УДК 651. 34 **doi: 10.26906/SUNZ.2019.3.080**

С. Ю. Гавриленко

Національний технічний університет «Харківський політехнічний інститут», Харків, Україна

МОДЕЛЮВАННЯ ТА ДОСЛІДЖЕННЯ АВТЕНТИФІКАТОРА ВЕБ-САЙТІВ НА ОСНОВІ SRP-ПРОТОКОЛУ

Проведено аналіз моделювання та дослідження автентифікатора веб-сайтів на основі SRP-протоколу. Доведено що запропонований підхід є стійким до атак посередника. Клієнт, за протоколом SRP не відправляє пароль користувача на сервер, а обчислює на його основі ключ. Маючи верифікатор пароля, отриманий при реєстрації, сервер також може обчислити цей ключ. По відкритому каналу передається не сам ключ, а спеціальні перевірочні значення. Розроблено мобільний клієнтський додаток для ОС Adroid та фремворк для серверу на мові PHP. Додаток виконує процедури реєстрації та автентифікації користувача відповідно до протоколу SRP. Оскільки передбачається, що користувач буде працювати із веб-сайтом з іншого пристрою, додаток генерую спеціальне значення ключа доступу на основі ключа сесії. Для доступу до сайту з будь-якого пристрою у формі автентифікації клієнт вводить значення не самого паролю, а ключ доступу, який надсилається на сервер і перевіряється.

Ключові слова: автентифікація, веб-сайти, безпека інформації, SRP-протокол.

Постановка проблеми

У наш час існує велика кількість різноманітних веб-сайтів та веб-додатків, що спрощують наше життя і дозволяють отримувати актуальну інформацію, обмінюватися даними, здійснювати покупки, тощо. При цьому, у процесі експлуатації веб-сайтів виникає необхідність збереження різних даних користувачів. Деякі веб-додатки зберігають персональні дані про користувача, такі як прізвище, ім'я, побатькові, номер телефону і т.д. Деякі зберігають фінансові дані користувача, такі як реквізити платіжних систем, номер банківського рахунку, тощо. Також, деякі веб-сайти зберігають стан балансу користувача в системі. З такою великою кількістю даних виникає необхідність у надійній системі захисту цих даних, адже їх отримання сторонніми особами може нанести суттєву шкоду, у першу чергу матеріальну, як користувачам, так і власникам вебсайту. Однією з процедур, що повинні забезпечувати безпечне зберігання даних користувачів є процедура автентифікації.

Аналіз літератури [1–14] показав, що одним з підходів забезпечення послуги автентифікації вебсайтів є використання протоколу SRP. В ряді літературних джерел [1, 5] показано, що оскільки протокол SRP має захист від атаки посередника, та має у основі своїх обчислень захист даних, який забезпечує складність задачі дискретного логарифмування, то його можна використовувати як основу для системи безпечної автентифікації на веб-сайтах.

Подібна ідея розглядалась у статті [2]. При цьому для реалізації даного протоколу на клієнтській стороні було запропоновано використовувати мову JavaScript. Тобто, алгоритм обчислень клієнтської сторони надходив клієнту мовою JavaScript разом із усією HTML-сторінкою у якості відповіді від сервера.

Такий підхід має певні недоліки, оскільки при атаці посередника, можна не тільки отримувати або відправляти від чогось імені дані (від імені клієнта або сервера), а також модифікувати їх. Таким чином, посередник може видалити код мовою JavaScript перед надсиланням сторінки клієнту. Отже, ввівши дані до форми автентифікації, та відправивши їх на сервер, клієнт передасть свій пароль у відкритому вигляді (оскільки JavaScript не виконає необхідних обчислень з паролем). Така схема роботи веб-додатку та реалізації атаки посередника зводить нанівець весь математичний апарат протоколу SRP, який повинен сприяти захисту даних від атак посередника. Отже, пароль клієнта повинен бути завжди закодованим перед надісланням серверу. Одним з можливих рішень даної задачі є розробка додатку для смартфонів з Android ОС.

Основна частина

Принципи функціонування автентифікатора на основі SRP-протоколу. При дослідженні виникає наступна задача: необхідно зв'язати пристрій, на якому користувач здійснює доступ до сайту (комп'ютер, ноутбук) із результатом автентифікації користувача на сервері через мобільний додаток. У звичайній ситуації, при успішній автентифікації через форми, сервер створює для користувача нову сесію, і зберігає ідентифікатор цієї сесії до файлу cookie. У ситуації ж з додатком таке рішення неможливе не тільки з приводу того, що користувач може використовувати інший пристрій для доступу до веб-сайту, а ще й тому, що для реалізації такої процедури необхідно було б створити власний браузер із підтримкою протоколу SRP.

