A Novel Message Authentication and Key Generation Model for Secure transmission of Data

Pudota Krishna Divya¹, Sindhe Swetha²

Final M.tech Student¹, Assistant Professor²

Dept of CSE, Dadi Institute of Eng. and Technology (jntuk), AP, India.

Abstract:
In present networks the data integrity is very complex in sending the messages. There are many security issues such as data leakage, corruption of data while data transfer. In transferring the data the message sender and receiver verification is also a hard task to users. So we proposed a method that consists of key generation and key transferring method transfer securely over secure channel. Our proposed approach improves the efficiency and more security of transferring the message.

I.INTRODUCTION

Data privacy is the main privilege to every user who are using the network services. In this data leakage is very big problem to users to secure their data in the transfer of the network. For this user authentication plays main role to achieve the security of the data in the network. For this researchers introduced text authentication, image authentication, video authentication and the voice authentication. But in any type of the authentication method converts the data to binary level data. [1]

So many data verification and the authentication methods are based on the binary level methods. In any of the authentication and the verification methods the data and authentication code is manipulated to bits. In this paper we proposed the methods based on the bit level only. There are many symmetric methods and asymmetric methods which are many operations based on the bit level data. For this researches suggested the hash methods.

Note that the key need not to be as long as the message, otherwise, such constructions will be impractical. That is, there are standard techniques so that the same key can be used to hash messages of arbitrary lengths [5, 9, 10]. The security of universal hash-function families based MACs depends on the probability of message collision. That is, if two distinct messages m and m₀ hash to the same image (i.e., h(m) = h(m₀)), then they will have the same authentication tag. Consequently, for a message-tag pair, if an adversary can come up with a different message that hashes to the same value, successful forgery can be accomplished with high probabilities. Therefore, the most important security property of universal hash functions is their probabilities of message collisions.

the security of the construction restricts the computations to be performed over an integer field. What we will describe here is a generalization of the construction appeared in [2], in which we allow operations to be performed over a finite integer ring instead of a field [3, 1]. As in the computationally secure constructions the codes in demand that operations must be performed over the integer field Zp; no previous work has studied the probability of deception of such codes when computations are performed over arbitrary finite integer rings. Other codes with secrecy include, but are not limited to

II. RELATED WORK

In previous works, there are many methods in data security issues. We have two types of attacks are based on human identity and the method based attacks. In human identity attacks corrupted users or un-authorized users entered in the network and changes the original data. There is also another way of this attack intruder changes the data and resend the data to authorized user. In another type of network attacks data corruption methods are used to attack the user authentication and the data. [8]

For this attacks the researches introduced symmetric cryptography methods and asymmetric methods. The symmetric methods consists of the arithmetical methods, hash based methods. The data is sending through secure channel using encoded methods by generating encryption of the data. Then the data verification used by the signature methods such as digital signature methods and the curve cryptographic methods.

In addition to research on encryption and signing in the presence of key-dependent messages, leakage resilience is of interest for the context of our paper [1, 2]. Leakage functions are used to model leaked information as occurring during a side-channel attack, which may include information about the secret key. Unlike in the case of typical leakage functions, the functions f that we allow the adversary to query may leak a complete secret state. However, in our setting an adversary does not obtain output values of f directly, but rather the result of
the tag generation algorithm when being applied to images under $f$, thus our discussion seems more adequate for dealing with 'structural' than with side-channel attacks.[6,7]

Informally, a MAC is KD-EUF (key-dependent existentially unforgeable) secure if it is secure despite a forger's ability to obtain tags on arbitrary (efficiently computable) functions $g$ of the state $K$. We begin by making this intuition more precise and then show how to achieve this security requirement in the random oracle model. While one may be tempted to think that the use of a random oracle makes the construction of a MAC trivial, the presence of key-dependent queries changes the situation significantly even with a random oracle there is no stateless KD-EUF-secure MAC.

