очень регулярное поведение с высокими вероятностями сеанса. Поэтому сигнализировать о появлении replay-атаки для каждого сеанса с вычисленной высокой вероятностью нецелесообразно, поскольку это приведет к большому количеству ошибок типа «false positives». Более того, даже если атака такого типа и была обнаружена, то, определить, какие из команд были вставлены или изменены таким способом невозможно.

Мы предлагаем новый способ анализа и обнаружения подобных атак, который базируется на возможности преобразования PST в автомат ([13]). Он дает возможность не только выявить модифицированный сеанс, но и выделить из нее «чужие» команды. Идея состоит в том, чтобы рассматривать полученный сеанса как зашумленный и стараться восстановить наиболее вероятную последовательность состояний автомата, через которые он прошел, чтобы сгенерировать его незашумленный вариант.

Эта и подобные задачи сводятся к известной общей задаче разметки структур цепного или более общего типа [13].

Предположим, что экспертным путем получена информация, согласно которой вероятность замены команды в сеансе составляет $0 < \rho < 1$. Интерпретируя PST как автомат, генерирующий последовательности команд, имеем, что, находясь в состоянии k с меткой s, автомат с вероятностью $(1-\rho)P(\sigma^* \mid k)$ перешел в состояние k^{**} с меткой $suffix(s\sigma^*)$, где σ^* — следующая команда в текущем сеансе; или с вероятностью $\frac{\rho}{|\Sigma|-1}P(\sigma' \mid k)$ — в некоторые состояния k' с метками $suffix(s\sigma')$, где $\sigma' \in \Sigma \setminus \{\sigma^*\}$ (рис. 3).

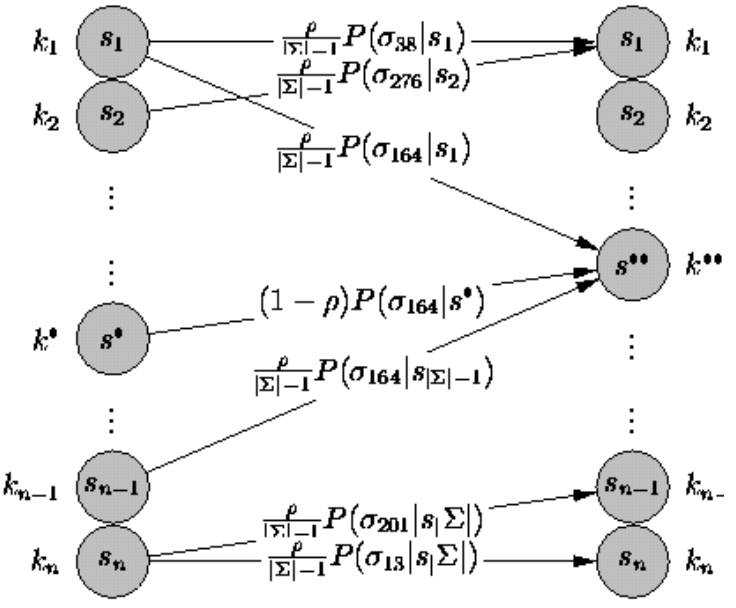

Рис. 3. Иллюстрация к replay-атаке

После построения такого автомата ставится задача найти наиболее вероятную последовательность его состояний. То есть нужно найти:

$$\bar{k} = \arg\max_{k \in \Sigma^N} \sum_{k_0 \in \Sigma} P_0(k_0) \prod_{i=1}^{N} P(x_i, k_i \mid k_{i-1}),$$

где $P_0(k_0)$ — вероятность первой команды, которая определяется из переходных вероятностей узла с меткой **ALPHA**.

Решая эту задачу по известным алгоритмам [3], получаем искомую последовательность состояний $\bar{k}$. Если для некоторого i состояние $\bar{k}_i$ имеет метку s, последний символ которой не совпадает с имеющейся для этого места в сеансе, значит, здесь имела место замена команды.

Можно пойти далее и рассматривать ситуацию, когда вероятно как изменение команды в неизвестном месте, так и вставка новой команды с «раздвиганием» соседних команд. В этом случае алгоритм поиска наиболее вероятной последовательности состояний должен быть обобщен с топологий типа «цепь» на простые сети (подробнее о теории разметок на простых сетях см. [3]).

4. Оценка параметров модели

Работа предложенной адаптивной модели на основе марковских цепей с переменным размером памяти зависит от ряда параметров. Основные из них — это коэффициент обучения α и максимальный порядок L, до которого наращивается PST.

4.1. Коэффициент обучения α

Для определения влияния коэффициента α на процесс адаптации к поведению пользователей мы провели эксперименты с разными значениями α. На рис. 4 изображены результаты обучения для двух пользователей. Как и ожидалось, близкие к единице значения α приводят к медленной адаптации и медленному росту значений коэффициента K, тогда как с меньшими значениями α процесс адаптации ускоряется.

