4. Yano K. On semi-symmetric metric connection // Revue Roumame de Math. Pure et Appliquees. – 1970. – № 15. – P. 1579–1586.

5. Agricola I., Kraus M. Manifolds with vectorial torsion // Differential Geometry and its Applications. – 2016. – V. 46.

6. Cerbo L.F. Generic properties of homogeneous Ricci solitons // Adv. Geom. – 2014 – V.14(2). – P. 225–237.

7. Клепиков П.Н. Оскорбин Д.Н. Однородные инвариантные солитоны Риччи на четырехмерных группах Ли // Известия АлтГУ. – 2015. – № 1/2(85)

УДК 681.3

Практические разработки на базе клеточных автоматов

А.И. Латышева, А.Н.Гамова

Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского г. Саратов

Потребность в безопасном хранении паролей и в безопасной передаче сообщений только растёт. Оба направления криптографии можно основывать на теории клеточных автоматов. Если для вычисления хэш-кода пароля достаточно применять одномерные клеточные автоматы с их классическими 256 правилами развития, то для шифрования – двумерные. При анализе результатов работы программ было отмечено, что подбор правила развития клеточного автомата и количество раундов подсчёта хэш-кода рекомендуется основывать не на логине, а на соли. Так уменьшается вероятность подбора нужных данных для вычисления хэш-кода третьими лицами.

Ключевые слова: *клеточный автомат, хэш-код, логин, пароль, одномерный клеточный автомат, соль, правило развития клеточного автомата.*

Одномерный клеточный автомат представляет собой массив, состоящий из клеток, следующее состояние которых определяется её нынешним состоянием и нынешним состоянием её соседей или, другими словами, окрестностью клетки. Как правило, рассматривают одномерные клеточные автоматы с двумя состояниями. Всего существует 8 всевозможных комбинаций состояний клетки и её

соседей. Правилами развития клеточных автоматов называются схемы, по которым происходит изменение состояний автомата в следующий момент времени.

О применении теории клеточных автоматов в криптографии говорят уже давно. Данные структуры очень просты в реализации в программном виде. При этом развитие клеточного автомата происходит относительно быстро, так как, по сути, это линейная замена одних символов на другие в соответствии с правилом развития клеточного автомата. Теорию клеточных автоматов можно применять и для вычисления хэш-кодов паролей и для шифрования сообщений. Одномерный клеточный автомат представляет собой массив, состоящий из клеток, следующее состояние которых определяется её нынешним состоянием и нынешним состоянием её соседей или, другими словами, окрестностью клетки. Всего существует 256 возможных правил для одномерных клеточных автоматов. Они были описаны Стивеном Вольфрамом [1] с помощью кодов Вольфрама, где названия правил совпадают с нумерацией кодов, то есть от 0 до 255. Основываясь на определении, код Вольфрама может быть вычислен следующим образом: определяем все возможные варианты окрестностей клетки. Отсортировываем варианты окрестностей по убыванию (интерпретируя варианты окрестностей как число). Для каждого варианта окрестности определяем состояние, которое будет у клетки в следующий момент времени, в соответствии с правилом. Преобразовываем полученный список состояний в десятичное число. Это и будет кодом Вольфрама.

Хэш-функция – это функция, отображающая строки бит в строки бит фиксированной длины и удовлетворяющая следующим свойствам:

1) По данному значению функции сложно вычислить исходные данные, отображаемые в это значение;

2) Для заданных исходных данных сложно вычислить другие исходные данные, отображаемые в то же значение функции;

3) Сложно вычислить какую-либо пару исходных данных, отображаемых в одно и то же значение.

Хэш-код – это строка бит, являющаяся выходным результатом хэш-функции. Хэш-функции используются в следующих случаях: проверка целостности данных; система аутентификации; создание и проверка электронной цифровой подписи.

В данной работе хэш-функции будут применяться для получения хэш-кодов паролей. Применение хэш-функции для вычисления хэш-кодов паролей является односторонним процессом, то есть, нет необходимости в обратимости функции перехода. Следовательно, для

получения хэш-кода пароля можно использовать классические 256 правил развития одномерного клеточного автомата, которые являются необратимыми [2]. Далее рассмотрим влияние логина и пароля на формирование хэш-кода.

Вход: логин и пароль (которые передаются на вход программы через консоль);

Выход: логин, соль и хэш-код в символьном представлении (записываются в выходной файл txt).

