УДК 621.391.00

О.Г. Король, В.В. Федько, В.В. Огурцов

Харьковский национальный экономический университет имени С. Кузнеца, Харьков

ОЦЕНКА СТАТИСТИЧЕСКИХ СВОЙСТВ ХЕШ-ФУНКЦИЙ С ПОМОЩЬЮ ПАКЕТА NIST STS 800-22

Рассматривается программный пакет NIST STS 800-22, используемый для тестирования генераторов случайных и псевдослучайных чисел, позволяющий оценить статистическую безопасность криптографических примитивов. С помощью данного пакета исследуется статистическая безопасность ключевых хеш-функций, применяемых для формирования MAC-кодов.

Ключевые слова: статистическая безопасность, криптографические примитивы, MAC-коды.

Введение. Потребность Украины в развитой системе национальных платежей сегодня намного выше, чем была раньше. Развитие экономики невозможно без внедрения современной системы денежного обращения и использования эффективных платежных механизмов. Быстрый рост объемов, обрабатываемых данных в современных внутриплатежных системах (ВПС), появление новых форм электронных услуг, стремительное развитие вычислительной техники выдвигают новые требования к надежности и обеспечению безопасности в автоматизированных банковских системах (АБС) – сложных многоуровневых системах критического управления, обеспечивающих важный канал проведения финансовых транзакций.

Для обеспечения целостности и подлинности банковских транзакций в современных банковских системах используются электронные цифровые подписи (ЭЦП) и ключевые хеш-функции [1 – 5]. Проведенный анализ [1 – 3] показал, что современные механизмы обеспечения целостности и подлинности данных в АБС не обеспечивают растущие потребности в криптостойкости и оперативности обработки данных. Для оценки криптостойкости алгоритмов как правило используются различные методы оценки случайности сформированных последовательностей [5 – 10].

Целью статьи является рассмотрение математического аппарата программного пакета NIST STS, методики проведения исследований статистической безопасности криптографических примитивов. Оценка с помощью данного пакета ключевых хеш-функций, используемых в АБС.

Основная часть.

Критерии принятия решений по случайности последовательности. Наиболее распространенным на практике подходом к определению псевдослучайности является эвристический подход. При этом подходе псевдослучайный генератор рассматривается как программа (алгоритм), которая порождает битную последовательность $S = S_0, S_1,..,S_{n-1}$ конечной длины n, которая проходит некоторые особые статистические тесты. Таким образом свойства случайной или псевдослучайной последовательности могут быть охарактеризованы и описаны в вероятностном смысле.

Существует множество тестов, дающих оценку, является ли последовательность случайной. Однако никакой конечный набор тестов не считается достаточным. Кроме того, результаты статистического теста должны интерпретироваться с некоторым предостережением, чтобы избежать неверных выводов по определенному генератору.

Статистический тест формулируется для проверки определенной нулевой гипотезы H_0 о том, что последовательность случайна. С этой нулевой гипотезой связана альтернативная гипотеза H_a, о том, что последовательность неслучайна. Для каждого используемого теста можно сделать вывод о принятии или отклонении нулевой гипотезы, исходя из сформированной генератором последовательности.

Для каждого теста должна быть выбрана адекватная статистика случайности на основе которой может быть принята или отклонена нулевая гипотеза. Согласно предположению о случайности, такая статистика имеет распределение случайных значений. Теоретически для нулевой гипотезы распределение этой статистики определяется математическими методами. С этого эталонного распределения определяется критическое значение. Во время проведения теста рассчитывается значение тестовой статистики. Это значение сравнивается с критическим значением. Если значение тестовой статистики превышает критическое значение, то

нулевая гипотеза для случайности отклоняется. В противном случае нулевая гипотеза принимается.

Проверка статистических гипотез работает благодаря тому, что эталонное распределение и критическое значение зависят и генерируются в соответствии с предыдущим предположением о случайности. Если предположение о случайности – истина, то результат тестовой статистики для нее будет иметь очень низкую вероятность превышения критического значения (например, 0,01). Если расчетное значение тестовой статистики превышает критическое значение (то есть возникает событие с низкой вероятностью), то с точки зрения проверки статистической гипотезы событие с низкой вероятностью не может встречаться. Поэтому, когда расчетное значение тестовой статистики превышает критическое значение, делается вывод, что первое предположение о случайности является ошибочным. В этом случае делается вывод об отклонении H_0 (случайность), и принятии H_α (не случайность). Проверка статистической гипотезы является процедурой генерации выводов, при выполнении которой возможно или принять H_0 (данные случайные) или отклонить H_0 (данные не случайны). Табл. 1 связывает истинное (неизвестное) состояние данных с выводом, полученным процедурой проверки.

Таблица 1

Ситуация	Вывод	
	Принять H_0	Принять H_a
Данные случайны (H_0 – истина)	ошибки нет	ошибка 1-го рода
Данные неслучайны (H_a – истина)	ошибка 2-го рода	ошибки нет

Если данные в действительности случайные, то вывод об отклонении нулевой гипотезы будет приниматься очень редко. Этот вывод имеет название ошибки первого рода. Если данные не случайны, то вывод о принятии нулевой гипотезы (то есть данные случайные) называется ошибкой второго рода.

Вероятность ошибки первого рода называется уровнем значимости теста. Эта вероятность может быть установлена к тестам и обозначается как α. Для теста суть α – вероятность того, что тест покажет не на случайность последовательности, тогда как на самом деле она случайна. То есть на то, что последовательность имеет неслучайные свойства даже когда ее сформировал “хороший” генератор.

Вероятность ошибки второго рода обозначается как β. Для теста β – вероятность того, что тест покажет на случайность последовательности, когда на самом деле она неслучайна. То есть “плохой” генератор сформировал последовательность, которая, как кажется, имеет случайные свойства. В отличие от α ошибка второго рода β не является фиксированным значением. Она может принимать множество различных значений, так как существует множество ситуаций, когда поток данных может быть случайным, и каждая из них выдает разные β. Вычисление ошибки второго рода является более сложным из-за большого количества возможных типов неслучайности.