Тому було вирішено, що додаток для мобільного пристрою генерує спеціальний код, і вводить його на пристрої, де необхідно розпочати сесію (для підтвердження того, що користувач успішно пройшов процедуру автентифікації). Код генерується на основі частини від ключа K [6] за допомогою хеш-функції. Це зроблено для того, щоб додатково ускладнити відновлення ключа користувача K, а з нього – пароля користувача.

Слід зауважити, що такий підхід не зовсім захищає від атаки посередника. Додаток зможе захистити від отримання паролю посередником, однак, атака посередника може бути здійснена для отримання ідентифікатору сесії, що буде передаватися від

браузера клієнта серверу. Захист ідентифікатора сесії потребує окремих досліджень, оскільки використовувати протокол SRP для шифрування та передачі ідентифікатора сесії є неоптимальним, оскільки для реалізації протоколу використовуються великі числа. Так, наприклад, при обчисленні верифікатора пароля генератор групи зводиться у ступінь результату хеш функції. Оскільки для тестування було взято хеш-функцію SHA-256, яка на виході дає 256-розрядне число, то і генератор необхідно звести у ступінь 256-бітного числа. Навіть при використанні 1024-бітної групи затримка при виконанні таких обчислень є помітною для користувача (до цього ще слід додати можливу повільну передачу даних у мережі). Отже, використання такого алгоритму для операції передачі ключа сесії, яка відбувається досить часто, при кожному оновленні сторінки, може суттєво вповільнити роботу та доступ до веб-сайту.

Окрім цього, алгоритм реалізації потребує захисту від атак на базу даних, оскільки саме в ній зберігаються дані про згенеровані ключі клієнтів до того моменту, як для заданого автентифікованого користувача буде відкрито нову сесію. Знаючи ключ K, згенерований у результаті роботи протоколу SRP та алгоритм генерації ключа сесії для користувача, зловмисник може видати себе за нього.

Також, можливе створення вірусу для мобільних пристроїв, на яких працюватиме додаток.

Реалізація клієнтської частини автентифікатора. Клієнтська частина автентифікатора представляє собою додаток для операційної системи Android не нижче 14 версії Android SDK.

Клієнтський додаток має 3 екрани (activity):

– MainActivity – головне вікно програми, яке дозволяє клієнту зробити вибір між необхідною процедурою: реєстрація або автентифікація;

– RegisterActivity – вікно програми, у якому здійснюється процедура реєстрації;

– LoginActivity – вікно програми, у якому здійснюється процедура автентифікації.

Кожне із зазначених вікон має відповідний файл мовою XML, у якому зазначене розташування графічних елементів, специфічних для ОС Android. До кожного файлу XML із графічним інтерфейсом є відповідний клас мовою Java, який містить програмну реалізацію процедур створення цих вікон, оброки дій (переважно, натискання на кнопку), обчислення та передачі даних.

Ще одним Java-класом, важливим для реалізації даного автентифікатора є клас SRP.java, який виконує обчислення усіх необхідних даних за протоколом SRP. Оскільки у обчисленнях беруть участь великі числа (більші за 64 розряди), то для роботи із цими числами було використано спеціальний клас java. math. BigInteger. Характерною особливістю вікна із реєстрації користувача RegisterActivity є те, що воно має дві строки вводу пароля для перевірки правильності введення. Окрім цього, виконується перевірка достатньої складності пароля (у даному випадку, пароль має складатися не менше, ніж з 5 символів).

Клас SRP.java при реєстрації клієнта генерує сіль та обчислює верифікатор пароля. Далі, дані отримуються класом RegisterActivity.java, який відправляє запит про реєстрацію із заданими параметрами на сервер. Вікно реєстрації виводить текстове повідомлення, яке відображає результат реєстрації. Це або успішна реєстрація (повідомлення "You was registeres successfully"), або клієнта не було зареєстровано, оскільки у системі вже користувач із заданою адресою електронної пошти (повідомлення "User with email: already registered!").