III. PROPOSED WORK

In this section we are mainly proposed message authentication approach that is faster than the existing approach. Before performing message authentication the key generation center will generate secret key and sent to public channel members for message encryption and decryption. After generating secret key the channel member or group member will send message to specified member of the group. Before sending message the group member will encrypt message and generate signature for that message. After completion of encryption and signature generation the group member will send message and signature to specified member of the group. The specified group member will retrieve the cipher message and signature. After retrieving the group member again generate signature and compare both signature are equal the message is authenticated or not equal it will block the message. The following concepts are specifying generation of secret key, encryption and decryption of message and generate signature for encrypted message.

Users Registration:

This module explains the process computation of key and users registration. After registering users the KGC will generate id for individual users $U_i$ and sent to users. During registration process each user will choose a random secret value $S_i$ and send to KGC. Once user registration process is completed, KGC assigns a permanent secret id, denoted by $P_i$ for each Member $U_i$ in the group.

Key generation and distribution to group member:

In this module each user will request for group key, the KGC will randomly generate secret. After generating secret key the KGC will send that key to each user with in secure manner. By providing security for secret key the KGC will generate a message and send to all users. The KGC will now generation of message and its value is calculated by using following formula.

$$\text{Message}=(P_1 \odot S_1) \times (P_2 \odot S_2) \times \ldots \times (P_n \odot S_n) + \text{secret key}.$$  

After generating message the KGC will send the message to all group members. The group member will retrieve the message get secret key from the message. Upon receiving the message $M$, each member in the groups will generate the key in the following manner.

$$\text{Secret key} = M \mod (P_i \circ S_i) \text{ for all } i.$$  

After completion of secret key each user will encrypt the message by using following algorithm.

Reverse Binary xor Encryption Algorithm:

We will be presenting the steps of the encryption algorithm of the reverse binary xor Algorithm. The following steps are as shown below:

1. Input secret key and transferring message to encryption process.
2. Get each character from the message and convert into ASCII values.
3. After converting ascii values each value xor with key until the length of message is completed.
4. The completion of xor operation each ascii value can converted into binary format.
5. Reverse previous binary data until completion length of message.
6. After reversing binary data that data can be perform once complement.
7. The previous complement data convert into ascii format.
8. Divide each ascii value by secret key and get remainder and coefficient until completion of length of message.
9. Each character of remainder and coefficient become one point and those points send to specified group member.

Before sending the cipher data to specified group member the user will generate signature for encrypted message by using MD5 algorithm. After generating signature the user will send cipher message and signature to specified group member. The specified group member will retrieve the cipher
message and signature and again will generate signature for cipher message. The group member will compare both signatures are equal the message is authenticated otherwise the message will corrupt and block the message. If the message is authenticated then specified group member will retrieve cipher message and get the original message by performing decryption process of reverse binary xor encryption algorithm. The Decryption process of Reverse binary xor encryption algorithm as follows.

1. Retrieve the point from the sender group member.
2. Get the single ascii value from the point by using following formula.
   \[
   \text{Asciival} = \text{quotient} \times \text{secretkey} + \text{remainder}.
   \]
3. The previous asciiwill be convert into binary format.
4. The previous binary data can be perform the once complement.
5. After performing once complement that binary data will be reverse until the completion message length.
6. After reversing that binary data will convert into ascii format.
7. The previous ascii values will be xor with secret key until completion of message length.
8. After performing xor operation that ascii values can be converted characters get the original message.

**Conclusion:**

In this paper we proposed secure authentication and secure data transfer in network channel. We implemented simple message conversions and authentication methods. In this bit level secure method of encoding is implemented and authentication method is also generates strong signature to break. It reduces maximum data attacks and secure the data transfer. Before performing authentication of encrypted message the KGC will generate secret key and send to all group member. Each user will use the secret key for encryption and decryption process . By implementing these concepts we can improve efficiency and security of transferring message.

**REFERENCES**


**ALOMAIR AND POOVENDRAN: EFFICIENT AUTHENTICATION FOR MOBILE AND PERVERSIVE COMPUTING 479**


**BIographies**

Pudotakrishna Divya is studying M.Tech in dadi institute of engg and technology(jntuk) anakapalli, in department of cse.

Sindhe Swetha, she is working as Assistant Professor , Department of CSE, DIET (Dadi Institute of Engineering & Technology) college, Anakapalle