Для принятия решений о прохождении последовательностью случайных чисел статистического теста используются три основных подхода.

Пусть данная двоичная последовательность $S = \{s_1, s_2, \ldots, s_n\}$, $s_i \in \{0,1\}$ длиной n бит. Необходимо принять решение, проходит ли эта последовательность статистический тест на случайность или нет. Возможны следующие подходы к решению этой задачи.

1. Критерий принятия решения на основе установления предельного уровня. Этот подход базируется на вычислении статистики теста $c(S)$, с ее сравнением с некоторым пороговым уровнем $c_{пор}(S)$. Критерий принятия решения формируется следующим образом*: считается, что двоичная последовательность S не проходит статистический тест каждый раз, когда статистика теста c(S) принимает значение меньше предельного уровня* $c_{пор}(S)$.

Например, при проверке сложности последовательности с использованием теста на основе алгоритма Лемпеля-Зива, для заданной двоичной последовательности S рассчитывается ее сложность $c(S)$. Для того чтобы определить, прошла эта последовательность тест или нет, необходимо сравнить полученное значение $c(S)$ с предельным значением $n/\log_2 n$ [9]. Однако такой подход не является достаточно надежным. Как показали практические исследования [10], использование такого критерия часто приводит к ошибочным решениям.

2. Критерий принятия решений на основе установления фиксированного доверительного интервала. При таком подходе критерий принятия решения формируется следующим образом: *считается, что двоичная последовательность S не проходит статистический тест, если значение статистики теста c(S) находится за границей доверительного интервала значений статистики, вычисленного для установленного уровня значимости* α. Например, пусть в двоичной последовательности S длиной n = 800 бит применяется

частотный тест. Значение статистики теста *c(S)* является число единиц в последовательности *S*, причем ожидается, что в последовательности будет примерно 400 единиц и 400 нулей. Если зафиксировать уровень значимости на уровне 5% (α = 0,05), то последовательность *S* не пройдет частотного теста, если число единиц будет находится за пределами доверительного интервала $400 \pm 1{,}96/2 \times \sqrt{800} = [373{,}427]$.

Данный критерий по сравнению с первым является более надежным. Необходимо только учитывать, что разным уровням значимости будут соответствовать различные доверительные интервалы.

3. Третий подход построения критерия принятия решения опирается на расчет для статистики теста *c(S)* соответствующего значения вероятности $P-value$. Статистика теста рассматриваются как реализация случайной величины и подчиняется известному закону распределения. Статистика теста строится таким образом, чтобы ее меньшие значения указывали на любой дефект в случайности последовательности. Значение вероятности $P-value$ – вероятность того, что статистика теста примет значение больше значения, полученного при испытании последовательности на случайность. Таким образом малые значения ($P-value$ < 0,05 или $P-value$<0,01) интерпретируются как доказательство того, что последовательность неслучайна. *Решающее правило формируется так*: для фиксированного уровня значимости α, двоичная последовательность *S* не проходит статистический тест, если значение вероятности $P-value<\alpha$.

Значение α рекомендуется выбирать из интервала [0,001, 0,01]. Значение α, равное 0.001, говорит о том, что из 1000 случайных последовательностей не прошла бы тест только одна. При $P-value\geq 0.001$ последовательность рассматривается как случайная с доверием 99,9%. При $P-value<0.001$ последовательность рассматривается как неслучайная с доверием 99,9%.

Значение α, равное 0.01, говорит о том, что из 100 случайных последовательностей не прошла бы тест только одна. При $P-value\geq 0.01$ последовательность рассматривается как случайная с доверием 99%. При $P-value<0.01$ последовательность рассматривается как неслучайная с доверием 99%.

Одной из основных целей тестов в третьем подходе является минимизация вероятности ошибки второго рода, иначе говоря, минимизация вероятности принятия последовательности, сформированной “плохим” генератором за последовательность, сформированную “хорошим” генератором. Вероятности α, β связаны друг с другом и с длиной *n* последовательности, если два из этих значений определены, третье определяется автоматически. На практике обычно выбирают размер *n* и значение для α (вероятности ошибки первого рода). Тогда критическая точка выбирается таким образом, чтобы получить наименьшее значение (вероятность ошибки второго рода) [6 – 8].

Таким образом, на сегодня основным подходом, который можно использовать при исследовании свойств является эвристический подход. Именно этот подход в дальнейшем используется в наших исследованиях. Одной из важных задач применения эвристического подхода на практике является обоснование набора статистических тестов. Состав тестов зависит от назначения генератора и способов использования последовательностей.

С философской точки зрения не существует возможности эвристическими методами доказать случайность последовательности. В этом случае конкретная последовательность должна проходить континуум тестов. Но и здесь нельзя утверждать, что такая последовательность случайна, потому что возможно создание нового теста, по которому она уже не пройдет испытаний. Благодаря этой причине нельзя так же и построить универсальные тесты, например универсальный тест Маурера.

Набор тестов NIST STS 800-22 был предложен в ходе проведения конкурса на новый национальный стандарт США блочного шифрования. Этот набор использовался для исследований статистических свойств кандидатов на новый блочный шифр. На сегодня методика тестирования, предложенная Национальном институтом стандартов и технологий США NIST, является наиболее распространенной у разработчиков криптографических средств защиты информации.

Пакет включает в себя 16 статистических тестов, которые разработаны для проверки гипотезы о случайности двоичных последовательностей произвольной длины. Фактически, в зависимости от входных параметров вычисляется 189 значений вероятности, которые можно рассматривать как результат работы отдельных тестов. Все тесты направлены на выявление различных дефектов случайности. Основным принципом тестирования является проверка нулевой гипотезы H_0, заключающейся в том, что тестируемая последовательность является случайной.

Альтернативной гипотезой H_α является гипотеза о том, что тестируемая последовательность не случайна. По результатам применения каждого теста нулевая гипотеза либо принимается, либо

отвергается. Решение о том, что будет ли заданная последовательность нулей и единиц случайной или нет принимается по совокупности результатов всех тестов. Таким образом, в результате тестирования двоичной последовательности формируется вектор значений вероятности. Анализ составляющих данного вектора позволяет указать на конкретные недостатки случайности последовательности тестируемого.

