Implementation of Rivest Cypher 4 algorithm in Security Assertion Mark-up Language protocols on Single Sign-On services

C Slamet*, U Syaripudin, F M Kaffah and B E Tiasto

Informatics Department, Faculty of Science and Technology, UIN Sunan Gunung Djati Bandung, Jl. A.H. Nasution No. 105 Cibiru Bandung, Indonesia

*cepy_lucky@uinsgd.ac.id

Abstract. Single Sign-On (SSO) is an authentication service that allows users to use a set of credential data to access multiple applications. The SSO can be implemented using Security Assertion Mark-up Language (SAML) which is recognized as a framework or standard for sending open messages that allow identity and security information to be shared to each entity. Credential data sent over the network enable unauthorized users to capture the data and this will be fatal since the data can be used to access all applications in the SSO entity. This study aims to propose a credible data security approach that is reliable so that the data is not easily read by unauthorized users even with the dictionary attacks. This security method is performed using the Rivest Cypher 4 (RC4) algorithm. This study is resulting in two major phenomena, (1) the utilization of the RC4 algorithm on SAML in SSO services has proven to be an effective data security approach for dictionary attacks, (2) the formulation of the RC4 algorithm on SAML has proven to not interfere with SSO services in the user authentication process.

1. Introduction

The development of information systems become one of the most interesting studies for both practitioners and researcher [1]. As a consequence, a good level of security in using many applications are demanded to be more reliable [2,3]. The utilized security system in accessing applications must also concern the user conveniences. On the other hand, some applications require specific authentication processes to guarantee their user securities. This results in the users are required to have many different accounts to access each application, this requirement is making the users uncomfortable since it pushing up them to loop the authentication processes. These problems can be solved by using one account to access fulfilling applications, this system is called Single Sign-On (SSO) system [4,5].

SSO supports one authentication process by using a set of accounts to access many applications. SSO can be implemented using several protocols, one of which is Security Assertion Mark-up Language (SAML). The SAML protocol allows account data and some user data attributes to be exchanged among system entities over the network [6]. The problem is the data exchange among entities of the system across the network can be intercepted by third parties (unauthorized users) and can be reused to access all applications within SSO entities which is the credential data is text-based.
formatted that would be easy to read. Referring to the types of credential data that can be used, this study is selecting text-based usernames and passwords by applying cryptographic methods to secure data [5] and using Rivest Cypher 4 (RC4) algorithm to secure a decryption penetration of a dictionary attack [7].

Single Sign-On system is one of the services that is considered efficient for user integration in the entire services, while authentication security is known as a major factor in securing services since it is the first step to run a system [8]. In this study, the author intends to increase the security theme so that it can be implemented in security services.

2. Literature review

2.1. Single Sign-On (SSO)

Single Sign-On (SSO) is an authentication service that allows users to use a set of credential data (username, password) in accessing several applications. SSO can be used by companies, organizations, and also individually to reduce the management of various credential data. Applications that are connected in this service have a module that retrieves credential data from a central SSO server, while repositories of user data storage are usually stored in a special directory, for example, Lightweight Directory Access Protocol (LDAP). This service authenticates users for all applications by granting access rights and eliminates login requests to re-access other applications while still in the same session [8].

2.2. Security Assertion Mark-Up Language (SAML)

SAML is a framework or open message delivery standard that allows identity and security information to be shared through the security domain, this standard is written in XML format [9,10]. SAML is needed by a system to be able to exchange its security information to other systems, not only this standard SSO can also be applied for other purposes. The following are some examples of the application of the SAML standard including single sign-on, federation identity, and web services.

In a minimum condition, the SAML exchange involves two entities namely: the party sending, and the party receiving. In some case using this standard, users who might use a web browser or application running SAML, are also entities, and may even act as the sending party. The sending party is a system entity that makes a SAML statement, this party is commonly called a SAML authority. While the recipient is an entity that uses statements that have been made [4]. To support the SSO system, SAML defines entities that act as Identity Providers (IdP), and entities that act as Service Providers. The statement contained in IdP contains information about user data such as e-mail addresses. The user is authenticated to enter the system using a password, then the Service Provider can use the email information to determine user access rights [11].

