УДК 004.056.53 https://doi.org/10.33619/2414-2948/80/35

АНАЛИЗ МЕТОДОВ ГЕНЕРАЦИИ ОДНОРАЗОВЫХ ПАРОЛЕЙ И ВЫСОКАЯ СТЕПЕНЬ СЛУЧАЙНОСТИ ГЕНЕРИРУЕМЫХ ПАРОЛЕЙ

©Арзиева Ж. Т., *Каракалпакский государственный университет им. Бердаха, г. Нукус, Узбекистан, a_jamila@karsu.uz*
©Арзиев А. Т., *Ташкентский университет информационных технологий им. Мухаммада аль-Хоразмий, г. Нукус, Узбекистан*

ANALYSIS OF METHODS FOR GENERATING ONE-TIME PASSWORDS AND A HIGH DEGREE OF RANDOMNESS OF GENERATED PASSWORDS

©Arzieva J., *Karakalpak State University named after Berdakh, Nukus, Uzbekistan, a_jamila@karsu.uz*
©Arziev A., *Tashkent University of Information Technologies named after Muhammad Al-Khwarizmi, Nukus, Uzbekistan*

Аннотация. Рассматриваются методы аутентификации. Генерация «чего-либо» (например, паролей) является значительной частью данного процесса. При проверке стойкости генерированных паролей к каким-либо атакам (например, атака с польным перебором) выполняется измерение энтропии пароля. Существующие методы генерации случайных чисел в широко распространенных языках программирования обеспечивают достаточными значениями для одноразовых паролей. Но при использовании данных генераторов требуется внутреннее обновление их на основе случайных значений.

Abstract. Authentication methods are considered. Generating “something” (e. g., passwords) is a significant part of this process. When checking the resistance of generated passwords to any attacks (for example, a brute-force attack), the entropy of the password is measured. Existing methods for generating random numbers in widely used programming languages provide sufficient values for one-time passwords. But when using these generators, they need to be internally updated based on random values.

Ключевые слова: генерация, аутентификация, генерации пароля, генерации одноразовых паролей, генераторы псевдослучайных чисел.

Keywords: generation, authentication, password generation, one-time passwords generation, pseudo-random number generators.

Высокая степень случайности генерируемых паролей обеспечивает требуемую безопасность. При этом в общем случае генераторы паролей состоят из следующих составляющих [5]:

Набор входящих значений. Данные значения являются необходимыми для генерации паролей, и они могут быть различными с учетом требуемого пароля.

Функция генерации пароля. Данная функция генерирует пароль в соответствии ассоциирования входящих значений.

Т. 8. №7. 2022
https://doi.org/10.33619/2414-2948/80

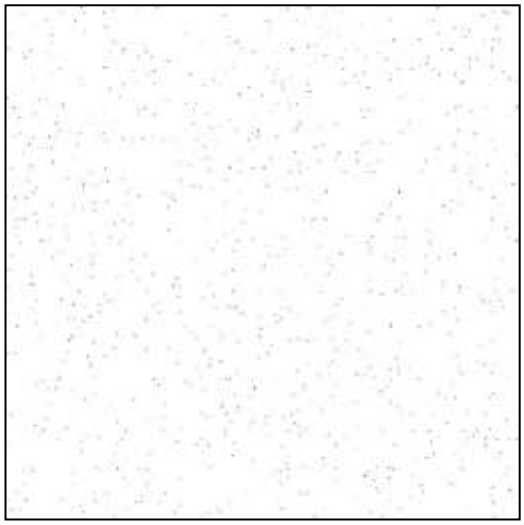
Метод распечатки паролей. Данная функция выполняет действия распечатки для пользователя или трансфера в сайт аутентификации генерированного пароля для введения в требуемом поле какой-либо системы.

