УДК 004.056.55

Ступень П. В.,
канд. техн. наук, доцент, Одеський національний політехнічний університет, м. Одеса, Україна
Золкіна О. Ю.,
Одеський національний політехнічний університет, м. Одеса, Україна
Соколов С. О.,
Одеський національний політехнічний університет, м. Одеса, Україна

АНАЛІЗ АЛГОРИТМІВ ШИФРУВАННЯ ДАНИХ ДЛЯ ЗАХИСТУ ІНФОРМАЦІЇ НА ХМАРНИХ СХОВИЩАХ

Хмарні сховища дозволяють зберігати свої файли на серверах та передавати їх іншим користувачам в Інтернеті. Робота побудована на синхронізації даних між різними пристроями. У наш час практично кожен користувач комп'ютера стикався під час роботи з хмарними сховищами. Незважаючи на те, що такі системи отримали широке розповсюдження, вони мають суттєві недоліки, перш за все, це проблеми інформаційної безпеки. Будь-яка людина, котра отримає доступ до облікового запису користувача, може з легкістю роздобути конфіденційні дані або знищити важливу інформацію, яка знаходиться в сховищі.

Найпопулярніші ресурси, такі як Google, Dropbox, Mail.ru, Яндекс, котрі надають своїм користувачам можливість безкоштовного користування хмарним сховищем – не мають жодних можливостей для шифрування даних, та не дають жодних гарантій щодо їх збереження. Звісно, можна використати спеціальні програми, та за допомогою них зашифрувати дані і вже в такому вигляді завантажити у хмару. Проте, такі методи дуже не зручні, адже доводиться використовувати стороннє програмне забезпечення. За його відсутності неможливо розшифрувати дані та переглянути їх на будь-якому пристрої, хоча саме для цього і створені хмарні сховища. Метою є вибір алгоритму для використання його у веб-додатку для захисту інформації на хмарних сховищах.

Ключові слова: *хмарні сховища; алгоритми шифрування даних; аналіз алгоритмів шифрування; швидкість шифрування; швидкість розшифрування.*

Для проведення аналізу було обрано три алгоритми шифрування: AES, TripleDES, Rabbit.

Advanced Encryption Standard (AES), також відомий під назвою Rijndael – симетричний алгоритм блочного шифрування, фіналіст конкурсу AES і прийнятий у якості американського стандарту шифрування урядом США. Вибір припав на AES із розрахуванням на широке використання і активний аналіз алгоритму, як це було із його попередником, DES. Станом на 2009 рік AES є одним із найпоширеніших алгоритмів симетричного шифрування [1, с. 84].

TripleDES (3DES) – симетричний блочний шифр, створений Уітфілдом Діффі, Мартіном Хеллманом і Уолтом Тачманном у 1978 році на основі алгоритму DES, з метою усунення головного недоліку останнього – малої довжини ключа (56 біт), котрий може бути зломаний методом повного перебору ключа [2].

Rabbit – високошвидкісний потоковий шифр вперше представлений у лютому 2003 року на 10-му симпозіумі FSE. У травні 2005, він був відправлений на конкурс eStream, метою якого було створення європейських стандартів для поточних систем шифрування. Шифр було розроблено з метою використання в програмному забезпеченні, як той, що володіє високою швидкістю шифрування. Шифр також виявився швидким і компактним при реалізації в апаратному забезпеченні.

Аналіз проводився з використанням бібліотеки CryptoJS, котра розроблена на JavaScript та знаходиться у вільному доступі. Проаналізовано роботу алгоритмів з різними типами файлів та різними розмірами (табл. 1). Під час аналізу було встановлено залежність швидкості шифрування файлів від довжини ключа.

Проаналізувавши отримані результати одразу можна зробити перший висновок – розмір файлу не залежить від алгоритму шифрування, кожен із трьох обраних алгоритмів продемонстрував ідентичні результати.

Алгоритм, котрий буде обрано для роботи з хмарними сховищами повинен мати найкращі показники по швидкості шифрування файлів. Проаналізувавши отримані дані, можливо зробити висновок, що алгоритм Rabbit здатен зашифрувати файл найшвидше (рис. 1), а TripleDES потребує набагато більше часу.

Таблиця 1

Аналіз швидкості зашифровки файлу

Алгоритм	Довжина ключа, символів	Тип файлу	Розмір до шифрування, кб	Розмір після шифрування, кб	Час шифрування, мс	Швидкість зашифровки, кб/с
AES	6	.xlsx	159	282	103	1544
		.jpg	505	897	401	1259
		.rar	904	1608	934	968
AES	12	.xlsx	159	282	92	1728
		.jpg	505	897	505	457
		.rar	904	1608	1250	723
TripleDES	6	.xlsx	159	282	1091	146
		.jpg	505	897	3051	166
		.rar	904	1608	5996	151
TripleDES	12	.xlsx	159	282	1001	159
		.jpg	505	897	3402	148
		.rar	904	1608	5911	153
Rabbit	6	.xlsx	159	282	111	1432
		.jpg	505	897	381	1325
		.rar	904	1608	966	936
Rabbit	12	.xlsx	159	282	156	1019
		.jpg	505	897	578	874
		.rar	904	1608	1034	874