Порядок тестирования отдельной двоичной последовательности S выглядит следующим образом [6]:

1. Выдвигается нулевая гипотеза H_0 – предположение о том, что данная двоичная последовательность S случайна.

2. По последовательности S вычисляется статистика теста $c(S)$.

3. С использованием специальной функции и статистики теста вычисляется значение вероятности $P = f(c(S)), P \in [0, 1]$.

4. Значение вероятности P сравнивается с уровнем значимости α, $\alpha \in [0{,}001, 0{,}01]$. Если $P \geq \alpha$, то гипотеза H_0 принимается. В противном случае принимается альтернативная гипотеза.

Как уже было сказано, пакет включает в себя 16 статистических тестов. Но фактически, в зависимости от входных параметров вычисляется 189 значений вероятности P, которые можно рассматривать как результат работы отдельных тестов. В табл. 2 приводятся сводные данные по всем тестам с указанием количества вычисляемых значений вероятности P, физического смысла статистики теста и дефекта, на выявление которого направлен тест.

Таблица 2

Сводные данные по тестам

№ п/п	Статистический тест	Количество значений вероятности P	Статистика теста $c(S)$	Выявляемый дефект
1	Частотный (монобитный) тест	1 (1)	Нормализованная абсолютная сумма значений элементов последовательности	Слишком много нулей или единиц в последовательности
2	Частотный тест внутри блока	1 (2)	Мера согласования наблюдаемого количества единиц внутри блока с теоретически ожидаемым.	Локализованные отклонения частоты появления единиц в блоке от идеального значения S.
3	Проверка накопленных сумм	2 (3-4)	Максимальное отклонение значения накопленной суммы элементов последовательности от начальной точки отсчета (точка 0)	Большое значение единиц или нулей вначале, или в конце двоичной последовательности.
4	Проверка серий	1 (5)	Общее количество серий на всей длине последовательности.	Слишком быстрая или слишком медленная перемена знака в ходе генерации последовательности.
5	Проверка максимальной длины серии в блоке	1 (6)	Мера согласования наблюдаемого значения максимальной длины единичной серии с теоретически ожидаемым значением.	Отклонение от теоретического закона распределения максимальных длин серий единиц.
6	Проверка ранга двоичной матрицы	1 (7)	Мера согласования наблюдаемого значения рангов различного порядка с теоретически ожидаемым.	Отклонение эмпирического закона распределения значений рангов матрицы от теоретического, что указывает на зависимость символов в последовательности.
7	Спектральный тест на основе дискретного преобразования Фурье	1 (8)	Нормализованная разница между наблюдаемым и ожидаемым количеством частотных компонент, которые превышают 95% порогового уровня.	Выявление периодических составляющих (трендов) в двоичной последовательности.

Окончание табл. 2

№ п/п	Статистический тест	Количество значений вероятности P	Статистика теста c(S)	Выявляемый дефект
8	Проверка перекрывающихся шаблонов	1 (9)	Мера согласования наблюдаемого количества перекрывающихся шаблонов в последовательности с теоретическим значением.	Большое количество *m*- битных серий из единиц в последовательности.
9	Универсальный тест Маурера	1 (10)	Сумма логарифма расстояния между *l*-битными шаблонами.	Сжимаемость последовательности.
10	Энтропийный тест	1 (11)	Мера согласования наблюдаемого значения энтропии источника с теоретически ожидаемым для случайного источника.	Неравномерность распределения *m*-битных слов в последовательности (регулярность свойств источника).
11	Проверка случайных отклонений	8 (12-19)	Мера согласования наблюдаемого количества визитов при случайном блуждании в заданное состояние внутри цикла с теоретически ожидаемым.	Отклонение от теоретического закона распределения визитов в конкретное состояние при случайном блуждании.
12	Проверка случайных отклонений (вариант)	18 (20-37)	Общее количество визитов в заданное состояние при случайном блуждании	Отклонение от теоретического ожидаемого общего количества визитов при случайном блуждании в заданное состояние.
13	Последовательный тест	2 (38-39)	Мера согласования наблюдаемого количества всех встретившихся вариантов *m*-битных шаблонов с теоретически ожидаемым.	Неравномерность распределения *m*-битных слов в последовательности.
14	Проверка сжатия по алгоритму Лемпеля-Зива	1 (40)	Количество непересекающихся различных слов в последовательности.	Большая степень сжатия тестируемой последовательности по сравнению с ожидаемой степенью сжатия для случайной последовательности.
15	Проверка неперекрывающихся шаблонов	148 (41-188)	Мера согласования наблюдаемого количества непериодических шаблонов в последовательности с теоретическим значением.	Большое количество заданных непериодических шаблонов в последовательности.
16	Проверка линейной сложности	1 (189)	Мера согласования наблюдаемого количества событий, заключающихся в появлении фиксированной длины эквивалентного ЛРР для заданного блока с теоретически ожидаемым.	Отклонение эмпирического распределения длин эквивалентных ЛРР для последовательности фиксированной длины от теоретического закона распределения для случайной последовательности, что указывает на недостаточную сложность тестируемой последовательности.

Методика NIST STS может применяться как средство комплексного контроля. Выбор методики обусловлен тем, что он содержит необходимый набор статистических тестов, совокупность которых обоснована, и предлагает критерии принятия решения относительно не только отдельной последовательности, но и в отношении всего ГСЧ.

Существенным недостатком является отсутствие официальной версии пакета NIST STS под ОС Windows 7, 8, 10 (пакет разработан в 2000

под ОС Windows 2000/XP). Отдельные тесты могут использоваться и в реальном времени.

Методика тестирования хеш-функций с помощью пакета NIST STS:

1. Для каждой хеш-функции необходимо оценить и принять решение о том, что сформированные хеш-функции являются случайными бинарными последовательностями, и удовлетворяют критериям случайности. Генератор должен формировать двоичную последовательность произвольной длины *n*.

