программа СтегоАссемблер позволяет: внедрять сообщение большого размера, хоть и в определённой пропорции к размеру исходной программы; снимать подозрения в наличии закладки в заполненном контейнере, превращая стегоконтейнер в исполняемый файл с последующей реализацией; выдерживать визуальные стегоатаки, учитывая, что язык ассемблера знает не столь широкий круг программистов.

ЛИТЕРАТУРА

1. *Cox I. J., Miller M. L., and Bloom J. A.* Digital Watermarking. London: Morgan Kaufmann, 2002. 542 p.
2. *Шутько Н. П.* Алгоритмы реализации методов текстовой стеганографии на основе модификации пространственно-геометрических и цветовых параметров текста // Труды БГТУ. Сер. VI: Физ.-мат. науки и информатика. 2016. Вып. 6. С. 160–165.
3. *Блинова Е. А.* Стеганографический метод на основе изменения междустрочного расстояния неотображаемых символов строк электронного текстового документа // Труды БГТУ. Сер. VI: Физ.-мат. науки и информатика. 2016. Вып. 6. С. 166–169.
4. *Ярмолик С. В., Ярмолик В. Н.* Стеганографическая система передачи информации «Stegano-1S» // Технические средства защиты информации: тезисы докл. II Белорусско-российской науч.-технич. конф. (Минск–Нарочь, 17–21 мая 2004 г.) С. 54–62. `http://www.doklady.bsuir.by/m/12_104571_1_56218.pdf`
5. *Абашев А. А., Жуков И. Ю. и др.* Ассемблер в задачах защиты информации. М.: КУДИЦ-ОБРАЗ, 2004. 544 с.

УДК 004.056 DOI 10.17223/2226308X/10/31

О ВЕРИФИКАЦИИ СОБСТВЕННОРУЧНОЙ ПОДПИСИ

А. В. Епишкина, А. В. Береснева, С. С. Бабкин, А. С. Курнев, В. Ю. Лермонтов

Проанализированы способы онлайн-верификации собственноручной подписи на основе KNN-алгоритма, Range Classifier алгоритма, алгоритма на основе скрытой модели Маркова и простейшей перцептронной нейронной сети. Особенности применения данных алгоритмов исследованы в ходе реализации и тестирования с целью дальнейшей модификации и разработки наиболее эффективного по всем параметрам метода верификации.

Ключевые слова: *верификация собственноручной подписи, скрытые модели Маркова, нейронные сети.*

1. Подходы к верификации собственноручной подписи

На данный момент разработано несколько различных подходов к задаче верификации собственноручной подписи. Автономная система проверки подписей, представленная в [1], построена на основе нескольких статистических методов, в частности используются скрытые модели Маркова (СММ) в построении эталонной модели для каждого локального объекта.

Другая система, предложенная в [2], основана на машинном обучении. Для применения машинного обучения при верификации подписи необходима обучающая выборка. В процессе исследования изучалась возможность применения таких алгоритмов, как KNN, метод опорных векторов и метод логистической регрессии.

В [3] описана методика верификации подписи, которая основана на использовании нейронной сети. Для каждого объекта устанавливается специальный двухступенчатый

перцептрон и внедрена структурная классификация. Технология применения нейронных сетей является широко распространённой для решения подобного рода задач.

2. Анализ основных алгоритмов верификации

Преимущества использования динамических признаков в том, что их гораздо сложнее подделать, так как они не видны при рассмотрении бумажной копии подписи. Результаты тестирования алгоритмов верификации подписи представляются в виде соотношения ошибок 1-го и 2-го рода. Ошибки 1-го рода связаны с отказами в доступе законному пользователю, ошибки 2-го рода — ложной идентификацией.

Используемые при дальнейшей обработке характеристики подписи:

— графическое изображение (в графической или векторной форме);
— количество отрывов пера от поверхности устройства;
— временные характеристики (минимальное, максимальное, среднее, полное время без отрыва пера от экрана);
— характеристики скорости перемещения пера (минимальные, максимальные значения проекций скоростей на оси и модуля скорости).

Проанализированы следующие подходы, позволяющие произвести верификацию собственноручной подписи:

— KNN-алгоритм;
— алгоритм Range Classifier;
— алгоритм на основе скрытой модели Маркова;
— простейшая перцептронная нейронная сеть.

KNN-алгоритм классификации [2] (k Nearest Neighbours, k ближайших соседей) на вход принимает вектор, содержащий значения характеристик подписи, а на выходе выдаёт решение, подлинная подпись или подделка. Для классификации каждой из характеристик на основе обучающей выборки необходимо последовательно выполнить следующие операции:

— вычислить расстояние до каждого из объектов обучающей выборки;
— отобрать k объектов обучающей выборки, расстояние до которых минимально.

Далее принимается положительное решение в случае, если характеристики находятся в пределах допустимого отклонения. Данный алгоритм имеет следующие недостатки:

— низкая точность;
— возникновение ошибок 1-го и 2-го рода при поворотах, масштабировании, сдвигах подписи.

Алгоритм верификации подписи Range Classifier состоит из следующих этапов:

— для каждого образца подписи рассчитывается центроид;
— для каждой подписи формируются векторы из углов и длин радиус-векторов от центроида до каждой точки;
— накладываются последовательности векторов тестируемой подписи и объектов обучающей выборки с учётом погрешности;
— определяется диапазон значений каждого вектора согласно обучающей выборке.

Если пороговое число параметров попадает в указанный диапазон и наложение векторов совпадает, подпись признаётся подлинной.

