Vector key pre-distribution scheme

S V Belim[1,2] **and S Yu Belim**[1]

[1]Omsk State Technical University, 11, Mira ave.,Omsk, 644050, Russia

[2]Siberian State Automobile and Highway University, Omsk, Russia

Abstract. In article the keys preliminary distribution scheme using calculations in the any vector space is suggested. The secret information is generated on the server as linear operator. The linear operator has to be the symmetric. The open vector is compared to each user. Key materials are calculated by the server based on the secret linear operator and open vectors. Key materials are transferred to users via a secure channel. Pair keys are calculated based on open vectors and key materials. Two vector spaces are considered. The first case is space over Zp. The linear operator is a square symmetrical matrix. Restrictions for the matrix size are received. It is shown that the Blom's keys preliminary distribution scheme is a special case of this scheme. The second case this vector space of smooth functions. Secret is defined by the linear differential operator. If to be limited to final basis, then the second case comes down to the first.

1. Introduction

Keys preliminary distribution schemes are applied to decrease the size of key materials. Key materials are transferred to users on the confidential channel and stored in protected mode. The main idea of keys preliminary distribution schemes consists in calculation the pair keys of the symmetric enciphering based on some data set. The server transfers to each user individual data (key materials) on a secure channel. The user initiates the encoded communication channel to other user. After that the user calculates an enciphering key based on open data about the second user and the key materials. The second user carries out the same operations. Key materials are created so that both users calculate identical keys. The key materials volume has to be less simple pair keys list.

The most known algorithms of keys preliminary distribution are the Blom's scheme [1] and the KDP-scheme [2]. In the Blom's scheme the key materials are calculated based on a symmetric polynomial from two variables. The polynomial is stored on the server. Polynomial coefficients from one variable are transferred to each user via a secure channel. The polynomial from one variable is calculated from the polynomial which is stored on the server. Numbers, unique for each user, are open information. Pair keys are calculated as value of a polynomial from two variables in the point determined by individual numbers of users. In the KDP scheme as key materials some set of numbers is used. Pair keys are calculated as the sum of a numbers subset from the key materials determined by open information.

Keys preliminary distribution schemes are widely used at realization of Wireless Sensor Networks (WSN) [3]. Use the keys preliminary distribution schemes in these systems is caused by the small memory size in sensors and a large number of network subscribers. Use the open keys infrastructure for sensors is impossible for practical reasons. Asymmetric enciphering demands labor-consuming calculations. This requirement significantly increases sensor cost. Keys preliminary distribution schemes use the symmetric cryptography. This approach significantly reduces requirements to memory size for enciphering keys storage.

Recently many key schemes were developed for wireless sensors networks. However, all these schemes use a uniform master key. If master key is compromised, then ciphered information transferred between all sensors is disclosure. If keys preliminary distribution scheme is used, then compromise the key materials of one sensor leads to disclosure only this sensor data. In article [4] the randomized keys distribution scheme is suggested. This scheme allows to save memory and not to store all pair keys for all sensors. Each sensor receives an accidental incomplete set of couples of keys for communication with neighbors. After that sensors available to it are defined. Paths with available intermediate clusters are used for contact with inaccessible sensors. This scheme is flexible and doesn't demand large volumes of memory. This scheme has restrictions on scalability. For this scheme several improvements were suggested. For example, in work [5] instead the pair keys the general keys for q of subscribers were used ($q > 1$). Such modification allows to reduce the volume of key materials, but reduces network safety.

In articles [6,7] it is suggested to use threshold schemes instead of random distribution of pair keys for ensuring scalability. This approach allows to secure network in general if the number of the compromised clusters are less than threshold value. For a guarantee of establishment of connection of each pair of sensors the Blom’s scheme is used.

The schemes based on splitting subscribers into groups [8-14] are suggested for increase in scalability of network. In each group the sensor responsible for contact with certain group of sensors is appointed. Such schemes have vulnerabilities that does them impractical.

In article [15] the keys preliminary distribution scheme in which key materials are coordinated to a point of arrangement the sensor is suggested. Communication between sensors is carried out through the allocated anchor clusters. Safety for such network depends on resistance to the attacks on anchor clusters.

In work [16] the scheme for session keys formation based on hashing function for initial keys is suggested. This scheme isn't steady against a compromise of an initial key. For increase the network security the scheme for change the generation of key information is suggested in articles [18,19]. This scheme demands big computing power from the keys distribution server. In articles [20-25] alternation the keys generations are carried out based on the hash functions.