2. Для фиксированного значения *n* формируют множество из *m* двоичных последовательностей:

$$S_1 = s_1, s_2, .., s_n$$
$$S_2 = s_1, s_2, .., s_n$$
$$\ldots\ldots\ldots\ldots\ldots .$$
$$S_m = s_1, s_2, .., s_n$$

Таким образом, для тестирования необходимо сформировать выборку объемом $N = m \times n$.

3. Каждую последовательность проверяют с использованием пакета NIST STS. В результате формируется хеш-функция H (табл. 3).

Таблица 3

Статистический портрет хеш-функции *H*

номер последовательности *i*	номер теста *j*			
	1	2	…	q
S_1	$P_{1,1}$	$P_{1,2}$	…	$P_{1,q}$
S_2	$P_{2,1}$	$P_{2,2}$	…	$P_{2,q}$
⋮	⋮	⋮	…	⋮
S_m	$P_{m,1}$	$P_{m,2}$	…	$P_{m,q}$

Статистическим портретом хеш-функции *H* является матрица размерностью $m \times q$, где m – количество двоичных последовательностей, проверяемых а, q – количество статистических тестов, используемых для тестирования каждой последовательности. Элементы матрицы $P_{ij} \in [0,1]$ где $i = \overline{1,m}$, $j = \overline{1,q}$ представляют собой значение вероятности, полученной в результате тестирования *i*-й последовательности *j*-м тестом.

4. Полученным статистическим портретом определяют судьбу последовательностей, которые прошли каждый статистический тест. Для этого задают уровень значимости $\alpha \in [0{,}001,\ 0{,}01]$ и осуществляют подсчет значений вероятностей, превышающих установленный уровень α для каждого из q тестов, то есть определяют коэффициент

$$r_j = \frac{\#\left\{P_{ij} \geq \alpha \middle| i = 1,2,\ldots,m\right\}}{m}.$$

В результате формируется вектор коэффициентов $R = \{r_1, r_2, ..., r_q\}$, элементы которого характеризуют в процентах прохождение последовательностью S_i всех статистических тестов.

Правило 1. Считается, что хеш-функция *H* прошла тестирование по *j*-му тесту, если значение коэффициента r_j находится в пределах доверительного интервала $[r_{max}, r_{min}]$. Границы доверительного интервала определяются в соответствии выражения:

$$r_{max(min)} = \hat{p} \pm 3\sqrt{\frac{\hat{p}\left(1-\hat{p}\right)}{m}},$$

где $\hat{p} = 1 - \alpha$.

5. Осуществляется статистический анализ статистического портрета. Полученные значения вероятностей P_{ij} должны подчиняться равновероятному закону распределения на интервале $[0,1]$. Для каждого вектора-столбца статистического портрета строится гистограмма частоты F_k попадания значений в каждый из k = 1, 2, ..., 10 подинтервалов, на которые разбит интервал [0, 1]. Равновероятность распределения значений вероятностей P_{ij} проверяется с использованием критерия χ^2. Для этого рассчитывается статистика вида:

$$\chi_j^2 = \sum_{k=1}^{10} \frac{\left(F_k - m/10\right)^2}{m/10},$$

которая подчиняется распределению χ^2 с девятью степенями свободы.

Правило 2. Считается, что хеш-функция *H* прошла тестирование по *j*-му тесту, если выполняется условие $P(\chi_j^2) > 0{,}0001$.

6. Окончательное решение принимают в соответствие с *правилом*: считается, что хеш-функция *H* прошла статистическое тестирование пакетом NIST STS, если значения коэффициентов r_j для всех $j = \overline{1,q}$, находящихся в пределах доверительного интервала $[r_{max}, r_{min}]$ и выполняется условие $P(\chi_j^2) > 0{,}0001$ для всех $j = \overline{1,q}$.

Общий вид и рабочее окно приложения приведены на рис. 1 – 2.

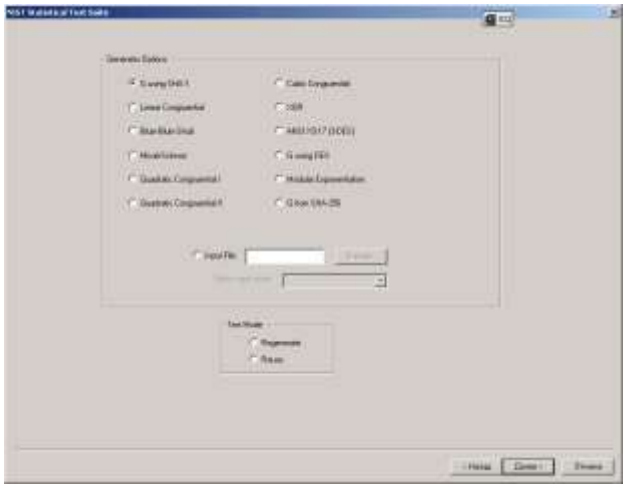


Рис. 1. Общий вид приложения

Рассмотрим более подробно элементы управления приложением.

Существует возможность протестировать уже существующую последовательность, которая сохраняется в файле, выбрав переключатель Входной файл (Input File), или сформировать новую с помощью генераторов, для это необходимо выбрать один из переключателей в группе Опции генератора (Generator Options).

В табл. 4 приведен перечень возможных для использования генераторов.