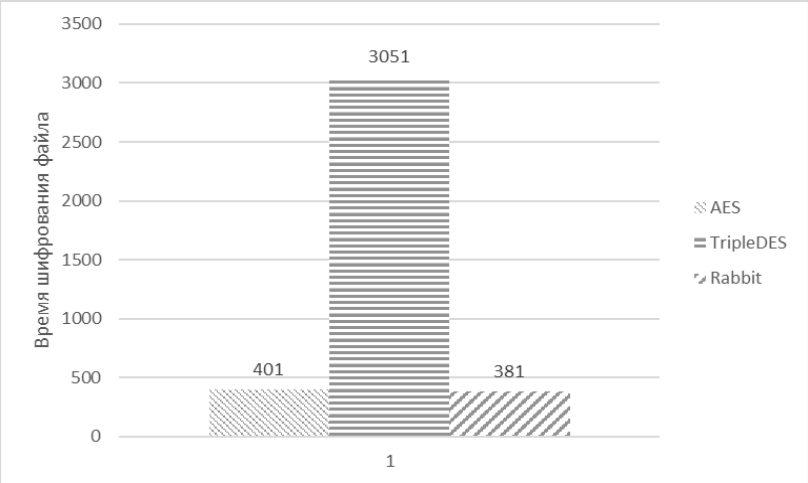

Рис. 1. Час, витрачений на зашифровку зображення

Для того, аби вибрати алгоритм з найбільшою швидкістю шифрування, було проведено аналіз залежності швидкості шифрування від довжини ключа (рис. 2). У порівнянні з іншими алгоритмами, TripleDES демонструє найгірші результати за швидкістю шифрування, не залежно від довжини ключа. Саме тому не рекомендується використання цього алгоритму для роботи з хмарними сховищами. AES показує гарні результати, проте швидкість шифрування стрімко падає при збільшенні розміру файлу. Найбільш стабільні результати при високій швидкості шифрування демонструє алгоритм Rabbit, саме тому він рекомендується для використання у хмарних сховищах.

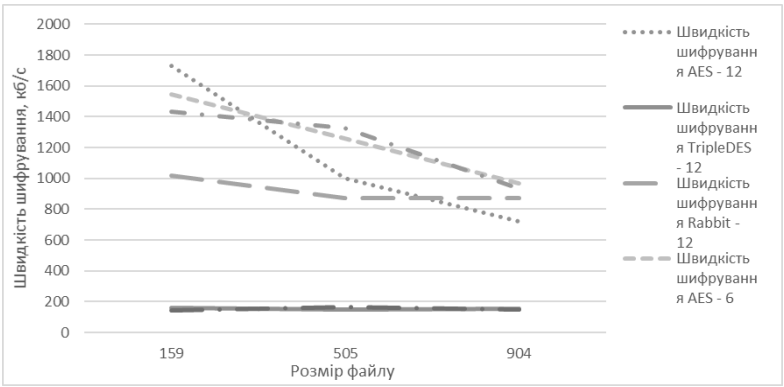

Рис. 2. Залежність швидкості шифрування від довжини ключа

На другому етапі аналізу ми розглянули роботу алгоритмів із різними типами файлів та різними розмірами під час розшифрування (табл. 2).

Таблиця 2

Аналіз швидкості розшифрування файлу

Алгоритм	Довжина ключа, символів	Тип файлу	Час розшифрування, мс	Швидкість розшифрування, кб/с
AES	6	.xlsx	241	660
		.jpg	658	767
		.rar	1126	803
AES	12	.xlsx	241	660
		.jpg	540	935
		.rar	885	1021
TripleDES	6	.xlsx	1110	143
		.jpg	3403	148
		.rar	6189	146
TripleDES	12	.xlsx	1017	156
		.jpg	3286	154
		.rar	5838	155
Rabbit	6	.xlsx	108	1472
		.jpg	685	737
		.rar	1133	798
Rabbit	12	.xlsx	128	1242
		.jpg	633	798
		.rar	1122	806

Для того, аби вибрати алгоритм з найбільшою швидкістю шифрування, було проведено аналіз залежності швидкості шифрування від довжини ключа (рис. 3).

