Key pre-distribution scheme over the continuous periodic functions space

S V Belim1,2, S Yu Belim1

1 Omsk State Technical University, 11 Mira ave., Omsk, 644050, Russia
2 Siberian State Automobile and Highway University, Omsk, 644080, Russia

E-mail: sbelim@mail.ru

Abstract. Implementation for the key preliminary distribution scheme in the continuous functions space is proposed. Key materials are formed based on linear differential operator. User characteristics are presented in the form of a harmonic series. We show that this approach allows you to develop a common encryption key. A linear differential operator is formed only from even order derivatives. Odd order derivatives do not affect the encryption key value.

1. Introduction

Key pre-distribution schemes are used as an alternative to asymmetric encryption when key information can be transmitted over a secure channel. These schemes are used to reduce the amount of key storage. Most modern key pre-distribution are based on the Blom’s scheme [1] and the KDP-scheme [2]. Key pre-distribution schemes are used in sensor networks [3-6] and wireless networks [7-9].

Key pre-distribution schemes are actively developing in various directions. Article [10] proposes a hierarchical scheme for key pre-distribution for large mobile networks with several clouds. This scheme allows you to establish secure connections both within an individual cloud and between clouds. The scheme is highly scalable at low memory costs. Article [11] proposes a key management scheme based on a quadratic polynomial. The schema uses the key set session key, group key, network key, and private key. The private key is set based on the Fourier series for identity authentication. Session key based on asymmetric quadratic polynomial. Session key information is encrypted with a private key. The group key is calculated based on the personal and session keys using the Lagrange interpolation polynomial.

The session key is generated similarly to the group key. Article [12] proposes a pre-distribution key scheme based on coding and an interaction model. The interaction model is based on Reed-Solomon code and Goppa binary code. Flexibility of the scheme is ensured by selection the coding parameters. Article [13] proposes a pre-distribution key scheme based on a polynomial pool using a sparse matrix. In articles [14-16], a new scheme for pre-distribution of hierarchical keys based on a combinatorial design for nebulous networks is proposed. Article [17] proposes the key preliminary distribution scheme for networks with placement objects in the square grid nodes. This scheme uses 3 keys per node. Memory reduction is achieved by considering the linear structure of the network. Article [18] proposes a matrix the key preliminary distribution scheme for SCADA systems. Article
[19] proposes a keying scheme for wireless networks. This approach combines Q-composite schemes with a polynomial scheme. In article [20], the keying scheme is used for end-to-end routing.

Key preliminary distribution schemes have the ability to implement additional functions. These schemes allow you to implement a security policy on distributed systems [21-24]. Key pre-distribution schemes are used for discrete signals. If analog signals are used in the system, they are sampled.

All key pre-distribution schemes require a secure channel to forward key materials. Some systems require concealing the existence of such a secure channel. Steganography methods can be used in such systems. The transmitted signals are added to the multimedia files. The files you are adding have a small amplitude. Such embedding is resistant to interference in communication channels and can be used in chaotic masking systems. The key pre-distribution scheme shall use continuous functions for such embedding.

In this article, we show that the key pre-distribution scheme can be implemented for analog signals. This result will allow the creation of secure technologies at the physical layer of data transfer.

2. Basic scheme

A generalized key pre-distribution scheme over an arbitrary vector space is proposed by the authors in the article [24]. We will summarize the main provisions of this scheme.

We view the network with n users \{u_1, u_2, ..., u_n\}. Linear space V is used for calculations. The scalar product \(V \times V \rightarrow R\) is defined in V as a binary operation. R is the set of real numbers. The unique element of the vector space \(v_i \in V\) is mapped to each user \(u_i\). Key distribution server generates elements \(\{v_1, v_2, ..., v_n\}\). The server provides elements \(v_i\) at the request of users in open form.

The server \(S\) generates a linear symmetric operator \(A\) over \(V\) (\(A: V \rightarrow V\)). The operator is kept secret on the server. This operator is used to calculate the key materials for users. The server calculates the key materials \(g_i\) for each user \(u_i\).

\[ g_i = A \cdot v_i. \]

Server \(S\) forwards key materials to each user via a secure channel.

\[ S \rightarrow u_i: g_i. \]

If the user \(u_i\) wants to establish secure communication with the user \(u_j\), then he asks the server for an appropriate number \(v_j\).

\[ u_i \rightarrow S: u_i. \]
\[ S \rightarrow u_i: v_j. \]

The user \(u_i\) calculates the encryption key.

\[ k_{ij} = v_j \cdot g_i. \]
\[ u_i \rightarrow u_j: u_i. \]

The \(u_j\) user performs the same chain of actions.

\[ u_j \rightarrow S: u_i. \]
\[ S \rightarrow u_j: v_i. \]
\[ k_{ji} = v_i \cdot g_j. \]

The keys match because of the symmetry property of operator \(A\).
\[ k_{ij} = k_{ji}. \]
\[ v_i \cdot A v_j = v_j \cdot A v_i. \]

3. Periodic Function Space

We considered the application of this scheme to finite linear spaces [24]. We are now considering the application of this scheme to differential operators over continuous space. The vector space of harmonic functions is used in the scheme.

\[ V = \{ \sin(2\pi x), \cos(2\pi x) \}. \]

We define a scalar product on space \( V \).

\[ v_1 \cdot v_2 = \int_{-1}^{1} v_1(x)v_2(x)dx. \]

The user vectors on the server are formed as the sum of the harmonic series.

\[ v_i = \sum_{n=0}^{k} h_n^{(i)} \sin(2\pi nx). \]

The coefficients \( h_n^{(i)} \in R \) are unique for each user. The number of terms \( k \) is the same for all users. The operator is selected as a linear combination of different orders derivatives.

\[ A = a_2 \frac{d^2}{dx^2} + a_1 \frac{d}{dx} + a_0. \]

The coefficients \( a_i \) are real and kept secret. The server calculates key materials for each user.

\[ g_i = A \cdot v_i = \sum_{n=0}^{k} h_n^{(i)} \left( (a_0 - 4\pi^2 n^2 a_2)\sin(2\pi nx) + a_1 2\pi n \cos(2\pi nx) \right) \]

The user \( u_i \) establishes communication with the user \( u_j \).