Таблица 4

№	Английское название	Русское название
1	G using SHA-1	Генератор использующий SHA-1
2	Linear Congruential	Линейный конгруэнтный генератор
3	Blum-Blum-Shub	Генератор Блюм-Блюма
4	Micali-Schnorr	Генератор Micali-Schnorr
5	Quadratic Congruential I	Квадратичный конгруэнтный генератор (1 вариант)
6	Quadratic Congruential II	Квадратичный конгруэнтный генератор (2 вариант)
7	Cubic Congruential	Кубичный конгруэнтный генератор
8	XOR	Генератор, применяющий XOR
9	ANSI X9.17 (3-DES)	Генератор на основе тройного DES
10	G using DES	Генератор использующий DES
11	Modular Exponentiation	Ступенчатый модульный генератор
12	G from SHA-256	Генератор на основе SHA-256

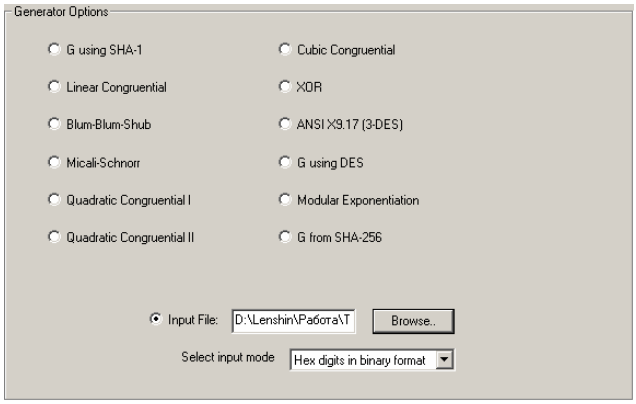


Рис. 2. Рабочее окно приложения

После выбора способа, с помощью которого была получена последовательность, необходимо выбрать в каком виде будет представлена эта последовательность бит в ASCII формате или в бинарном формате 16-разрядной последовательности. Следующим шагом является указание режима, в котором будет функционировать тест, – или регенерация (Regenerate) или повторное использование (Reuse) рис. 3.

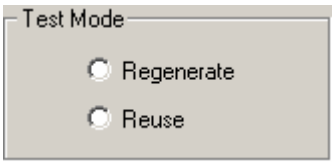


Рис. 3. Выбор режима тестирования

Назначение следующего диалогового окна – выбор тестов, которые будут применяться и внесения заданий их параметров (рис 4).

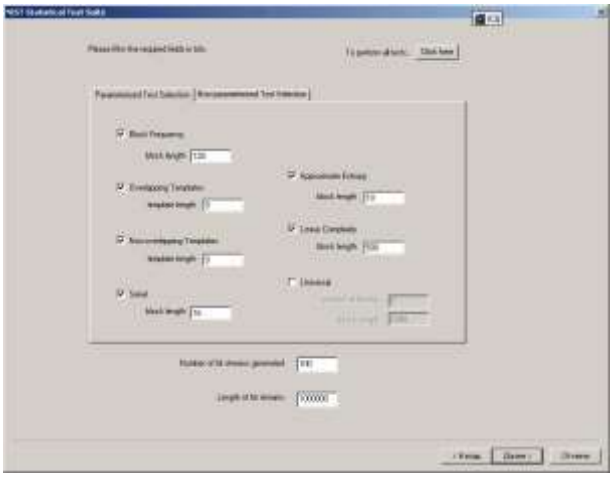


Рис. 4. Выбор и определение параметров для тестов с параметрами

Всего по методике NIST STS случайные и псевдослучайные последовательности тестируются

на соответствие 16 тестами. Но учитывая то, что некоторые тесты могут запускаться с различными параметрами, общее количество тестов равно 189. Все тесты можно условно разделить на тесты с параметрами (Parameterized) и тесты без параметров (Non-parameterized). В средстве тестирования эти тесты размещены на двух разных вкладках. Для параметризированных тестов вводятся значения для тестирования (см. рис. 5), для непараметризированных тестов для активации ставится флажок (рис. 6).

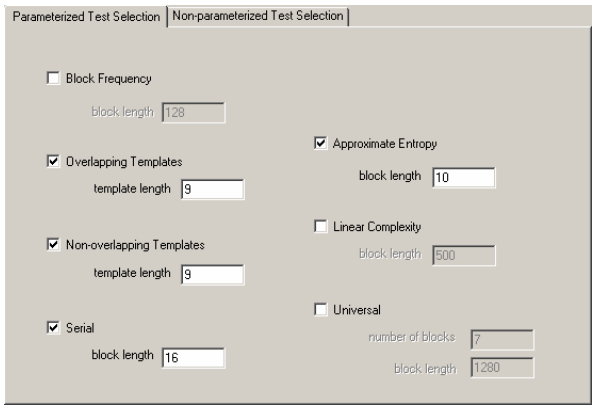


Рис. 5. Выбор и определение параметров для тестов с параметрами

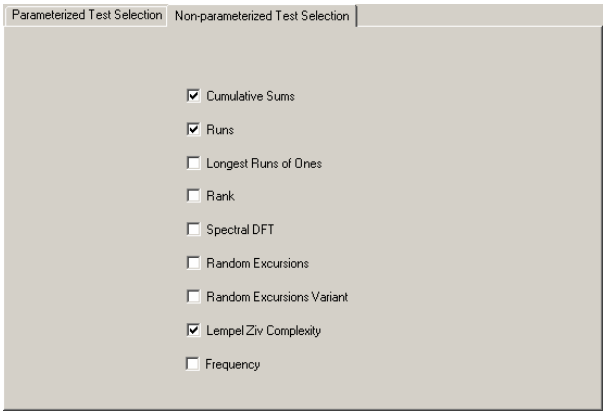


Рис. 6. Выбор и определение параметров для тестов без параметров

После выбора тестов, необходимо в соответствующих элементах управления определить длину последовательности (Length of bit stream) и количество последовательностей (Number of bit streams generated). Нажатие кнопки “Далее” приводит к выводу на экран окна о выбранных настройках тестирования (рис. 7).

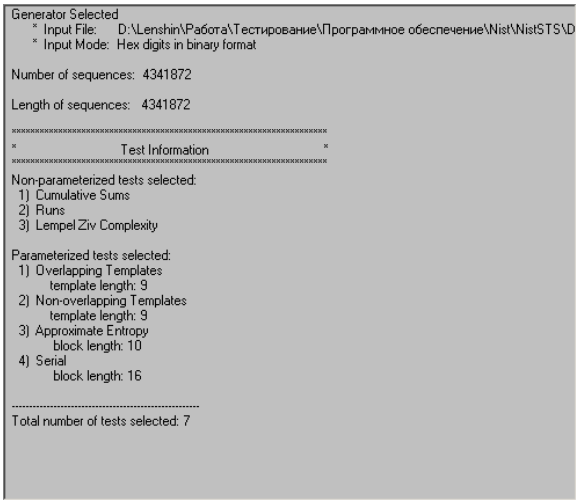


Рис. 7. Показ настроек тестирования

Следующее диалоговое окно предназначено для отображения состояния процесса тестирования последовательности на случайность. Тестирование запускается нажатием кнопки (Run Tests).