Генерация пароля осуществляется по-разному исходя из возможных сред его использования. В настоящее время при генерации одноразовых паролей можно применить разные методы, которых можно разделить в общем случае на следующие группы:

1. Генераторы, основанные на псевдослучайных числах использующих в качестве параметра временную метку. В практике широко применяющиеся методы генерации одноразовых паролей (*OTP — One-Time Password*) основаны на синхронизации времени, и среди них особое значение имеет TOPT *(Time-based One-Time Password Algorithm)* алгоритм. В настоящее время широко применяющиеся приложения третьей стороны основаны на TOPT алгоритмов и в качестве примера можно привести Google Authenticator, Microsoft Authenticator и др. В частности, в сегодняшний день во многих системах (например, Gmail.com, facebook.com, GitHub, Twitter, Dropbox) Google Authenticator применяется для реализации аутентификации, основанную на двух факторах. Существование возможностей применения данных приложений в разных системах обосновывается использование временных меток в качестве распределенных параметров в системах TOTP. В приложении Google Authenticator генерируется OTP, состоящий из 6 символов. С целью проверки степени случайности OTP генерированных при помощи данного приложения, оно соединено для систем Gmail, Facebook, Dropbox и Github (Рисунок 1 a). Здесь значения являются разными из-за предоставления различных ключей со стороны системы и из-за различности временных меток. Было собрана генерированных 1000 OTP для данных приложений. Каждый OTP имел цифру из шести значений, и его графический вид приведен на Рисунке 1 б [4].

2.

a) OTP генерированные в приложении Google Authenticator



b) Графическое изображение OTP

Рисунок 1. OTP и его графическое изображение

Для отображения графического вида на основе OTP, каждый пароль разделяется на две части. Например, если OTP равна к OTP =458234, то первая часть OTP будет равна к X=458, а вторая часть OTP будет равна к Y=234. На основе двух чисел точка *P(X, Y)* отображается в двумерной системе (999, 999) координат. Как и отображена в графике возможные варианты паролей генерированных с помощью Google Authenticator будет равна к 10^6. В приложении

Т. 8. №7. 2022
https://doi.org/10.33619/2414-2948/80

Google Authenticator для генерации OTP применяется временная метка и с целью устранения определенных отклонений, OTP за каждые 30 сек. или 60 сек. поменяется.

OTP, которые состоят только из цифр являются легко вводимыми в систему, но при этом они не считаются обладателями высокой степени стойкости. Поэтому, обычно при генерации OTP (если они состоят только из цифр) уделяется внимание на длину паролей. Например, в социальной сети Facebook заранее генерированные OTP (TAN), которые передаются пользователям, состоят из 8 цифр (Рисунок 2).

2. Генераторы псевдослучайных чисел, основанные на использовании счетчиков в качестве параметра. Одним из примеров, который относиться к данным методам генерации OTP является алгоритм HOTP. Функционирование данного алгоритма идентично со схемой, приведенной на Рисунке 1. Отличие — использование вместо метки времени счетчика [4]. На Рисунке 3 приведены результаты, полученные от отображения в графическом виде начальных 1000 OTP генерированных на основе HOTP из единого ключа.

Кроме этого, алгоритмы генерации OTP входящие в данную группу широко применяются в протоколах аутентификации типа «вопрос-ответ» (Challenge-Response), основанные на одноразовые пароли. При этом вместо счетчика вводится значения «вопроса» и выполняется процесс аутентификации через сопоставление его ответа.

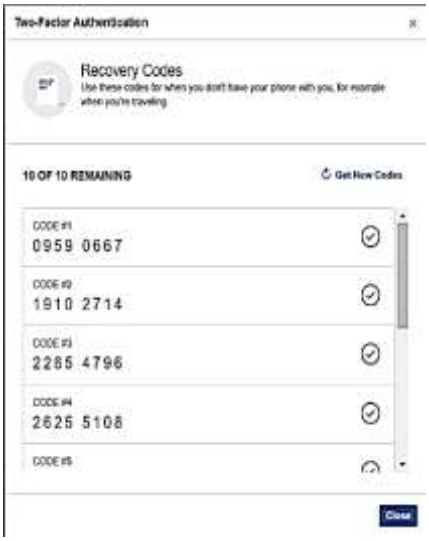


Рисунок 2. Список TAN в системе Facebook.