2.3. RC4

RC4 is a stream cypher cryptographic algorithm, which encrypts each digit of the plaintext one by one, and uses a simple encryption transformation that is key-dependent. In practice, the digits used are usually bits or bytes. This stream cypher is almost always faster and uses far fewer keys than the block cypher algorithm, and RC4 is the most widely used stream cypher today because of its simplicity and high efficiency. RC4 was also chosen as an encryption algorithm in several communication schemes for Radio Frequency Identification (RFID) [12].

This algorithm was designed in 1987 by Ron Rivest for RSA data security (RC stands for Ron’s Code). The design has been a trade secret ever since and began posting anonymously on the web in 1994 [13]. One of the main ideas for constructing a stream cypher is to depend on creating pseudo-random permutations which then extract pseudo-random word sequences from the permutations. RC4 follows this principle to extract pseudo-random bytes from the permutation. RC4 stream cypher consists of two main components, namely Key Scheduling Algorithm (KSA), and Pseudorandom Generation Algorithm (PRGA) [13].
Algorithm 1: Key Scheduling Algorithm (KSA)

\[
\begin{align*}
&\text{for } i = 0 \text{ to } 255 \text{ do} \\
&S[i] = i; \\
&K[i] = k[i \mod l]; \\
&j = 0; \\
&\text{for } i = 0 \text{ to } 255 \text{ do} \\
&j = (j + S[i] + K[i]) \mod 256; \\
&\text{swap } S[i], S[j]; \\
\end{align*}
\]

Algorithm 2: Pseudo-random Generation Algorithm (PRGA)

\[
\begin{align*}
i, j &= 0; \\
\text{while } (\text{true}) \\
i &= (i + 1) \mod 256; \\
j &= (j + S[i]) \mod 256; \\
\text{swap } S[i], S[j]; \\
\text{Output: } k &= S[S[i] + S[j]] \mod 256; \\
\end{align*}
\]

Internal states in RC4 contain permutations of 8-bit words \((N = 2^8 = 256 \text{ bytes})\), and KSA produces the initial pseudo-random permutations of RC4 by randomizing identity permutations using a secret key. The secret key \((k)\) of RC4 has a length typically of 5 to 32 bytes, which then produces a key \((K)\) of length \(N = 256 \text{ bytes}\) with simple repetition. If the key length \(k\) is 1 byte (usually \(5 \leq 1 \leq 32\)), then the key is built as \(K[i] = k[i \mod l]\) for \(0 \leq i \leq N - 1\). The initial permutation produced by KSA acts as an input to the PRGA process which then produces a keystream \([7]\). PRGA is a recurring procedure where each repetition produces 1 byte of pseudo-random as a keystream which is then XORed with 1 byte of the plaintext, while a 256-byte S permutation and both 1-byte index \(i\) and \(j\) are updated \([13]\).

3. Analysis and design

SAML is sent via several network protocols where packets exchanged between these systems can be captured, and if the cryptographic algorithm used is one-way encryption it will be very easy for the attacker to get the original plaintext from the credential data. Several types of credential data can be used as mentioned in Irwansyah et al. \([14]\). We still choose to use the text data type (username and password) in the Something you know type, by using the RC4 algorithm to encrypt credential data.

Figure 1. The system architecture.

The system architecture describes the relationships between entities in the SSO system along with the application and also the packet delivery flow. The system architecture design that can be seen in Figure 1, there are two servers, each of which has an application, SAML, RC4 module, and a login form. Credential data information is exchanged between entities through a SAML packet that has been encrypted by the RC4 module contained in each application.
4. Result and discussion
To find out the extent of the compatibility of the results of the study with the problem at hand, we conducted two systems tests, namely the data encryption test, and system authentication test. Each of these tests was carried out to ascertain whether implementing the RC4 algorithm could secure user attribute data from a dictionary attack, and to ensure the SSO system could still run with the application of the cryptographic algorithm.