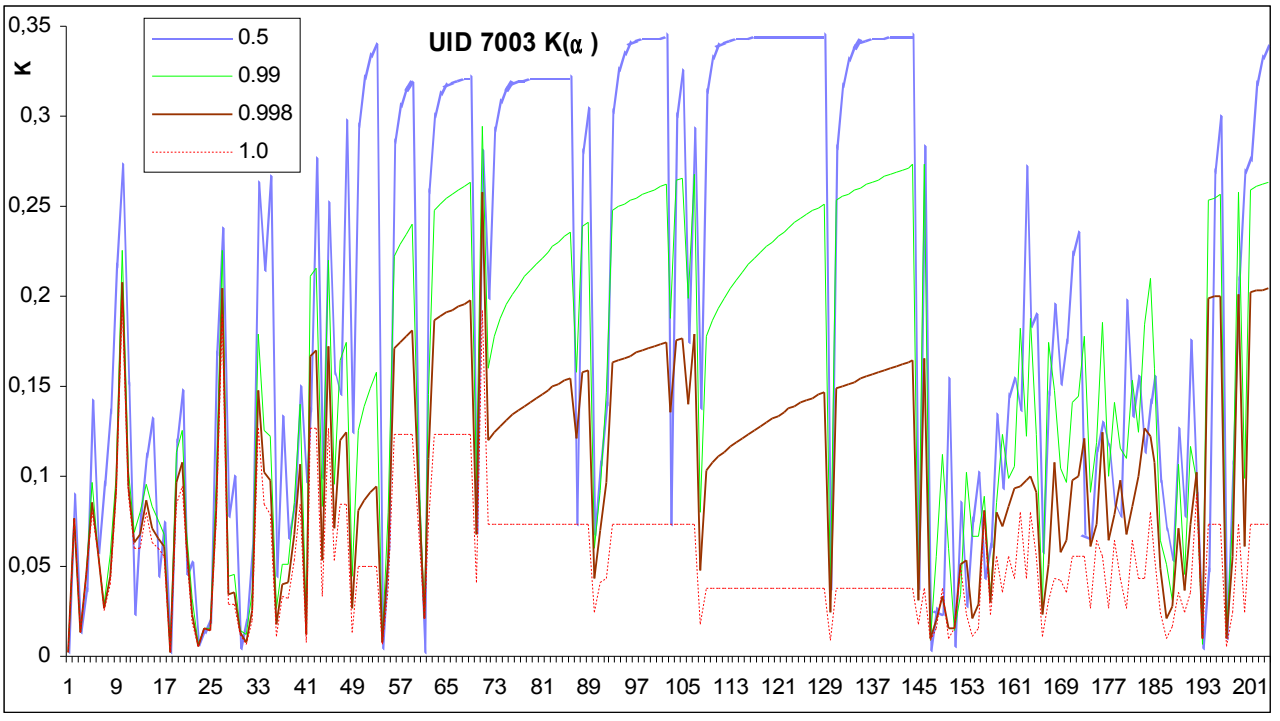

Рис. 4. Зависимость поведения пользователей от α

Выбирая конкретное значение коэффициента α, надо придерживаться индивидуального подхода, поскольку разным людям соответствует разный характер изменений поведения. Важно не задавать весьма малое α, поскольку это может привести к повышению уровня ошибок типа «false negatives», хотя при этом обучение будет происходить быстрее.

4.2. Параметр максимального порядка *L*

Параметр L определяет возможную длину предыдущего контекста. Оптимальным значением L можно считать такое значение, при котором характеристики построенного PST уже не улучшаются. То есть при дальнейшем увеличении L значения коэффициента K не увеличиваются.