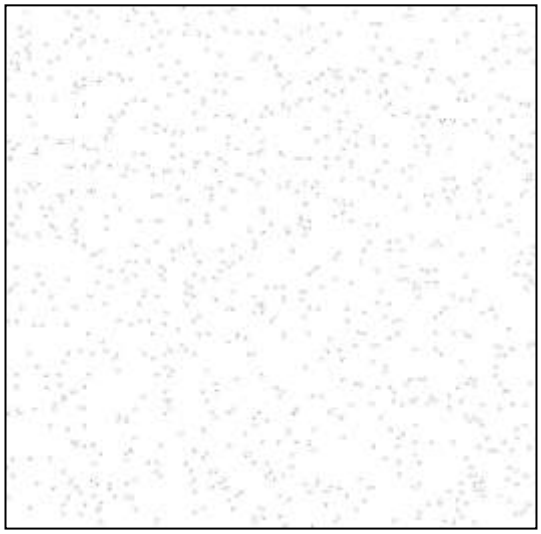


Рисунок 3. Графический вид результата алгоритма HOTP

3. Генерация паролей, основанные на использования накопления определенных символов. OTP, которые состоят только из цифр и являются не стойкими к атакам полного перебора и обычно считаются очень уязвимыми.

Кроме этого, в системах требующих высокую безопасность применяются OTP, которые состоят из маленьких латинских букв (26), больших латинских букв (26) и цифр (10) [3]. В данном случае если длина пароля равна к 6, тогда вариация возможных паролей будет равна к $62^6 \approx 5.68 \times 10^{10} \approx 2^{35.7}$ и значение рассматривается как стойкий пароль. OTP такого вида в многих системах используются для формирования TAN списка, в частности Gmail, Dropbox. Также OTP таких видов можно использовать в качестве статических паролей.

В области генерации паролей также уделяется высокое внимание разработке генераторов паролей, которые используют набор определенных символов, легко в произношении и сохранении в памяти, но при этом имеющийся высокий степень

случайности. В качестве примера можно привести генератор PRONOUNCE3, разработанный со стороны Лионарда и других [5]. В данном генераторе можно генерировать OTP, которая равно к 30.8 бит энтропии, используя гласных (a, e, i, o, u) и негласных (b, c, ch, d, f, g, h, j, k, l, m, n, p, ph, r, s, st, v, w, x, y, z) букв.

Рисунок 4. Оценка OTP в системе http://www.passwordmeter.com/

В области генерации паролей также уделяется высокое внимание разработке генераторов паролей, которые используют набор определенных символов, легко в произношении и сохранении в памяти, но при этом имеющийся высокий степень случайности. В качестве примера можно привести генератор PRONOUNCE3, разработанный со стороны Лионарда и других [5]. В данном генераторе можно генерировать OTP, которая равно к 30,8 бит энтропии, используя гласных (a, e, i, o, u) и негласных (b, c, ch, d, f, g, h, j, k, l, m, n, p, ph, r, s, st, v, w, x, y, z) букв.

В случайном выборе из набора определенных символов используются функции существующих в различных языках программирования (*rand(), random()*), и складываются символы на основе определенного алгоритма.

```
1.  int main(void)
2.   {
3.   /* Длина пароля */
4.   unsigned short int length = 8;
5.
6.     /* rand()обновить внутреннее состояние функции */
7.   srand((unsigned int) time(0));
8.
9.   /* символы ASCII от 33 до 126 */
10.  while(length--) {
11.  putchar(rand() % 94 + 33);
12.  }
13.  printf("\n");
14.  return EXIT_SUCCESS;
15. }
```

4. Генерация паролей, основанный на генераторе случайных чисел. Генератор данного типа обычно мало распространены и применяются при случаях, где существуют генераторы случайных или псевдослучайных чисел. Например, ниже приведен алгоритм генерации паролей 8 длины, с использованием генераторов псевдослучайных чисел $rand()$ на языке программирования C:

Также во многих операционных системах существуют стойкие генераторы случайных чисел (например, для семейства Unix */dev/random* и */dev/urandom* или для Windows *CryptGenRandom*) и с помощью их можно генерировать паролей с высокой степенью случайности.