Вікно автентифікації LoginActivity дозволяє клієнту отримати ключ автентифікації для сайту. При цьому усі необхідні параметри автентифікації також обчислює клас SRP.java, а клас LoginActivity.java (додаток В) забезпечує відправку та отримання даних від сервера. У результаті процедури автентифікації клієнту може бути виведено повідомлення про невідповідність даних вимогам протоколу SRP. Окрім цього, клієнту можуть виводитися дані про неправильність порівняння параметрів, обчислених клієнтом, та параметрів, обчислених сервером.

Якщо всі обчислення було здійснено правильно, то клієнту виводиться код доступу, за яким на іншому пристрої буде створено нову сесію. Кодом є останні є останні 10 байт від результату хешування частини ключа K (останні 16 байт) та солі. 10 байт беруться для того, щоб користувачу було не складно вводити, але при цьому уникнути повторень ключів у користувачів сайту. Останні 16 байт ключа K беруться через те, щоб у разі розкриття початкової строки хеш-функції з ключа K не було спроб відновити пароль.

Реалізація серверної частини автентифікатора. Серверна частина постачається розробникам у якості фреймворка, файли якого необхідно додати до проекту. Серверна частина розроблена для веб-додатків, написаних на мові PHP версії не менше 5.6.

Серверна частина складається з наступних файлів: register.php; login.php; verify.php; auth.php; settings.php.

Передбачається, що каталог "srp_auth" із цими файлами буде додана до кореневого каталогу веб-сайту.

До файлу "register.php" клієнтський додаток звертається при реєстрації клієнта. Сценарій цього файлу лише зберігає дані про нового користувача, а саме: адресу електронної пошти, верифікатор пароля та сіль. Окрім цього скрипт виконує перевірку того, що користувач із заданою адресою електронної пошти не був раніше зареєстрований на цьому сайті.

Сценарій файлу "login.php" передбачає початок процедури автентифікації на веб-сайті. На цьому етапі користувач надсилає серверу значення електронної пошти I та обчислене значення A. Сервер, у свою чергу, надсилає значення солі та згенероване значення B. Після цього здійснюється обчислення ключа сесії K за протоколом SRP, а також обчислюються перевірочні значення M, R із наступним занесенням до бази даних.

Після того, як клієнтський додаток обчислить власні перевірочні значення M, R, він надсилає запит до скрипту "verify.php" із значенням M. Якщо значення M, отримане від користувача і обчислене сервером співпадають, сервер надсилає користувачу перевірочне значення R.

Файл "auth.php" виконує автентифікацію користувача на сайті. До цього надходить запит на автентифікацію із HTML-форми. Порівнюється електронна пошта користувача та ключ сесії із обчисленим на основі ключа K. Якщо надіслано правильні дані, то для заданого користувача у сесію розміщується адреса його електронної пошти. При подальшому використанні для перевірки того, чи був користувач автентифікований, виконується перевірка, чи встановлено встановлено змінна сесії "user".

Окрім цього, для роботи фреймворка необхідно створити у базі даних дві таблиці. Перша - "users" містить постійні дані про клієнтів (адресу електронної пошти, верифікатор пароля та сіль). Друга таблиця – "session" зберігає тимчасові дані про процедуру автентифікації (зберігає обчислений ключ K за протоколом SRP, а також параметри перевірки M, R. після успішного відкриття сесії ці дані видаляються. Таблиця "users" має наступну структуру, наведену на рис 1. Структуру таблиці "session" наведено на рис. 2.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно
1	**id**	bigint(20)		UNSIGNED	Нет	*Нет*		AUTO_INCREMENT
2	**username**	tinytext	utf8_general_ci		Нет	*Нет*		
3	**salt**	text	utf8_general_ci		Нет	*Нет*		
4	**verifier**	text	utf8_general_ci		Нет	*Нет*		

Рис. 1. Структура таблиці "users

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно
1	**record**	bigint(20)		UNSIGNED	Нет	*Нет*		AUTO_INCREMENT
2	**user**	tinytext	utf8_general_ci		Нет	*Нет*		
3	**R**	text	utf8_general_ci		Нет	*Нет*		
4	**M**	text	utf8_general_ci		Нет	*Нет*		
5	**K**	tinytext	utf8_general_ci		Нет	*Нет*		
6	**generation_time**	timestamp			Нет	CURRENT_TIMESTAMP		

Рис. 2. Структура таблиці "session"

Процедура автентифікації. Для початку процедури автентифікації у головному вікні програми клієнт повинен натиснути кнопку "Log In", після чого ввести необхідні дані до форми автентифікації.

Далі і клієнтський додаток і сервер обчислюють параметри процедури автентифікації за протоколом SRP.