In sensors networks there can be bans on communication of some subscribers. Modifications the KDP-scheme for keys preliminary distribution with the forbidden channels are considered in articles [26-28]. Modification the Blom’s scheme with the forbidden channels are described in articles [29,30].

In this article the generalized scheme of keys preliminary distribution in the any vector space is suggested.

2. General vector keys pre-distribution scheme

Let there are n users in a system $\{u_1, u_2, ..., u_n\}$. We will carry out calculations over the vector space of V. Let's define over V binary operation $V\times V\rightarrow R$. We will call this operation a product. R is set of real numbers. We will designate the product by dot. Let's compare to each user of u_i the element $v_i\in V$. A set of elements $\{v_1, v_2,..., v_n\}$ is stored on the server in open form.

The linear operator A over V is generated on the server for formation the key materials ($A{:}V\rightarrow V$). The linear operator is stored on the server in a secret. Key materials for the user of u_i are calculated on a formula:

$$g_i=A\cdot v_i.$$

The user u_i takes on the server the value v_j for calculation the key for symmetric encryption with the user u_j. The key is calculated on a formula

$$k_{ij}=v_j\cdot g_i.$$

The user u_j carries out similar operation and calculates the key

$$k_{ji}=v_i\cdot g_j.$$

Equality of keys has to be carried out for ensuring information exchange

$$k_{ij}=k_{ji}.$$

or

$$v_i\cdot A\cdot v_j=v_j\cdot A\cdot v_i.$$

The linear operator A has to be the symmetric.

Let's consider implementation this scheme for a concrete vector space V.

3. Implementation the scheme in space Z_p^k

Let's consider the k-dimensional vector space case over the ring Z_p ($V=Z_p^k$). In this case each element of the vector space of V represents a k-dimensional vector $v_i=(v_1^{(i)},...,v_k^{(i)})$. The linear operator A can be written down as a matrix by the $n\times n$ size over Z_p^k. The server generates an accidental symmetrical matrix A for formation the key materials. For each user vectors are calculated

$$g_i=A\cdot v_i.$$

Vectors g_i are sent via a secure channel. The vector g_i contains k values. For storage all pair keys with all subscribers of network it is necessary to mark out memory for n values. The keys preliminary distribution scheme can be used only when performing the condition $k<n$. Pair keys for the symmetric enciphering are calculated based on formula:

$$k_{ij}=v_j^{\mathrm{T}}\cdot g_i= v_j^{T}\cdot A\cdot v_i.$$

Let's consider a scheme compromise question. If key k_{ij} is known to the malefactor, then he can write the equation:

$$k_{ij} = \sum_{l=1}^{k}\sum_{r=1}^{k} v_{jr} a_{rl} v_{il}.$$

In this equation unknowns are matrix elements a_{rl}. Total number of independent matrix elements is $k(k+1)/2$. The compromise of all scheme requires disclosure all matrix elements. It is necessary to make a system which contain $k(k+1)/2$ equation. For this purpose, it is necessary to know $k(k+1)/2$ pair keys.

Each user can calculate n keys. Any legal user can't read messages of other users if inequality is carried out:

$$n < k(k+1)/2.$$

From here one more restriction exist for matrix A size:

$$k > (\sqrt{8n+1}-1)/2.$$

For example, at creation the network of 1000 subscribers the matrix sizes need to be chosen in the limits:

$$45<k<1000.$$

Widely known Blom’s preliminary distribution keys scheme is a special case of the submitted vector scheme over Z_p^k. For implementation the Blom’s scheme it is necessary to choose v_i vectors:

$$v_i=(1,q^i_1,q^i_2,...,q^i_{k-1}).$$

The q_i are any numbers in Z_p^k.

4. Continuous realization

Let's consider a vector space of smooth functions V. On the server the linear differential operator A is generated for formation of key materials. The smooth functions $v_i(x)$ are elements of a vector space. The dot product is calculated on a formula

$$v_i \cdot v_j = \int_a^b v_i(x) v_j(x) dx.$$

Limits of the integration a and b are open constants. Vectors $g_i(x)$ are sent to users.

$$g_i(x)=Av_i(x).$$

The pair key of enciphering is calculated on the formula

$$k_{ij} = v_i \cdot g_j = v_i \cdot Av_j = \int_a^b v_i(x) Av_j(x) dx.$$

If the vector space V has final basis $\{f_i(x)\}$ (i=1,...,k), this case comes down to previous. Let's spread out vectors v_i and g_i on basis

$$v_i = \sum_{j=1}^{k} a_j^i f_i(x).$$

$$g_i = \sum_{j=1}^{k} b_j^i f_i(x).$$

For these vectors we receive representations

$$v_i=(a_1^i,a_2^i,...,a_k^i),$$

$$g_i=(b_1^i,b_2^i,...,b_k^i).$$

For the linear operator A we can gain representation by matrices

$$a_{ij} = f_i \cdot Af_j = \int_a^b f_i(x)Af_j(x)dx.$$

Further the scheme looks also as in the previous section.