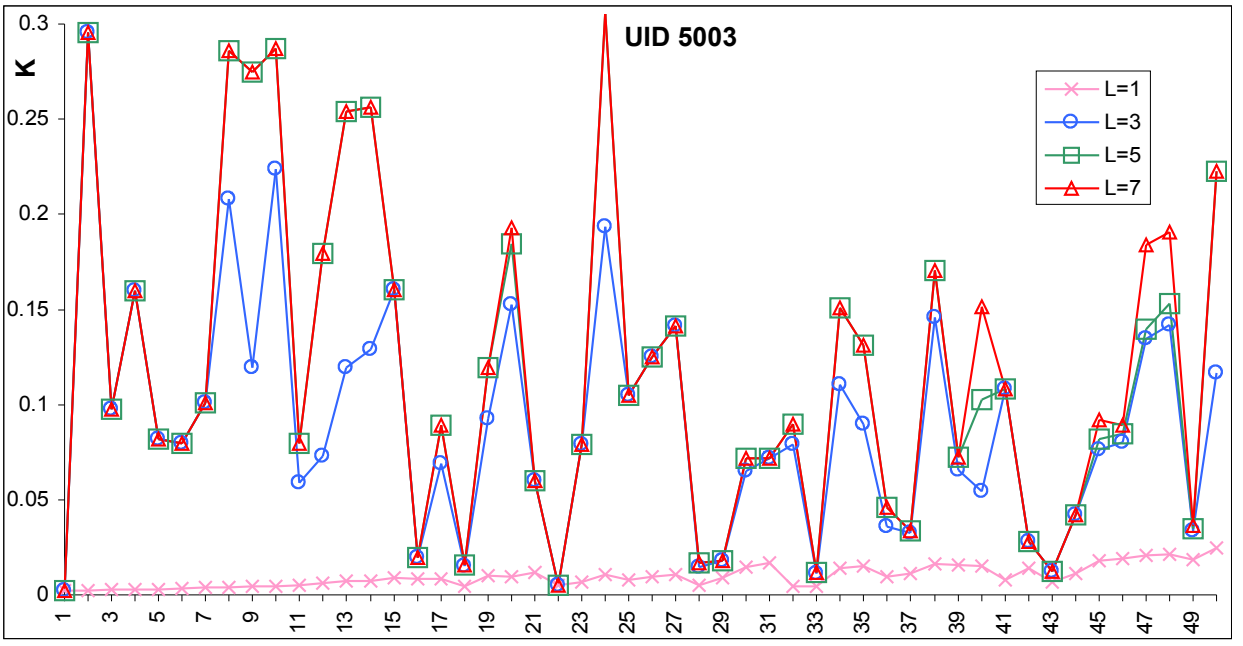

Рис. 5. Зависимость поведения пользователя от параметра *L*

На рис. 5 показано, как влияет постепенное увеличение *L* на значения *K*. Понятно, что наихудшим значением является *L*=1, поскольку тогда для вычисления вероятности сеанса учитывается только относительная доля каждой команды в аудит-файлах. При увеличении *L* наблюдается значительное улучшение результатов обработки сеансов, поскольку более длинный контекст дает возможность точнее уловить статистические закономерности поведения. Исходя из эксперимента, оптимальным значением порядка для данного пользователя можно считать *L=5*, поскольку уже при *L=5* и *L=7* графики почти не отличаются.

Выводы

В работе предложена адаптивная модификация исходной модели марковских цепей переменного порядка. Модифицированная модель может быть применена в различных областях моделирования естественных последовательностей, где статистические характеристики изменяются со временем. Приведен пример ее применения к проблеме обнаружения аномалий

в компьютерных системах. Адаптивная настройка марковской цепи с памятью переменной длины позволяет более точно подстроится к особенностям активных субъектов компьютерной системы и непрерывно уточнять модель, что согласуется с очевидной нестатичностью поведения субъектов.

Из-за невозможности четко формализовать критерии аномальности приходится пользоваться комплексом показателей, по совокупности которых делается окончательный вывод о наличии вторжения. В работе предложено несколько возможных технологий обнаружения аномалий и приведено их экспериментальное обоснование. А именно, предложен перекрестный классификационный подход, метод пороговой классификации и подход, который дает возможность обнаруживать аномальные replay-сеансы и конкретизировать, в чем именно заключается аномальность. Все проведенные эксперименты и предложенные технологии могут быть использованы в реальных системах обнаружения аномалий.

Одним из перспективных направлений дальнейшего исследования является построение адаптивных моделей с использованием ансамблей вероятностных деревьев [4], аналогичных тем, которые используются в нашей работе. Эти деревья, составляя ансамбли, учитываются с определенным весом при вычислении вероятности следующего события в аудит-потоке.

Надо обратить внимание на метод обнаружения аномалий на уровне системных вызовов программ, поскольку это дает возможность не принимать во внимание слишком нерегулярное поведение человека. Однако, при этом не следует полностью отказываться от анализа аудит-данных, исходящих непосредственно от пользователей (например, командные сеансы), поскольку только анализ на этом уровне обеспечит обнаружение некоторых вторжений, которые на уровне системных вызовов не проявляются (например, полностью законное с точки зрения подсистемы аутентификации использование украденного пароля).

Литература

1. 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
2. A. Somayaji. Automated response using system-call delays. In *Usenix Security Syposium 2000*, 2000.
3. M.I. Schlesinger and V. Hlavac. *Deset prednasek z teorie statistickeho a strukturniho rozpoznavani*. Vydavatelstvi CVUT, Praha, 1999.
4. 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.