Следует отметить, что те тесты, которые не были избраны сразу, отмечаются, как выполненные, а индикатор процесса работает только для избранных тестов (рис. 8).

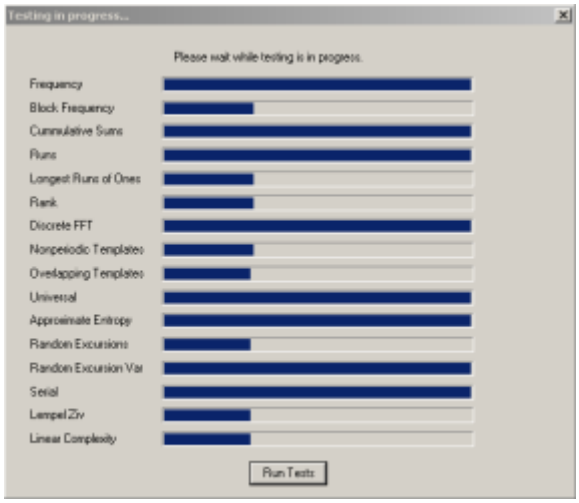
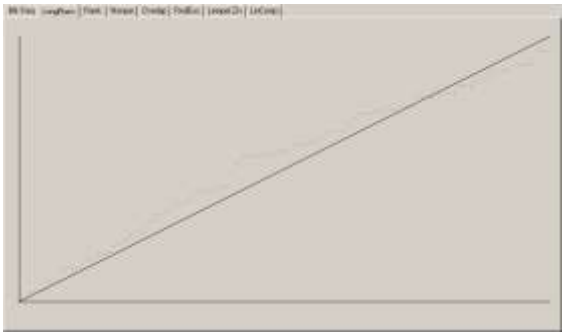


Рис. 8. Процес выполнения тестов

После окончания процесса тестирования для каждого теста строится теоретическая и эмпирическая функция распределения (рис. 9). Функция строится с предположением, что случайная величина распределена по равномерному закону.



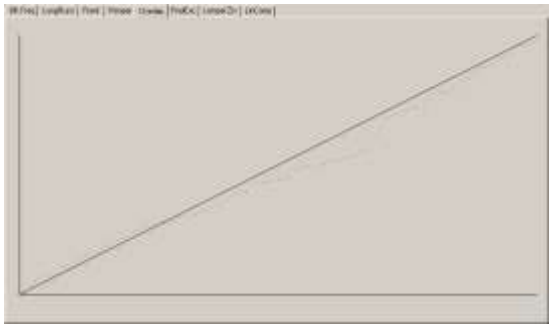
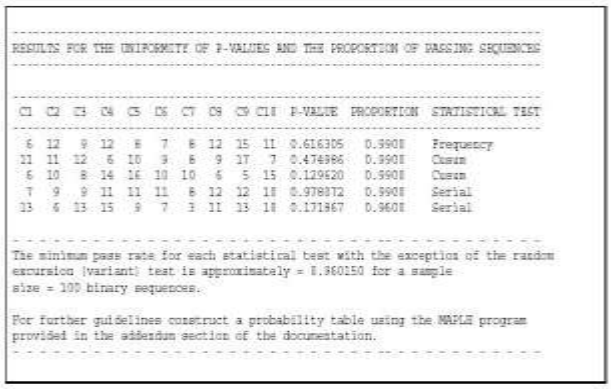


Рис. 9. Построение функции распределения

После окончания работы программы все суммарные расчетные данные размещаются в корневом каталоге в файле finalAnalysisReport. На рис. 10 представлен типичный отчет программного модуля по тестированию псевдослучайной последовательности путем применения частотного теста (frequency test), теста сумм накопления (cusum test) и теста серий (serial test).



```
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES

 C1  C2  C3  C4  C5  C6  C7  C8  C9 C10  P-VALUE  PROPORTION  STATISTICAL TEST
  6  12   9  12   8   7   8  12  15  11 0.616305   0.9900    Frequency
 11  11  12   6  10   9   6   9  17   7 0.474986   0.9900    Cusum
  6  10   8  14  16  10  10   6   5  15 0.129620   0.9900    Cusum
  7   9   9  11  11  11   8  12  12  18 0.978072   0.9900    Serial
 13   6  13  15   9   7   3  11  13  10 0.171867   0.9600    Serial

The minimum pass rate for each statistical test with the exception of the random
excursion (variant) test is approximately = 0.960150 for a sample
size = 100 binary sequences.

For further guidelines construct a probability table using the MAPLE program
provided in the addendum section of the documentation.
```

Рис. 10. Отчет программного модуля

В случае если последовательность не прошла тест, у его названия отображается звездочка.

Для проведения экспериментальных исследований статистической безопасности алгоритмов хеширования использовались программные макеты алгоритмов формирования хеш-кодов MASH, MASH-1, MASH-2 с использованием арифметики эллиптических кривых (MASH (EC)), HMAC-SHA-256, EMAC и UMAC-32.

Для проведения тестирования были взяты следующие параметры:

- длина тестируемой последовательности $n = 10^6$ бит;
- количество тестируемых последовательностей $m = 100$;
- уровень значимости $\alpha = 0{,}01$.

Таким образом, статистический портрет генератора (счетчика) содержит 18900 значений вероятности P. В идеальном случае при $m = 100$ и $\alpha = 0{,}01$ в ходе тестирования может быть отвергнута только одна последовательность из ста, то есть коэффициент прохождения каждого теста должен составлять 99%. Но это слишком жесткое правило. Поэтому, согласно методике, применяется правило на основе доверительного интервала для r_j. Нижняя граница в этом случае составит значение $r_{min} = 0{,}96015$.