4.1. Data encryption test
How to find out the Cyphertext generated by the RC4 algorithm cannot be solved is to do a dictionary attack technique, then capture data using SAML tracer, then later the data will be tried to be described using an online hash-decryption application.

Table 1. Testing data encryption.

| No. | Attribute | Cyphertext   | Result     |
|-----|-----------|--------------|------------|
| 1   | Username  | 602716c8929e | Not Found  |
| 2   | Password  | 44201fc8b18d60787c5e6e37820676ed9b19 | Not Found |
| 3   | Nama      | 672816da80cc6263687c6122c5136db5d04c52 | Not Found |
| 4   | Alamat    | 662013c68199276566776922 | Not Found |

Table 1 shows that none of the four data described was successfully resolved by the dictionary attack.

4.2. System authentication testing
The system built is a centralized authentication system, then testing the authentication using a sampling account data from the user with the black box testing scheme. This test is done to ascertain whether the authentication system can continue to run with the implementation of the RC4 algorithm in securing credential data on the system. The results of this test can be seen in the following table:

Table 2. System authentication testing.

| No. | username | password | Information  | Result         |
|-----|----------|----------|--------------|----------------|
| 1   | bagus    | Bagus    | password correct | Login successful |
| 2   | enggar   | enggar   | password correct | Login successful |
| 3   | bagus    | enggar   | password wrong | Login unsuccessful |
| 4   | enggar   | bagus    | password wrong | Login unsuccessful |

5. Conclusion
Based on the implementation of the RC4 algorithm in the SAML protocol on the SSO service, it can be concluded that the RC4 algorithm can be used to secure data in the SAML protocol. The algorithm implementation is written using server programming dan and performance RC4 algorithm can secure credential data on SSO services from a dictionary attack then a binary cypher key is needed to encrypt a character in the plaintext. RC4 is a stream cypher that encrypts plaintext with a key in each digit. The Algorithm (RC4) produces some Cyphertext characters that cannot be read by the database and also cannot be written using XML so it requires conversion of characters to hexadecimal numbers.

References
[1] Slamet C and Ramdhani M A 2019 Concepts of Strategic Management of Information Technology Journal of Physics: Conference Series 1175(1) 012264
[2] Coulouris G F, Dollimore J and Kindberg T 2005 Distributed systems: concepts and design (Pearson)
[3] Tanenbaum A S and Van Steen M 2007 Distributed systems: principles and paradigms (Amsterdam: Prentice Hall)
[4] M Rouse 2019 Single Sign-On (Tech Target)
[5] Telnoni P, Munir R and Rosmansyah Y 2014 SAML single sign-on protocol development using combination of speech and speaker recognition 2014 International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA) 299-304

[6] Ragouzis N, Hughes J, Philpott R, Maler E, Madsen P and Scavo T 2008 Security Assertion Markup Language (SAML) V20 Technical Overview (Massachusetts: OASIS)

[7] Daswani N, Kern C and Kesavan A 2013 Foundations of Security: What Every Programmer Needs to Know (New York: Apress)

[8] Majumder J 2012 Dictionary Attack on MD5 hash International Journal of Engineering Research and Applications 2(3) 721-724

[9] Groß T 2003 Security analysis of the SAML single sign-on browser/artifact profile In 19th Annual Computer Security Applications Conference, 2003 Proceedings 298-307

[10] Campbell B, Mortimore C and Jones M 2015 Security assertion markup language (SAML) 2.0 profile for OAuth 2.0 client authentication and authorization grants Internet Engineering Task Force (IETF)

[11] Rivest R L 2990 Cryptology (Massachusetts: MIT Laboratorium for Computer Science)

[12] Yu Q, Zhang C N, Orumiehchiha M A and Li H 2012 RC4-BHF: An improved RC4-based hash function In 2012 IEEE 12th International Conference on Computer and Information Technology 322-326

[13] Gupta S S 2013 Analysis and Implementation of RC4 Stream Cipher Indian Statistical Institute, India

[14] Irwansyah F S, Slamet C and Ramdhani M A 2018 The Analysis of Determinant Factors in Selecting Laboratory Equipment in Chemistry Education Experiment Chemical Engineering Transaction 63 793-798