Алгоритм на основе скрытой модели Маркова на вход также принимает вектор из характеристик подписи. Алгоритм состоит из следующих шагов:

— процесс подписания моделируется с несколькими состояниями, которые представляют собой цепь Маркова;
— каждое из этих состояний соответствует отдельной части подписи, которая не наблюдаются непосредственно (то есть скрыта);
— наблюдаемые данные связаны статистически с состояниями модели и условно независимы в каждом состоянии;
— при обучении параметры модели оцениваются по набору, содержащему достоверные подписи.

Во время верификации вычисляется вероятность того, что подпись подлинна. Если эта вероятность достигает установленного порогового значения, подпись принимается, в противном случае отвергается. Этот подход можно рассматривать как статистическое соответствие проверяемой подписи и подписи, построенной на основе скрытой модели Маркова.

Следующий алгоритм: нейронная сеть принимает на входы вектор, содержащий значения характеристик подписи. Сеть имеет 12 входов, 2 скрытых уровня по 6 нейронов каждый и 1 выход; функционирует по принципу «обучение с учителем». Обучение сети и верификация подписи происходит следующим образом:

1) на входы нейронной сети подаются характеристики подписи;
2) с помощью логистической функции активации задаются веса синапсов нейронной сети;
3) подпись признаётся верной, если на выходе нейронной сети значение превышает пороговое.

Данный алгоритм верификации отличается точностью распознавания и нечувствительностью к изменениям масштаба и сдвигах подписи.

3. Реализация и тестирование

Алгоритмы реализованы в виде мобильного приложения для платформы Android на языке Java. В ходе тестирования на выборке из 100 подписей для рассмотренных алгоритмов получены результаты, приведённые в таблице.

Результаты тестирования реализованных алгоритмов

Алгоритм	Ошибки 1-го рода, %	Ошибки 2-го рода, %	Время вычисления, мс
KNN-алгоритм	13	20	3,37
Range Classifier	4	20	5,17
Алгоритм на основе СММ	10	17	4,82
Нейронная сеть	8	12	2,16

Выводы

В ходе исследования выявлено, что наиболее перспективными для дальнейшей работы являются алгоритм на основе СММ и нейронная сеть, так как доля ошибок этих алгоритмов меньше, чем других. В рамках дальнейшего исследования предполагается разработать усовершенствованный алгоритм верификации собственноручной подписи, учитывающий силу нажатия пера, сократить количество ошибок 1-го и 2-го рода и время обучения.

ЛИТЕРАТУРА

1. *Kashi R. S., Hu J., Nelson W. L., and Turin W.* On-line handwritten signature verification using hidden Markov model features // IEEE Proc. 4th Intern. Conf. Document Analysis and Recognition, Ulm, Germany, 1997. P. 253–257.

2. *Beatrice D. and Thomas H.* On-line Handwritten Signature Verification using Machine Learning Techniques with a Deep Learning Approach. Master's Theses in Math. Sciences, Lund University, 2015. 90 p.
3. *McCabe A., Trevathan J., and Read W.* Neural network-based handwritten signature verification // J. Computers. 2008. V. 3. No. 8. P. 9–22.

УДК 519.7 DOI 10.17223/2226308X/10/32

САМОПРОГРАММИРУЕМЫЕ КЛЕТОЧНЫЕ АВТОМАТЫ В КРИПТОГРАФИИ

А. А. Ефремова, А. Н. Гамова

Рассмотрены и реализованы различные виды самопрограммируемых клеточных автоматов. Проведено исследование возможности их применения в качестве генератора псевдослучайных чисел. В результате тестирования получено, что самопрограммируемые клеточные автоматы могут применяться в качестве генератора псевдослучайных чисел в криптографии. Для улучшения криптостойкости данного генератора могут быть предложены следующие методы: 1) учёт значения ячейки не в каждый момент времени, а через разные отрезки; 2) применение техники клеточного программирования для подбора используемых правил; 3) комбинирование одномерных и двумерных клеточных автоматов; 4) увеличение числа ячеек и радиуса окрестности.

Ключевые слова: *клеточный автомат, самопрограммируемый клеточный автомат, генератор псевдослучайных чисел, криптография.*

В задачах криптографии часто применяются псевдослучайные последовательности, которые должны быть неотличимы от истинно случайных по своим статистическим свойствам. Для выработки таких последовательностей используют специальные алгоритмы — генераторы псевдослучайных последовательностей. К настоящему времени разработано большое количество таких алгоритмов, основанных на использовании теории чисел, свойствах различных алгебраических систем, применении конечных (в том числе клеточных) автоматов и т. д.

Впервые клеточные автоматы (КлА) были применены в качестве генератора псевдослучайных чисел (ГПСЧ) С. Вольфрамом [1]. Он использовал однородные одномерные КлА с радиусом окрестности $r = 1$ по правилу 30.

Правило 30 задаётся формулой $s_i(t+1) = s_{i-1}(t) \oplus (s_i(t) \vee s_{i+1}(t))$, где $s_i(t)$ — состояние ячейки i в момент времени t. Правило 30 так называется потому, что 30 — десятичное представление вектора значений соответствующей булевой функции (табл. 1).

Таблица 1

Получение кода описания правила 30

$s_{i-1}(t)s_i(t)s_{i+1}(t)$	111	110	101	100	011	010	001	000
$s_i(t+1)$	0	0	0	1	1	1	1	0

С первого взгляда кажется, что восстановить ключ (начальное состояние автомата) по фрагменту последовательности сложно, но известно наличие успешных атак за приемлемое время. Атаки основываются на восстановлении правых (левых) смежных подпоследовательностей последовательности, полученной с помощью ячейки i.