Список літератури

1. Fleischer CC, Wu J, Qiu D, Park SE, Nahab F, Dehkharghani S. The Brain Thermal Response as a Potential Neuroimaging Biomarker of Cerebrovascular Impairment. AJNR Am J Neuroradiol. 2017;38(11): 2044-2051. doi:10.3174/ajnr.A5380

2. Crook D. Power BI Embedded, IoT, and Machine Learning for brain thermal pattern recognition with BTT [Електронний ресурс] / David Crook // Technical Case Studies. – 2016. – Mode of access: https://microsoft.github.io/techcasestudies/iot/power%20bi%20embedded/2017/02/13/BTTCorp.html

3. Schvepsss. Power BI Embedded, IoT и машинное обучение для обработки термограмм мозга [Електронний ресурс] / Schvepsss // Хабр. – Режим доступу: https://habr.com/ru/company/microsoft/blog/323200/

УДК 004.056.5:[004.383.2:004.738.5

Якубич К. О., здобувач освіти,
Потапова Н.А., к.е.н., доцент,
доцент кафедри інформаційних технологій

БЕЗПЕКА У ВЕБ-СЕРВІСАХ

Донецький національний університет імені Василя Стуса, м. Вінниця

Безпека – важлива функція в будь-якій веб-програмі. Оскільки майже всі веб-програми відкриті для Інтернету, завжди існує ймовірність загрози безпеці веб-додатків. Отже, при розробці веб-додатків завжди рекомендується переконатися, що програма розроблена з урахуванням вимог безпеки. Щоб зрозуміти які загрози можуть бути, розглянемо простий сценарій веб-застосунку і розберемось, як він працює з точки зору безпеки.

Одним із заходів безпеки, доступних для HTTP, є протокол HTTPS - це безпечний спосіб зв'язку між клієнтом та сервером через Інтернет. HTTPS використовує рівень захищених сокетів або SSL для безпечного зв'язку. І клієнт, і сервер будуть мати цифровий сертифікат, щоб ідентифікувати себе як справжній, коли відбувається будь-який зв'язок між клієнтом та сервером. [1]

При стандартному зв'язку HTTPS між клієнтом та сервером виконуються такі кроки:

1. Клієнт надсилає запит на сервер через сертифікат клієнта. Коли сервер бачить сертифікат, він робить запис у своїй системі кешування, щоб він знав, що відповідь має повертатись тільки цьому користувачу.

2. Потім сервер автентифікує себе клієнтом, надсилаючи свій сертифікат. Це гарантує, що клієнт спілкується з сервером.

3. Після цього весь зв'язок між клієнтом та сервером шифрується. Це гарантує, що інші користувачі не зможуть прочитати чи отримати дані.

Але вищезгаданий тип безпеки не працюватиме у всіх ситуаціях. Може

настати момент, коли клієнт може спілкуватися з кількома серверами. Приклад, наведений нижче, показує, що клієнт одночасно звертається до бази даних та веб-сервера. У такому випадку, не вся інформація може проходити через протокол https.

Тут SOAP набирає чинності, щоб подолати такі перешкоди, маючи специфікацію WS Security.

SOAP – це скорочення від Simple Object Access Protocol. Це протокол обміну повідомленнями на основі XML для обміну інформацією між комп'ютерами. SOAP є програмою специфікації XML. [1]

WS Security – це стандарт, який вирішує питання безпеки під час обміну даними в рамках веб-служби. Це ключова функція SOAP, яка робить його дуже популярним для створення веб-служб. [2]

Необов'язковий елемент SOAP-заголовок містить специфічну для програми інформацію про повідомлення SOAP (наприклад, аутентифікації, оплати тощо). [3]

Елемент SOAP-заголовок може містити наведену нижче інформацію

1. Якщо повідомлення SOAP було підписано будь-яким ключем безпеки, цей ключ можна визначити в елементі заголовка.

2. Якщо якийсь елемент у тілі SOAP зашифрований, SOAP-заголовок міститиме необхідні ключі шифрування, щоб можна було розшифрувати повідомлення, коли воно досягне пункту призначення.

У середовищі з кількома серверами вищезазначена техніка аутентифікації SOAP допомагає в таким чином.

1. Оскільки тіло SOAP зашифроване, його може розшифрувати лише веб-сервер, на якому розміщена веб-служба.

2. Припустимо, що якщо повідомлення передається на сервер бази даних у вигляді запиту HTTP, його неможливо розшифрувати, оскільки в базі даних немає відповідних механізмів для цього.

3. Тільки коли запит дійсно досягає веб-сервера у вигляді протоколу SOAP, він зможе розшифрувати повідомлення та надіслати відповідну відповідь клієнту.

Стандарт WS-Security обертається навколо включення визначення безпеки в SOAP-заголовок. Облікові дані в SOAP управляються двома способами:

− По-перше, він визначає спеціальний елемент з ім'ям UsernameToken. Це використовується для передачі імені користувача та пароля на веб-сервіс.

− Другий спосіб – використовувати двійковий токен через BinarySecurityToken. Це використовується в ситуаціях, коли використовуються методи шифрування, такі як Kerberos або X.509.

Запит може бути відправлений від клієнта веб-служби до служби маркерів безпеки. Цей сервіс може бути проміжним веб-сервісом, спеціально створеним для надання імен користувачів / паролів або сертифікатів для реального веб-сервісу SOAP. Токен безпеки передається клієнту веб-служби. Клієнт викликає веб-службу, але цього разу гарантує, що токен безпеки вбудований у повідомлення SOAP. Потім веб-служба розпізнає повідомлення SOAP з токеном автентифікації і може зв'язатися зі службою токенів безпеки, щоб

дізнатися, чи токен безпеки автентичний чи ні.

Нижче наведений фрагмент коду показує формат частини автентифікації, яка є частиною документа WSDL. Тепер, ґрунтуючись на наведеному нижче фрагменті, повідомлення SOAP міститиме 2 додаткових елементи, один з яких буде ім'ям користувача, а інший паролем.

```
<xs:element name="UsernameToken">
        <xs:complexType>
                <xs:sequence>
                        <xs:element ref="Username"/>
                        <xs:element ref="Password" minOccurs="0"/>
                </xs:sequence>
        <xs:attribute name="Id" type="xs:ID"/>
</xs:complexType></xs:element>
```

Коли повідомлення SOAP фактично передається між клієнтами та сервером, частина повідомлення, що містить облікові дані користувача, може виглядати так, як показано вище. Ім'я елемента WSSE - це спеціальний елемент з ім'ям, визначеним для SOAP, і означає, що він містить інформацію, що базується на безпеці.

Список літератури

1. *SOAP. URL: https://coderlessons.com/tutorials/akademicheskii/izuchite-soap/soap-kratkoe-rukovodstvo.*
2. *SOAP Версія 1.2 Частина 0: Підручник для початківців. URL: https://www.w3.org/TR/soap12-part0/.*
3. *w3big. URL: http://www.w3big.com/ru/soap/soap-header.html .*
4. *O'Neill Mark. Web service (XML-RPC,SOAP, SOA) security documentation. URL: https://www.cgisecurity.com/ws.html.*