We write the user vector \( u_j \) in general form.

\[ v_j = \sum_{m=0}^{k} h_n^{(j)} \sin(2\pi mx). \]

The user \( u_i \) calculates the encryption key.
The user $u_i$ performs the same operations and finds the encryption key.

$$k_{ji} = v_j g_i = \sum_{n=0}^{k} \sum_{m=0}^{k} b_n^{(j)} b_m^{(i)} \left[ (a_0 - 4\pi^2 n^2 a_2) \right] \delta_{nm}.$$

The encryption key $k_{ij}$ is independent of the coefficient $a_1$. The second term in statement $A$ can be excluded. Odd order derivatives can be excluded from operator $A$.

4. Example
We consider an example of using the proposed scheme. We limit ourselves to two users $u_1$ and $u_2$. The first user corresponds to vector $v_1$.

$$v_1 = 0.10 + 0.23 \sin(2\pi x) + 0.35 \sin(4\pi x).$$

The second user corresponds to vector $v_2$.

$$v_2 = -0.27 + 0.31 \sin(2\pi x) - 0.53 \sin(4\pi x).$$

The server generates a differential operator with random coefficients.

$$A = \frac{d^2}{dx^2} - 0.64.$$

The server calculates the key materials of the user $u_1$.

$$g_1 = A \cdot v_1 = -0.064 - (0.3864\pi^2 + 0.1472) \sin(2\pi x) + (2.352\pi^2 - 0.224) \sin(4\pi x).$$

The server calculates the key materials of the user $u_2$. 
\[ g_2 = A \cdot v_2 = -0.1728 - (0.5208\pi^2 + 0.1984)\sin(2\pi x) + (3.5616\pi^2 + 0.3392)\sin(4\pi x). \]

User \( u_1 \) calculates the encryption key.

\[ k_{12} = g_2 \cdot g_1 = \int_{-1}^{1} v_2(x) g_1(x) dx = -13.39490672498369043874 \cdot 10^2. \]

User \( u_2 \) calculates the encryption key.

\[ k_{21} = g_1 \cdot g_2 = \int_{-1}^{1} v_1(x) g_2(x) dx = -13.39490672498369043874 \cdot 10^2. \]

The keys \( k_{12} \) and \( k_{21} \) match. The binary key code must be written for use in symmetric encryption algorithms.

\[ k_{12} = k_{21} = 1101.01100100011000101. \]

We limit ourselves to 16 decimal places.

\[ k = 0110010100011100. \]

5. Conclusion
We have considered the implementation of the key pre-distribution scheme over the continuous functions linear space. Key materials are formed based on differential linear operator. We have shown that odd order derivatives do not affect the encryption key value.

References

[1] Blom R 1985 EUROCRYPT’84 335
[2] Mitchell C J, Piper C 1988 Discrete and Applied Math. 21 215
[3] Du W, Deng J, Han Y S, Varshney P K 2003 Proceedings of the 10th ACM Conference on Computer and Communication Security 42
[4] Liu D, Ning P 2003 Proceedings of 10th ACM Conference on Computer and Communications Security 52
[5] Liu D, Ning P 2003 Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks 72
[6] Anjum F 2006 Proceedings of WiSe 21
[7] Shi E, Perrig A 2004 IEEE Wireless Communications 11(6) 38
[8] Liang Y, Poor H V, Shamai S 2009 Found. Trends Commun. Inf. Theory 5 355
[9] Parakh A, Kak S 2011 Fifth IEEE International Conference on Advanced Telecommunication Systems and Networks (ANTS) 1
[10] Masaeli N et al 2020 Journal of Information Security and Applications 54 102519
[11] Wang X et al 2020 Sensors 20 4388
[12] Biswas C, Dutta R 2020 J Ambient Intell Human Comput
[13] Choudhary V, Taruna S 2020 J. of Discrete Mathematical Sciences and Cryptography 23(1) 95
[14] Akhbarifar S et al 2019 Iran J Sci Technol Trans Sci 43 2399
[15] Kumar A, Pais A R 2019 J Ambient Intell Human Comput 10 2401
[16] Kumar A, Pais A R 2019 Wireless Netw 25 1185
[17] Shah K A, Jinwala D C 2019 Wireless Pers Commun 108 939
[18] Pramod T C et al 2019 Journal of Critical Infrastructure Protection 24 111
[19] Xiong C et al 2019 Journal of Physics: Conf. Seri 1229 012066
[20] Albakri A, Harn L 2019 IEEE Access 7 31615
[21] Belim S V, Belim S Yu 2020 Journal of Physics: Conf. Series 1441 012033.
[22] Belim S V, Belim S Yu 2019 Journal of Physics: Conf. Series 1210 012009
[23] Belim S V, Belim S Yu 2018 Automatic Control and Computer Sciences 52(8) 1134
[24] Belim S V, Belim S Yu 2020 Automatic Control and Computer Sciences 54(8) 860