На рис. 11 представлен статистический портрет программной реализации ключевой хеш-функции MASH-1

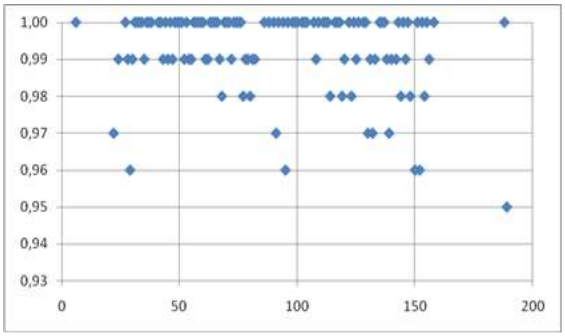


Рис. 11. Статистический портрет программной реализации ключевой хеш-функции MASH-1.

Анализ рис. 11 показывает, что хеш-функция MASH-1 в целом соответствует требованиям безопасности, но одна атака на данный алгоритм позволяет получить положительный результат, что свидетельствует о неслучайности сформированной хеш-функцией последовательности и соответственно ее криптостойкости.

На рис. 12 представлен статистический портрет программной реализации ключевой хеш-функции MASH-2.

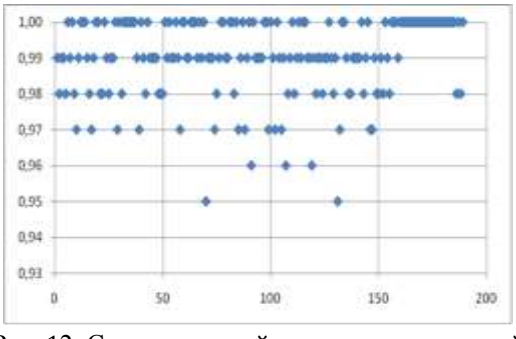


Рис. 12. Статистический портрет программной реализации MASH-2

Этот алгоритм хеширования в целом имеет хорошие статистические свойства безопасности, но два теста имеют результаты вероятности прохождения теста ниже нижней границы.

На рис. 13 представлен статистический портрет программной реализации алгоритма блочного симметричного шифрования MASH (EC) в режиме счетчика.

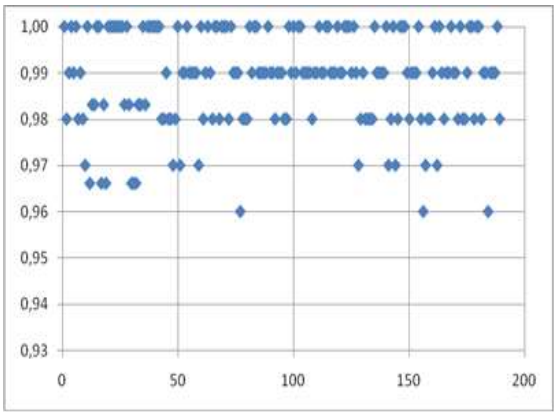


Рис. 13. Статистический портрет программной реализации MASH (EC)

Анализ рис. 13 показывает, что использование арифметики эллиптических кривых позволяет существенно повысить устойчивость ключевой хеш-функции MASH.

На рис. 14 представлен статистический портрет программной реализации ключевой хеш-функции EMAC. Анализ рис. 14 показывает, что ключевая хеш-функция EMAC соответствует предъявляемым требованиям по криптостойкости.

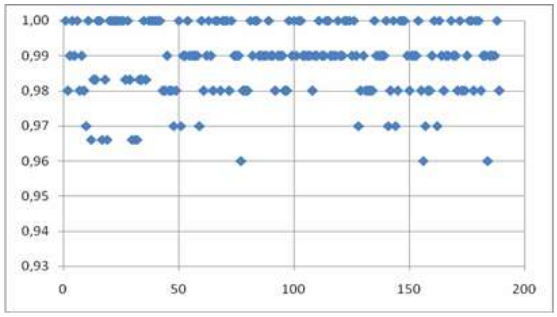


Рис. 14. Статистический портрет программной реализации EMAC

На рис. 15 представлен статистический портрет программной реализации алгоритма блочного симметричного шифрования SHA-1 в режиме счетчика.

Анализ рис. 15 показывает, что хеш-функция SHA-1 формирует хеш-коды, не обеспечивающие криптостойкость.

Статистический портрет функции универсального хеширования UMAC-32 приведен на рис. 16.

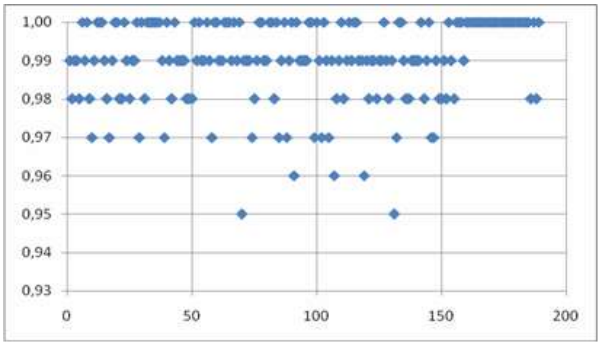


Рис. 15. Статистический портрет программной реализации SHA-1

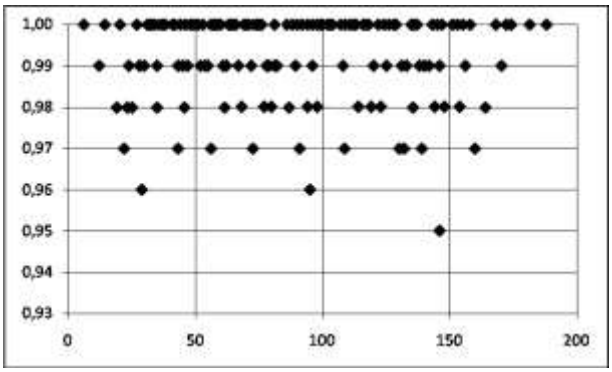


Рис. 16. Статистический портрет программной реализации UMAC-32

Анализ рис. 16 показывает, что алгоритм UMAC-32 имеет хорошие статистические свойства, но один тест был пройден с вероятностью ниже допустимой, что свидетельствует о возможной реализации одной из атак на данный алгоритм формирования MAC-кода.