Спочатку користувач надсилає серверу значення електронної пошти I та обчислене значення A. Сервер, у свою чергу, надсилає значення солі та згенероване значення B. Після цього обидві сторони здійснюють обчислення ключа сесії K за протоколом SRP, а також обчислюють перевірочні значення M, R із наступним занесенням до бази даних.

Протягом обміну даних параметри клієнтського додатку зберігаються у екземплярі класу SRP.java (відображення обчислених даних наведено на рис. 3), а серверний скрипт зберігає дані, необхідні для подальшого виконання процедури автентифікації, до бази даних (рис. 4).

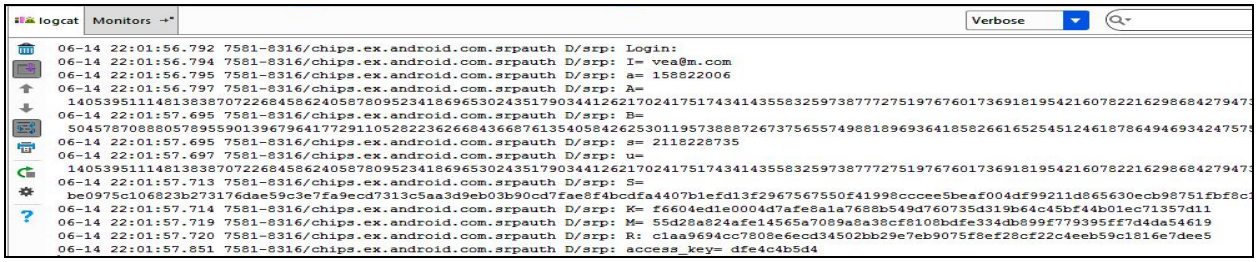

Рис. 3. Параметри процедури автентифікації, що обчислює клієнтський додаток

record	user	R	M	K
101	vea@m.com	c1aa9694cc7808e6ecd34502bb29e7eb9075f8ef28cf22c4ee...	55d28a824afe14565a7089a8a38cf8108bdfe334db899f7793...	f6604ed1e0004d7afe8a1a

Рис. 4. Параметри процедури автентифікації, що були обчислені та збережені сервером у БД

Висновки

Таким чином, проведено моделювання та дослідження автентифікатора веб-сайтів на основі SRP-протоколу.

Перевагами даного автентифікатора є те що:

– автентифікатор використовує алгоритм шифрування SRP, який є стійким до атаки посередника;

– реалізація у вигляді мобільного додатку захищає саму процедуру автентифікації від атаки посередника;

– надійність даних забезпечує складність задачі дискретного логарифмування.

Разом із цим, зазначимо, що автентифікатор має такі недоліки:

– можлива атака посередника із перехопленням ключа сесії вже після процедури автентифікації;

– необхідно захищати базу даних від атак, оскільки у ній містяться дані про ключі сесії користувачів, які автентифікувалися, але не відкрили сесію;

– перехоплення паролю у нешифрованому вигляді можливе при розробці спеціального вірусу для мобільної платформи.