На рисунке 1 продемонстрирована консоль программы, где нужно произвести выбор действия (вычислить хэш-код или сравнить хэш-код введённого пароля с имеющимся в файле), а также ввести логин, пароль, и правило развития клеточного автомата.

```
C:\Users\Latys\PycharmProjects\hashcellautomata\venv\Scripts\python.exe
Выбирите, что нужно выполнить:
1 - вычислить хэш-код;
2 - сравнить логин и пароль с имеющимся хэш-кодом;
Выбор: 1
Введите свой логин и пароль.
Логин:  login
Пароль: password
Правило №:  5
Правило развития клеточного автомата:
['111', '110', '101', '100', '011', '010', '001', '000']
['0', '0', '0', '0', '0', '1', '0', '1']
```

Рисунок 1 – Вывод на консоль

На рисунке 2 показан результат работы для введённых данных. В выходном файле первое слово – это логин, второе слово – это соль, третье слово – это хэш-код.

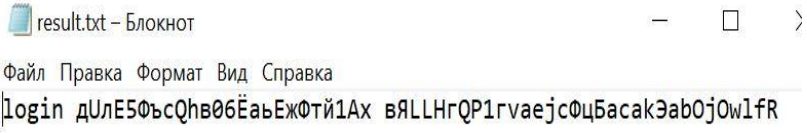

Рисунок 2

Изменим логин для того же входного файла с рисунка 2. Как видим на рисунке 3, программа вывела сообщение о несовпадении логинов.

```
C:\Users\Latys\PycharmProjects\hashcellautomata\venv\Scripts\python.exe
Выбирите, что нужно выполнить:
1 - вычислить хэш-код;
2 - сравнить логин и пароль с имеющимся хэш-кодом;
Выбор: 2
Введите свой логин и пароль.
Логин:  logi
Пароль: password
Ошибка! Логины не совпадают.
```

Рисунок 3

```
C:\Users\Latys\PycharmProjects\hashcellautomata\venv\Scripts\python.exe
Выбирите, что нужно выполнить:
1 - вычислить хэш-код;
2 - сравнить логин и пароль с имеющимся хэш-кодом;
Выбор: 2
Введите свой логин и пароль.
Логин:  login
Пароль: password
Правило №:  5
Хэш-коды совпадают.
```

Рисунок 4 -Результат проверки хэш-кодов

```
C:\Users\Latys\PycharmProjects\hashcellautomata\venv\Scripts\python.exe
Выбирите, что нужно выполнить:
1 - вычислить хэш-код;
2 - сравнить логин и пароль с имеющимся хэш-кодом;
Выбор: 2
Введите свой логин и пароль.
Логин:  login
Пароль: passwor
Правило №:  5
Ошибка! Хэш-коды не совпадают.
```

Рисунок 5 – Результат проверки хэш-кодов при неправильном пароле

Библиографический список

1. Wolfram, S.A new kind of science [Электронный ресурс] // wolframscience.com [Электронный ресурс]: Online-библиотека. – URL: https://www.wolframscience.com/nks/ (дата обращения: 06.10.2020).

2. Озорзин, А. Простейшие клеточные автоматы и их практическое применение [Электронный ресурс] // Хабр [Электронный ресурс]: Сообщество IT-специалистов. URL: https://habr.com/ru/post /273393/#:~:text=А%20что%20же%20тогда%20такое,клеток%20в%20 предыдущий%20момент%20времени (дата обращения: 03.09.2020).

УДК 514.765

Об одном уравнении Эйнштейна на группах Ли с полусимметрической связностью

А.А. Павлова[1], Е.Д. Родионов[1], О.П. Хромова[1]
[1]АлтГУ, г. Барнаул

В настоящейработе исследуется уравнение Эйнштейна вида $Symr = \Lambda g$, где $Symr$ – симметрическая часть тензора Риччи, g – метрический тензор, Λ – некоторая константа на трехмерных группах Ли с левоинвариантной римановой метрикой и полусимметрической связностью.

Ключевые слова: *уравнение Эйнштейна, трехмерные группы Ли, полусимметрическая связность.*

Пусть *(M; g)* – (псевдо) риманово многообразие размерности *n*. Определим полусимметрическую связность формулой:

$$\nabla_X Y = \nabla^g_X Y + g(X,Y)V - g(V,Y)X, \qquad (1)$$

где ∇^g – связность Леви-Чивиты.

Замечание. Полусимметрическая связность впервые была открыта Э. Картаном и изучалась в работах многих математиков [1-8]. Данную связность также называют связностью с векторным кручением.

Определим тензор кривизныR и тензор Риччи r риманова многообразия *(M, g)*, используя полусимметрическую связность, формулами:

$$R(X,Y)Z = \nabla_Y \nabla_X Z - \nabla_X \nabla_Y Z + \nabla_{[X,Y]} Z,$$

где $[\cdot;\cdot]$ – скобка Ли векторных полей;

$$r = tr(U \to R(X,U)Y).$$

Тензор Риччи полусимметрической связности, вообще говоря, не является симметрическим, поэтому вместо тензора Риччи в уравнении Эйнштейна $r = \Lambda g$, рассмотрим его симметрическую часть, то есть будем решать задачу

$$r_{ij} + r_{ji} = \Lambda g_{ij}. \qquad (2)$$