

# АНАЛІЗ ПРАКТИЧНОЇ СТІЙКОСТІ ПРОТОКОЛІВ ДВОФАКТОРНОЇ АВТЕНТИФІКАЦІЇ

М. В. Блик<sup>1, а</sup>

<sup>1</sup> Національний технічний університет України «Київський політехнічний інститут»,  
Фізико-технічний інститут

## Анотація

У даній роботі розглядається схема двофакторної автентифікації та проводиться практичний аналіз одноразових паролів, отриманих від Google.

**Ключові слова:** двофакторна автентифікація, одноразові паролі, безпека мобільних пристроїв

## Вступ

Широко розповсюджені в інтернеті онлайн – сервіси мають потребу в кращій автентифікації для своїх користувачів. В наш час користувачі вимушені керувати десятками різних паролів. Не дивлячись на те, що можливості людського мозку в запам'ятовуванні складних паролів досі є цілком дослідженими [1], факт того, що користувачі з складнощами справляються з кількістю паролів, які вони мають завжди пам'ятати. Тому онлайн сервіси почали вводити додаткові серверні механізми для захисту від крадіжок паролів. Один з них, можливо найбільш відомий, – двофакторна автентифікація.

Другий фактор може бути додатком для мобільного телефону, текстовим повідомленням отриманим від провайдера сервісу, власним вбудованим обладнанням. Всі вони будуть містити деяку одноразову інформацію. Ця інформація має бути надана сервісу разом з постійним паролем для доступу до нього. Наявність двофакторної автентифікації робить набагато складнішим процес отримання доступу до акаунтів користувачів для хакерів та авторів шкідливого програмного забезпечення.

В той же час розповсюдження мобільних платформ значно виросло протягом декількох останніх років [2]. Мобільні платформи створили нову екосистему для користувачів [2], тому ми розглянемо двофакторну автентифікацію базовану на використанні мобільних пристроїв.

## 1. Схема двофакторної автентифікації

Двофакторна автентифікація – це метод ідентифікації користувача в довільному сервісі (як правило, в Інтернеті) за допомогою запиту автентифікаційних даних двох різних типів, що забезпечує більш ефективний захист акаунта від несанкціонованого проникнення.

Взагалі, автентифікаційний фактор це щось, що доводить ідентичність користувача. Існує декілька варіантів для цього процесу:

- щось, що користувач знає;
- щось, що користувач має у володінні;
- щось, що є користувачем

*Щось, що користувач знає*, зазвичай є паролем. *Щось, що користувач має у володінні*, може бути апаратним токеном, що певним чином доводить, що користувач володіє ним, наприклад генеруванням паролю на основі попередньо спільного секрету. *Щось, що є користувачем*, це фізичні риси користувача, що можуть бути відсканованими спеціальним обладнанням, наприклад біометрією.

Четвертий фактор автентифікації був запропонований в [3]. Четвертий фактор є *кимось, кого користувач знає*. Ідея цього фактору заключається в тому, що користувачі можуть ручатись один за одного, і таким чином, засвідчувати ідентичність один одного.

Комбінуючи два чи навіть більше факторів, досягається вищий рівень безпеки автентифікації користувачів. Введення другого фактору, такого як одноразові токени згенеровані додатковим обладнанням, значно зменшує ризики втрати акаунту користувача, бо вимагає від зловмисника також наявності можливості фізичного доступу до пристрою. Важлива примітка: дійсно *щось, що користувач має у володінні* не є схильним до перехоплення та вимагає від користувача фізичного володіння токеном, наприклад володіння смарт-карткою. Тому двофакторна автентифікація побудована на відправці одноразових паролів за допомогою SMS до мобільних пристроїв користувачів не повністю задовільняє вимоги фактору *щось, що користувач має у володінні*, оскільки дана інформація може бути вразливою до прослуховування.

Ми будемо розглядати схему двофакторної автентифікації, що захищає автентифікаційний процес великих інтернет-сервісів (Google) від фішингових атак та атак базованих на використанні кілогерів.

<sup>а</sup>mykola.blyk@gmail.com

## 2. Схема двофакторної автентифікації Google

У випадку Google одноразові паролі генеруються на стороні серверу та в подальшому відправляються клієнту окремим каналом, в нашому випадку за допомогою мереж стільникового зв'язку у вигляді SMS. Генерація одноразових паролів реалізована за допомогою алгоритму TOTP.