Результаты тестирования ключевых хэш-функций сведены в табл. 5.

Таблица 5

Результаты тестирования

Генератор	Количество тестов, где тестирование прошло> 99% последовательностей	Количество тестов, где тестирование прошло> 96% последовательностей
MASH-1	101 (53%)	188 (99%)
MASH-2	126 (67%)	187 (98%)
MASH(EC)	141 (74%)	189 (100%)
HMAC-SHA-256	134 (71%)	187 (98%)
EMAC	138 (73%)	189 (100%)
RIPEMD-160	129 (68%)	189 (100%)
UMAC-32	173 (91%)	188 (99%)

Анализ табл. 5 показывает, что коды аутентичности и коды целостности сообщений в целом имеют хорошие статистические свойства безопасности. Но лучшими свойствами обладают алгоритмы UMAC-16 (UMAC-32), которые кроме этого имеют высокие показатели скорости

хеширования более чем 109 бит / с. Однако, применяемые методы универсального хеширования не позволяют обеспечить криптографическую стойкость к атакам злоумышленника [2, 3].

Выводы. Предложенный NIST (Американским Национальным Институтом Стандартов) пакет тестов NIST STS для тестирования генераторов случайных или псевдослучайных чисел является одним из подходов реализации задачи оценки статистической безопасности криптографических примитивов. Использование данного пакета позволяет с высокой долей вероятности судить о том, насколько последовательность, генерируемая исследуемым примитивом, является статистически безопасной. Проведенные исследования статистической стойкости хеш-функций подтвердили устойчивость механизмов обеспечения целостности и подлинности сообщений в АБС. Однако усиление устойчивости существующих алгоритмов хеширования строится на усложнении алгоритма формирования хеш-кода, в стремительных условиях развития вычислительной техники налагает определенные ограничения на их использование.

Перспективным направлением дальнейших исследований является разработка новых подходов к формированию устойчивых универсальных схем хеширования,

Список литературы

1. Король О. Г. Механізми безпеки внутрішньоплатіжних та Інтернет-платіжних систем комерційного банку України / О. Г. Король, С. П. Євсеєв // Сучасний захист інформації : науково-технічний журнал. Спец. випуск. – 2012. – С. 40–50.

2. Король О. Г. Обоснование выбора цикловой функции для итеративного хеширования информации / О. Г. Король, Л. Т. Пархуць, С. П. Евсеев // Системи обробки інформації. – 2013. – № 6(113). – С. 115– 119.

3. Король О. Г. Обоснование предложений по совершенствованию методов каскадного ключевого хеширования / О. Г. Король, Л. Т. Пархуць // Захист інформації і безпека інформаційних систем : матеріали II міжнародної науково-технічної конференції, м. Львів, 30 травня – 1 червня 2013 р. – 2013. – С. 142–143.

4. Король О. Г. Построение моделей атак на внутриплатежные банковские системы / О. Г. Король, С. П. Евсеев, А. И. Гончарова // Радіоелектроніка, інформатика, управління. – 2010. – № 1(22). – С. 56–66.

5. Король О. Г. Протоколы безопасности телекоммуникационных сетей / О. Г. Король // Системи обробки інформації. – 2012. – № 6(104). – С. 113– 120.

6. A.Rukhin, J.Soto. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22, 09.2000.

7. S. Pincus and R. E. Kalman. Not all (possibly) random sequences are created equal, Proc. Natl. Acad. Sci. USA. Vol. 94, April 1997, pp. 3513-3518.

8. 18. A. Rukhin (2000). Approximate entropy for testing randomness, Journal of Applied Probability. Vol. 37, 2000.

9. 19. Pal Revesz. Random Walk in Random And Non-Random Environments. Singapore: World Scientic, 1990.

10. 20. Frank Spitzer. Principles of Random Walk. Princeton: Van Nostrand, 1964, (especially p. 269).

Рецензент: ***ХОРОШКО Володимир Олексійович***

Національний Авіаційний Університет, Київ, доктор технічних наук, професор

Авторы: ***КОРОЛЬ Ольга Григорьевна***

Харьковский национальный экономический университет имени Семена Кузнеца, Харьков, кандидат технических наук, доцент кафедры информационных систем.

Раб. тел. – 702-18-31, E-mail – olha.korol2016@gmail.com

ФЕДЬКО Виктор Васильевич

Харьковский национальный экономический университет имени Семена Кузнеца, Харьков, кандидат физико-математических наук, профессор кафедры информационных систем.

Раб. тел. – 702-18-31, E-mail – vfedko@mail.ru

ОГУРЦОВ Виталий Вячеславович

Харьковский национальный экономический университет имени Семена Кузнеца, Харьков, кандидат экономических наук, доцент кафедры информационных систем.

Раб. тел. – 702-18-31, E-mail – vetalreal@ukr.net

ОЦІНКА СТАТИСТИЧНИХ ВЛАСТИВОСТЕЙ ГЕШ-ФУНКЦІЙ ЗА ДОПОМОГОЮ ПАКЕТУ NIST STS 822

О.Г. Король, В.В. Федько, В.В. Огурцов

Розглядається програмний пакет NIST STS 822, який використовується для тестування генераторів випадкових і псевдовипадкових чисел, що дозволяє оцінити статистичну безпеку криптографічних примітивів. За допомогою даного пакету досліджується статистична безпека ключових хеш-функцій, що застосовуються для формування MAC-кодів .

Ключові слова: статистична безпека, криптографічні примітиви, MAC-коди.

ASSESSMENT THE STATISTICAL PROPERTIES OF THE HASH FUNCTIONS USING A PACKAGE OF NIST STS 822

O. Korol, V. Fedko, V. Ohurtsov

We consider the software package NIST STS 822 that is used for testing random generators and pseudo-random numbers, allowing to estimate the statistical security of cryptographic primitives. With this package we investigated the statistical security of key hash functions used to generate the MAC codes.

Keywords: statistical security, cryptographic primitives, MAC codes.