СПИСОК ЛІТЕРАТУРИ

1. Виростков Д. Обзор способов и протоколов аутентификации в веб-приложениях [Електронний ресурс] / Д. Виростков // Хабр. – 2015. – Режим доступу до ресурсу: https://habr.com/company/dataart/blog/262817
2. Давлетханов М. Концепция одноразовых паролей в системе аутентификации [Електронний ресурс] / М. Давлетханов // BYTE. – 2006. – Режим доступу до ресурсу: https://www.bytemag.ru/articles/detail.php?ID=9101
3. Kuchuk G.A. An Approach To Development Of Complex Metric For Multiservice Network Security Assessment / G.A. Kuchuk, A.A. Kovalenko, A.A. Mozhaev // Statistical Methods Of Signal and Data Processing (SMSDP – 2010): Proc. Int. Conf., October 13-14, 2010.– Kiev: NAU, RED, IEEE Ukraine section joint SP, 2010. – P. 158 – 160.
4. Kuchuk G., Nechausov S., Kharchenko, V. Two-stage optimization of resource allocation for hybrid cloud data store. *International Conference on Information and Digital Technologies. Zilina*, 2015. P. 266-271. DOI: http://dx.doi.org/10.1109/DT.2015.7222982
5. Федоров А. SRP-6: аутентификация без передачи пароля. 2011. URL : https://habr.com/post/121021.
6. Фергюсон Н. Практическая криптографія / н. Фергюсон, Б. Шнайер, Вильямс 2004, 432 с.
7. Amin Salih M., Yuvaraj D., Sivaram M., Porkodi V. Detection And Removal Of Black Hole Attack In Mobile Ad Hoc Networks Using Grp Protocol. *International Journal of Advanced Research in Computer Science*. Vol. 9, No 6. P. 1–6, DOI: http://dx.doi.org/10.26483/ijarcs.v9i6.6335
8. Yogesh Awasthi, R P Agarwal, B K Sharma, "Intellectual property right protection of browser based software through watermarking technique", International Journal of Computer Applications, vol. 97, no. 12, 2014, pp. 32-36.
9. Yogesh Awasthi, R P Agarwal, B K Sharma, "Two Phase Watermarking for Security in Database", International Journal of Computing, vol. 4, no. 4, 2014, pp. 821-824
10. Saravanan S., Hailu M., Gouse G.M., Lavanya M., Vijaysai R. Optimized Secure Scan Flip Flop to Thwart Side Channel Attack in Crypto-Chip. International Conference on Advances of Science and Technology, ICAST 2018. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Vol 274. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-15357-1_34
11. Manikandan V, Porkodi V, Mohammed AS, Sivaram M, "Privacy Preserving Data Mining Using Threshold Based Fuzzy cmeans Clustering", ICTACT Journal on Soft Computing, Vol. 9, Is. 1, 2018, pp.1813-1816. DOI: 10.21917/ijsc.2018.0252
12. Ruban, I. Redistribution of base stations load in mobile communication networks / I. Ruban, H. Kuchuk, A. Kovalenko // Innovative technologies and scientific solutions for industries. – 2017. – No 1 (1) – P. 75-81. – DOI : https://doi.org/10.30837/2522-9818.2017.1.075
13. Teilor.D Using the Secure Remote Password (SRP) Protocol for TLS Authentication [Електронний ресурс] / Д. Тейлор // IETF. – 2007. – Режим доступу до ресурсу: https://tools.ietf.org/html/rfc5054#ref-MODP;
14. Semenov S. Identification of the state of an object under conditions of fuzzy input data / S. Semenov, O. Sira, S. Gavrylenko, N. Kuchuk // Eastern-european journal of enterprise technologies.– Kharkiv. 2019 No. 4(97). P.22-30 (SCOPUS)

Рецензент: д-р техн. наук, проф. С. Г. Семенов,
Національний технічний університет «Харківський політехнічний інститут», Харків
Received (Надійшла) 04.04.2019
Accepted for publication (Прийнята до друку) 15.05.2019

Modeling and studying authenticators of web sites based on SRP protocol

S. Gavrilenko

Analysis of simulation and research of site authenticator based on SRP-protocol. It is proved that the proposed approach is resistant to mediator attacks. The client, according to the SRP protocol, does not send the user password to the server, but calculates the key based on it. Having a password verifier received at registration, the server can also calculate this key. On the open channel is not the key itself, but the special checking values. Developed a mobile client application for the Android OS and Free make for the server in PHP. The application performs user registration and authentication procedures in accordance with the SRP protocol. Because it is assumed that the user will work with the website from another device, the application generates a special key value based on the session key. To access the site from any device in the form of authentication, the client enters the value of not the password itself, but the access key that is sent to the server and verifies.

Keywords: Authentication, Web Sites, Information Security, SRP-Protocol

Моделирование и исследование автентификатора веб-сайтов на основе SRP-протокола

С. Ю. Гавриленко

Проведен анализ моделирования и исследования автентификатора веб-сайтов на основе SRP-протокола. Доказано, что предложенный подход является устойчивым к атакам посредника. Клиент, по протоколу SRP не отправляйте пароль пользователя на сервер, а вычисляет на его основе ключ. Имея верификатор пароля, полученный при регистрации, сервер также может вычислить этот ключ. По открытому каналу передается не сам ключ, а специальные проверочные значения. Разработан мобильный клиентское приложение для ОС Android и фреймворк для сервера на языке PHP. Приложение выполняет процедуры регистрации и аутентификации пользователя в соответствии с протоколом SRP. Поскольку предполагается, что пользователь будет работать с сайтом с другого устройства, приложение генерируемого специальное значение ключа доступа на основе ключа сессии. Для доступа к сайту с любого устройства в форме аутентификации клиент вводит значения не самого пароля, а ключ доступа, который направляется на сервер и проверяется.

Ключевые слова: аутентификация, веб-сайты, безопасность информации, SRP-протокол