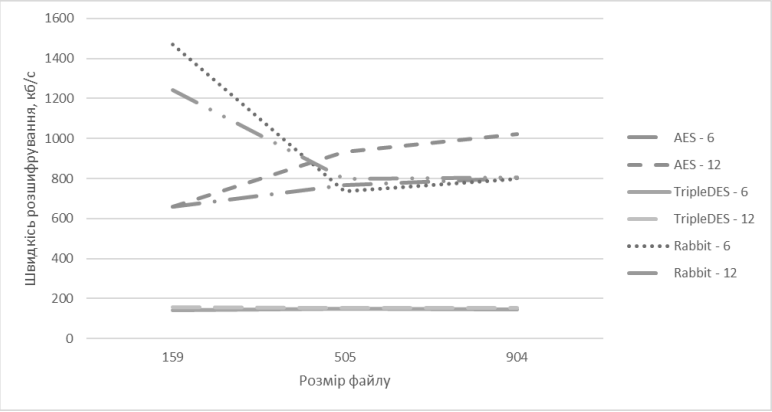

Рис. 3. Залежність швидкості розшифрування від довжини ключа

Наведений графік (рис. 3) демонструє, що алгоритм Rabbit найшвидше справляється з розшифруванням файлів невеликого розміру, та має стабільну швидкість розшифрування при збільшенні розміру файлу.

У результаті проведеного аналізу ми можемо зробити наступні висновки:

1. Розмір зашифрованого файлу не залежить від вибору алгоритму шифрування.
2. Для шифрування файлів найкращий результат по швидкості роботи продемонстрував алгоритм Rabbit.
3. Для алгоритму AES швидкість розшифрування файлу збільшується при збільшення довжини ключа.
4. Для використання у хмарних сервісах рекомендується використовувати алгоритм Rabbit, тому що він має найкращі показники по часу та швидкості шифрування файлів різних типів та розмірів.

ЛІТЕРАТУРА

1. Панасенко С. Алгоритмы шифрования. Специальный справочник / С. Панасенко. – СПб : БХВ-Петербург, 2009. – 576 с.
2. Баричев С. Г. Основы современной криптографии / С. Г. Баричев, В. В. Гончаров, Р. Е. Серов. – Горячая линия-Телеком, 2011. – 175 с.

Ступень П. В., Золкина О. Ю., Соколов С. А.,
Одесский национальный политехнический университет, г. Одесса, Украина

Анализ алгоритмов шифрования данных для защиты информации на облачных хранилищах

Облачные хранилища позволяют хранить свои файлы на серверах и передавать их другим пользователям в Интернете. Работа построена на синхронизации данных между различными устройствами. В наше время

практически каждый пользователь компьютера сталкивался при работе с облачными хранилищами. Несмотря на то, что такие системы получили широкое распространение, они имеют существенные недостатки, прежде всего это проблемы информационной безопасности. Любой человек, который получит доступ к учетной записи пользователя может с легкостью заполучить конфиденциальные данные, или уничтожить важную информацию, которая находится в хранилище.

Самые популярные ресурсы, такие как Google, Dropbox, Mail.ru, Яндекс, которые предоставляют своим пользователям возможность бесплатного пользования облачным хранилищем – не имеют никаких возможностей для шифрования данных, и не дают никаких гарантий их сохранности. Конечно, можно использовать специальные программы, и с помощью них зашифровать данные и уже в таком виде загрузить в облако. Однако такие методы очень неудобны, ведь приходится использовать стороннее программное обеспечение. При его отсутствии невозможно расшифровать данные и просмотреть их на любом устройстве, хотя именно для этого и созданы облачные хранилища. Целью является выбор алгоритма для использования его в веб-приложении для защиты информации на облачных хранилищах.

Ключевые слова: *облачные хранилища; алгоритмы шифрования данных; анализ алгоритмов шифрования; скорость шифрования; скорость расшифровки.*

Stupen P. V., Zolkina O. Y., Sokolov S. O.,
Odessa National Polytechnic University, Odessa, Ukraine

Analysis data encryption algorithms to protect data in cloud storages

Cloud storage can store their files on the servers and transmit them to other users through the Internet. The work is based on the synchronization of data between various devices. Nowadays, almost every computer user met with cloud storages during the work. Despite the fact that such systems are widespread, they have significant disadvantages, above all the problem of information security. Anyone who will get an access to users account can also easily get users private data or destroy important information stored in the cloud storage.

The purpose is selection of the algorithm for using in web applications to protect information on cloud storages. For the analysis have been put forward three encryption algorithms: AES, TripleDES, Rabbit. The analysis has been conducted using the library CryptoJS, which is developed in JavaScript and is freely available. Has been analysed the work of the algorithms with different file types and different sizes. During the analysis have been found the dependence between the speed of encryption files and the length of encryption key. As a result of analysis, we can draw the following conclusions: the size of the encrypted file does not depends on the encryption algorithm; the Rabbit algorithm has demonstrated the fastest speed of encryption files; in AES algorithm the decryption speed increases with increasing of encryption key length; for using in the cloud storages is recommended the Rabbit algorithm, because it has the best speed of encryption and decryption files of different types and sizes.

Key words: *cloud storage; data encryption algorithms; analysis of algorithms for encryption; speed encryption; decoding speed.*

© Ступень П. В., Золкіна О. Ю., Соколов С. О., 2015 *Дата надходження статті до редколегії 05.11.2015*