### 2.1. Алгоритм TOTP

TOTP (Time-based One Time Password Algorithm) [4] – алгоритм створення одноразових паролів для захищеної автентифікації, є покращенням алгоритму HOTP (HMAC-Based One-Time Password Algorithm) [5]. TOTP алгоритм – алгоритмом односторонньої автентифікації. Головна відміна TOTP від HOTP – це введення часу як параметру. При чому вводиться не точно вказаний час, а інтервал з заданими попередньо межами.

Нехай:

$T$  – дискретне значення часу, що використовується як параметр

$X$  – інтервал часу, протягом якого пароль дійсний (за замовчуванням 30 секунд)

$T_0$  – початковий час, необхідний для синхронізації сторін

$K$  – спільний секрет

$C$  – поточний час

Тоді

$$T = (C - T_0) / X,$$

$$HOTP(K, T) = \text{Truncate}(SHA1(K, T)),$$

$$TOTP = HOTP(K, T)$$

де  $SHA1(K, T)$  – генерація 20-ти байт на основі секретного ключа та часу за допомогою хеш-функції  $SHA - 1$ .

## 3. Дослідження одноразових паролів

Для отримання практичних результатів ефективності схеми двофакторної автентифікації реалізованої Google проведено аналіз основної частини двофакторної схеми автентифікації – одноразових паролів.

Для отримання достатньої множини одноразових паролів було написано автоматизовану систему, базовану на інструменті для автоматизації роботи в web-браузері Selenium. Система автоматично кожні 7 хвилин запускала приватну вкладку web-браузера Google Chrome, переходила на сторінку gmail.com та вводила логін та пароль попередньо зареєстрованих акаунтів Google з ввімкненою двофакторною автентифікацією. SMS, що містили одноразові паролі, накопичувались на мобільному телефоні і в подальшому конвертувались в більш придатний для аналізу вигляд.

Було отримано 3246 одноразових паролів, проте лише 30% з них було унікальними. Множина унікальних паролів не пройшла стандартний тест на випадковість [6], оскільки жоден з паролів не починався на 0.

## 4. Отримані результати

Було зроблено наступне спостереження відносно інвалідації одноразових паролів: якщо ми не закінчили успішно процес двофакторної автентифікації, то Google повторно створює той же самий одноразовий пароль для послідовності автентифікаційних спроб. Google анулює одноразові паролі лише після того як мине півгодини від першої спроби автентифікуватися чи після успішного завершення двофакторної автентифікації. Навіть якщо змінювати IP адрес, операційну систему чи браузер користувача, що робить спроби автентифікуватися, Google відправить той самий пароль. Атакувальник може використати цю вразливість наступним чином: перехопити одноразовий пароль та одночасно запобігти завершенню двофакторної автентифікації користувача. Відповідно отриманий одноразовий пароль залишається дійсним. Атакувальник може використати цей одноразовий пароль в окремій сесії автентифікації, оскільки Google очікує саме цей пароль. Схожі атаки залишаються можливими навіть, якщо одноразовий пароль є не дійсним, але вимагають набагато більших практичних навичок, оскільки у відповідності до рекомендацій стандарту час життя одноразового паролю має бути рівним 30 секундам [4].

## Висновки

В ході виконання даної роботи було знайдено вразливість схеми двофакторної автентифікації у Google та описано можливу атаку на неї. В подальшому планується провести аналогічне дослідження схем двофакторної автентифікації інших інтернет-сервісів.

## Перелік використаних джерел

1. Bonneau J., Schechter S. Towards reliable storage of 56-bit secrets in human memory. In 23rd USENIX Security Symposium (USENIX Security 14). — 2014. — P. 607–623.
2. Mobile Is Eating the World. — 2016. — URL: <http://www.slideshare.net/a16z/mobile-is-eating-the-world-2016>.
3. Fourth-factor authentication: somebody you know. / J. Brainard, A. Juels, R. L. Rivest, M. Szydlo. — P. 168–178.
4. TOTP: Time-Based One-Time Password Algorithm. — 2011. — URL: <https://tools.ietf.org/html/rfc6238>.
5. HOTP: An HMAC-Based One-Time Password Algorithm. — 2006. — URL: <https://tools.ietf.org/html/rfc4226>.
6. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications / J. Rukhin, A. Soto, J. Nechvatal, M. Smid et al. — National Institute of Standards and Technology, 2011. — URL: <http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf>.