5. Conclusion

The vector keys preliminary distribution scheme suggested in article is the general approach for creation the specialized schemes. The choice the vector spaces allows to create the realization with the given requirements. The choice a vector space is defined by the extent of available memory in the device and computing power. The secret of the scheme is defined by the choice the linear operator. The security of the scheme to compromise is provided with the choice the number of parameters in the operator. At implementation the scheme in real devices it is necessary to solve a problem of optimization the calculations speed and memory.

References

[1] Blom R 1985 *Lecture Notes in Computer Science* **209** 335.

[2] Mitchell C J and Piper C 1988 *Discrete and Applied Math.* **21** 215.

[3] Divya R and Thirumurugan T 2011 *International Journal of Scientific & Engineering Research*. **2** 5 108.

[4] Eschenauer L and Gligor V D 2002 *Proceedings of the 9th ACM conference on Computer and Communication Security* 41.

[5] Chan H, Perrig A and Song D 2003 *Proceedings of IEEE Symposium on Security and Privacy* 197.

[6] Du W, Deng J, Han Y S and Varshney P K 2003 *Proceedings of the 10th ACM Conference on Computer and Communication Security* 42.

[7] Liu D and Ning P 2003 *Proceedings of 10th ACM Conference on Computer and Communications Security* 52.

[8] Liu D and Ning P 2003 *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks* 72.

[9] Huang D, Mehta M, Medhi D and Harn L 2004 *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks* 29.

[10] Du W, Deng J, Han Y S., Chen S and Varshney P K 2004 *Proceedings of IEEE INFOCOM* 586.

[11] Liu D and Ning P 2005 *ACM Transactions on Sensor Networks* **1** 204.

[12] Zhou Y, Zhang Y and Fang Y 2005 *Proceedings of IEEE Wireless Communications and Networking Conference* 29.

[13] Yu Z and Guan Y 2005 *Proceedings of IEEE Wireless Communications and Networking Conference* 294.

[14] Du W, Deng J, Han, Y S and Varshney P K 2006 *IEEE Transactions on Dependable and Secure Computing* **3** 62.

[15] Anjum F 2006 *WiSe '06 Proceedings of the 5th ACM workshop on Wireless security* 21.

[16] Hussain S, Rahman M and Yang L 2009 *IEEE Computer Society* 1.

[17] Wangsheng F, Tao Z and Kang C 2009 *4th International Conference on Computer Science & Education. ICCSE '09* 321.

[18] Castelluccia C and Spognardi A 2007 Proceedings of the Third International Conference on Security and Privacy in Communications Networks 351.

[19] Ergun M, Levi A and Savas E 2009 *Proceedings of the 24th International Symposium on Computer and Information Sciences* 375.

[20] Shan T and Liu C 2008 *IEEE Asia-Pacific Conference on Services Computing* 1127.

[21] Su Z, Lin C, Ren F, Jiang Y and Chu X 2009 *WRI International Conference on Communications and Mobile Computing* **3** 333.

[22] Liu M, Wei W and Liu Z 2009 *4th IEEE Conference on Industrial Electronics and Applications* 1762.

[23] Zhang K and Wang C 2010 *International Conference on Computer Design and Applications (ICCDA)* **2** V2-626.

[24] Zhang T and Qu H 2010 *Second International Workshop on Education Technology and Computer Science (ETCS)* **1** 272.

[25] Kesavan V T and Radhakrishnan S 2012 *International Journal of Communication Networks and Information Security (IJCNIS)* **4** 1 68.

[25] Zhu S, Xu S, Setia S and Jajodia S 2003 *Proceedings of the 11th IEEE International Conference on Network Protocols* 326.

[26] Belim S V and Belim S Yu 2016 *Automatic Control and Computer Sciences* **50** 8 773.

[27] Belim S V and Belim S Yu 2019 *Journal of Physics: Conf. Series* **1210** 012009.

[28] Belim S V and Belim S Yu 2018 *Automatic Control and Computer Sciences* **52** 8 1124.

[29] Belim S V and Belim S Yu 2018 *Automatic Control and Computer Sciences* **52** 8 1134.

[30] Belim S V and Belim S Yu 2019 *Journal of Physics: Conf. Series* **1210** 012008.