Анализ стойкости паролей. Является важным анализ стойкости паролей, созданнные с помощью генераторов паролей, и их можно анализировать с помощью различных способов, с учетом вида применения их, то есть использование в качестве статического или одноразового пароля.

При проверке генерированных паролей широко применяется тестирование по свойствам таких как, состав, длина и не существование в списке широко распространенных паролей [3]. Пароли, отвечающие к этим требованиям, считаются стойкими к атакам «Грубая сила» и атака на основе словаря.

Кроме того, при проверке стойкости генерированных паролей к каким-либо атакам (например, атака с польным перебором) выполняется измерение энтропии пароля. Если символы пароля не основаны каким-либо законам и являются независимыми, то энтропия пароля определяется следующим уравнением:

$$H = L \log_2 N = L \frac{\log N}{\log 2}$$

Здесь N — количества возможных символов, $L -$ определяет количества символов в пароле. H измеряется в битах. В Таблице 1 приведены значения энтропии на каждые символы для набора различных номеров символов.

Таблица 1

ЭНТРОПИЯ СООТВЕТСТВУЮЩАЯ НА КАЖДЫЙ СИМВОЛ
ДЛЯ НАБОРА РАЗЛИЧНЫХ СИМВОЛОВ [5]

Набор символов	*Количества символов в наборе, N*	*Энтропия соответствующая к одному символу, H (бит)*
$0-9$	10	3,32
$0-9, A-F$	16	4,00
$a-z$ или $A-Z$	26	4,70
$a-z$ или $A-Z, 0-9$	36	5,17
$a-z, A-Z$	52	5,70
$a-z, A-Z, 0-9$	62	5,95
Все прописные символы ASCII	94	6,55

В Таблице 2 приведены затраченные время для определения с помощью польного перебора OTP, которые имеют различные высокие сложности с использованием вышеприведенных наборов символ. Для получения результатов использована онлайн система проверки паролей http://password-checker.online-domain-tools.com/

Результаты анализа показывают, что в операционных системах семейств Windows и UNIX существующие генераторы случайных чисел имеют достаточную степень безопасность для создания OTP.

Таблица 2

ВРЕМЯ, ЗАТРАЧЕННОЕ ДЛЯ ОПРЕДЕЛЕНИЯ РАЗЛИЧНЫХ ПАРОЛЕЙ МЕТОДОМ ПОЛЬНОГО ПЕРЕБОРА

	Состоящий из (0...9) с длиной 10 символов (5689743664)	*Состоящий из (0...9, a…z, A…Z) с длиной 10 символов (j63o1f9Avu)*	*Состоящий из (0...9, a…z, A…Z, ?,/,~,!,(,),| ...) с длиной 10 символов (c?Kxar/XM7)*
Стандарт ПК	2 мин	3 000 год	208 000 год
Быстрый ПК	25 сек	67 год	52 000 год
GPU	10 сек	27 год	21 000 год
Быстрый GPU	5 сек	13 год	10 000 год
Параллельный GPU	1 сек	1 год	87 год
Боты среднего числа	< 1 сек	2 час	6 день
Стойкость (от 100)	26%	55%	62%

Существующие методы генерации случайных чисел в широко распространенных языках программирования обеспечивают достаточными значениями для OTP. Но при использовании данных генераторов требуется внутреннее обновление их на основе случайных значений (например, показатели миллисекунды текущей времени).

Список литературы:

1. Shimizu A., Horioka T., Inagaki H. A password authentication method for contents communications on the Internet // IEICE transactions on communications. 1998. V. 81. №8. P. 1666-1673.

2. Tsuji T., Kamioka T., Shimizu A. Simple and secure password authentication protocol, ver. 2 (SAS-2) // ITE Technical Report 26.61. The Institute of Image Information and Television Engineers, 2002. P. 7-11. https://doi.org/10.11485/itetr.26.61.0_7

3. Dorrendorf L., Gutterman Z., Pinkas B. Cryptanalysis of the random number generator of the windows operating system // ACM Transactions on Information and System Security (TISSEC). 2009. V. 13. №1. P. 1-32. https://doi.org/10.1145/1609956.1609966

4. Karimov M. M., Khudoykulov Z. T., Arzieva Ja. T. A Method of Efficient OTP Generation Using Pseudorandom Number Generators // 2019 International Conference on Information Science and Communications Technologies (ICISCT). IEEE, 2019. P. 1-4. https://doi.org/10.1109/ICISCT47635.2019.9011825

5. Арзиева Ж. Использование одноразовых паролей для одного сеанса аутентификации // Zamonaviy innovatsion tadqiqotlarning dolzarb muammolari va rivojlanish tendensiyalari: yechimlar va istiqbollar. 2022. V. 1. №1. P. 149-150.

6. Akhmatovich T. K., Turakulovich K. Z., Tileubayevna A. J. Improvement of a security enhanced one-time mutual authentication and key agreement scheme // International Journal of Innovative Technology and Exploring Engineering. 2019. V. 8. №12. P. 5031-5036.

7. Арзиева Ж., Нукусбаев Н. Ж. Проблемы сетевой безопасности и эффективная защита от сетевых атак // Бюллетень науки и практики. 2021. Т. 7. №9. С. 479-485. https://doi.org/10.33619/2414-2948/70/45

References:

1. Shimizu, A., Horioka, T., & Inagaki, H. (1998). A password authentication method for contents communications on the Internet. *IEICE transactions on communications*, *81*(8), 1666-1673.

2. Tsuji, T., Kamioka, T., & Shimizu, A. (2002, September). Simple and secure password authentication protocol, ver. 2 (SAS-2). In *ITE Technical Report 26.61* (pp. 7-11). The Institute of Image Information and Television Engineers. https://doi.org/10.11485/itetr.26.61.0_7

3. Dorrendorf, L., Gutterman, Z., & Pinkas, B. (2009). Cryptanalysis of the random number generator of the windows operating system. *ACM Transactions on Information and System Security (TISSEC)*, *13*(1), 1-32. https://doi.org/10.1145/1609956.1609966

4. Karimov, M. M., Khudoykulov, Z. T., & Arzieva, J. T. (2019). A Method of Efficient OTP Generation Using Pseudorandom Number Generators. In *2019 International Conference on Information Science and Communications Technologies (ICISCT)* (pp. 1-4). IEEE. https://doi.org/10.1109/ICISCT47635.2019.9011825

5. Arzieva, Zh. (2022). Ispol'zovanie odnorazovykh parolei dlya odnogo seansa autentifikatsii. *Zamonaviy innovatsion tadqiqotlarning dolzarb muammolari va rivojlanish tendensiyalari: yechimlar va istiqbollar, 1*(1), 149-150.

6. Akhmatovich, T. K., Turakulovich, K. Z., & Tileubayevna, A. J. (2019). Improvement of a security enhanced one-time mutual authentication and key agreement scheme. *International Journal of Innovative Technology and Exploring Engineering*, *8*(12), 5031-5036.

7. Arzieva, J., & Nukusbaev, N. (2021). Network Security Issues and Effective Protection Against Network Attacks. *Bulletin of Science and Practice, 7*(9), 479-485. (in Russian). https://doi.org/10.33619/2414-2948/70/45

Работа поступила
в редакцию 25.05.2022 г.

Принята к публикации
30.05.2022 г.

Ссылка для цитирования:

Арзиева Ж. Т., Арзиев А. Т. Анализ методов генерации одноразовых паролей и высокая степень случайности генерируемых паролей // Бюллетень науки и практики. 2022. Т. 8. №7. С. 382-388. https://doi.org/10.33619/2414-2948/80/35

Cite as (APA):

Arzieva, J., & Arziev, A. (2022). Analysis of Methods for Generating One-Time Passwords and a High Degree of Randomness of Generated Passwords. *Bulletin of Science and Practice, 8*(7), 382-388. (in Russian). https://doi.org/10.33619/2414-2948/80/35