Anonymous Multi-Receiver Public Key Encryption Based on Lucas Sequences

L houssain El Fadil [1] , Abdelkader MOUMEN [2] and Mohamed BOUYE [3]

[1] *Department of Mathematics, King Khalid University.*
Faculty of Sciences, B.P. 1000, Abha, Kingdom of Saudi Arabia.
[1] *Polydisciplinary Faculty of Ouarzazat*
P.O. Box 638, Ouarzazat- Morocco.
lhouelfadil2@gmail.com

[2] *Department of Mathematics, King Khalid University.*
Faculty of Sciences, B.P. 1000, Abha, Kingdom of Saudi Arabia.
abdelkader.moumen@gmail.com

[3] *Department of Mathematics, King Khalid University.*
Faculty of Sciences, B.P. 1000, Abha, Kingdom of Saudi Arabia.
medeni.doc@gmail.com

Abstract

Multi-receiver encryption enables a sender to encrypt a message and transmit the ciphertext to a set of authorized users, while no one out this group of authorized users can decrypt the message. Multi-receiver encryption is of great importance in many sectors such as broadcast communication, cloud computing, wireless communications, networking applications, e-voting, lottery, and medical applications. This paper proposes an efficient multi-receiver public key encryption scheme based on Lucas sequences. The results of the computational analysis show that, projected scheme is better against renown attacks and prevailing anonymous multi-receiver algorithms.

Keywords: Lucas sequences, public key cryptosystems, Multi-receiver encryption scheme, Chinese Remainder Theorem, Anonymity.

1 Introduction

The goal of the multi-receiver encryption is to securely transmit any message to all authorized receivers via insecure channels. There are several proposed multi-receiver encryption schemes in the literature based on different technical [18, 7, 1, 14, 19, 21, 4, 9, 11].

In their paper Smith and Lennon [17], were the first to introduce linear sequences in cryptography in 1993. They suggested that the second order linear sequences could be used instead of the standard exponentiation used in RSA. Two years later, Chi-Sung et al has shown that the security

 doi:10.20944/preprints201810.0751.v1

of Lucas functions is polynomial-time equivalent to the generalized discrete logarithm problems [2]. In 2012, El Fadil proposed a cryptosystem based on second order linear sequences in which semantic security is ensured [10]. Moreover, as linear sequences are not multiplicative, the main advantage of Lucas cryptosystems is that they are not formulated in terms of exponentiation. This would make them invulnerable to various well known attacks that threaten the security of more traditional cryptosystems like RSA and Diffie Hellman [5]. However, many current networking applications involve multiple receivers and one-to-one type of encryption is not appropriate for such type of applications. The main motivation of multiple receivers is to secure that group communications which getting popularity in different domains.

In this paper, we propose a novel multi-receiver public key encryption scheme based on Lucas sequences. From analysis results, it is clear that our approach is more efficient and satisfying anonymity of receivers as compared to existing schemes. Section 2, consists of brief description of the second order linear sequences ("Lucas sequences") and their cryptography properties. After that, we present a new approach in Section 3 with security and computation analysis. Finally, we give our conclusion in Section 4.

2 Second order linear sequences ("Lucas sequences")

In this section, the main cryptographic properties of Lucas sequences are studied. A computational method to evaluate the k^{th} term are given, together with an analysis of its computational cost.

Throughout this section :

- p is a prime integer.
- $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ the finite field of p elements.
- $(a, b) \in \mathbb{Z}^2$ such that $b \equiv 1 \pmod{p}$.
- $f(X) = X^2 - aX + b$ is primitive polynomial over $\mathbb{F}_p[X]$.
- $A = \mathbb{F}_p[X]/(f(X))$.
- $\alpha = \bar{X}$ the class of X modulo the principal ideal of $\mathbb{F}_p[X]$ generated by $f(X)$.
- For every $x \in A$, l_x be the linear map of A defined by $l_x(y) = xy$.
- $T(x) = Tr(l_x)$ the trace of x, $Tr(l_x)$ the trace of the linear map l_x.
- $N(x) = det(l_x)$ the norm of x, $det(l_x)$ is the determinant of the linear map l_x.
- Define the sequence $s(a)$ as follows :

$$k \in \mathbb{Z},\ s_k(a) = T(\alpha^k) \text{ for every integer.} \tag{1}$$

Remark 2.1 *Since $f(\alpha) = 0$ and the map trace is linear, it follows that :*

$$s_{k+2}(a) \equiv a s_{k+1}(a) - s_k(a) \pmod{p} \text{ modulo } p. \tag{2}$$

So, $s(a)$ is a second order linear sequence (Lucas sequence), called the characteristic sequence generated by a.

Remark 2.2 *Let l_k be the endomorphism of A defined by :*

$$\begin{array}{rccc} l_k(x) : & A & \longrightarrow & A \\ & x & \longmapsto & l_k(x) = \alpha^k x \end{array}$$

M_k its matrix with respect to the basis $(1, \alpha)$.

Then $M_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ *and* $M_1 = \begin{pmatrix} 0 & -1 \\ 1 & \bar{a} \end{pmatrix}$.

Thus, $s_0(a) \equiv 2 \pmod{p}$ and $s_1(a) \equiv a \pmod{p}$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2018 doi:10.20944/preprints201810.0751.v1

2.1 Cryptographic properties

The cryptographic applications of Lucas sequences are listed in [16, 15]. To make this concept easy for the reader, we present some of these results in a more accessible form and without proofs.

1. For every integer k, $s_k(a) \equiv \alpha^k + \alpha^{-k} \pmod p$ and $s_k(a) \equiv s_{-k}(a) \pmod p$.
2. For every integers k and e, $s_e(s_k(a)) \equiv s_{ke}(a) \pmod p$.
3. Let $a \in \mathbb{Z}$ such that p does not divide a^2-4.
   If $f(X)$ is a primitive polynomial over p, then $\pi = p+1$ is the period of $s(a)$. More general if $\left(\frac{a^2-4}{p}\right) = -1$, then $f(X)$ is irreducible. So $N(\alpha) = \alpha^{1+p} = 1$. Thus the period π of $s(a)$ divides $p+1$. If $\left(\frac{a^2-4}{p}\right) = 1$, then $f(X)$ is reducible. The fact that $A \simeq \mathbb{F}_p \times \mathbb{F}_p$ implies that the period π of $s(a)$ is $p-1$, where $\left(\frac{-}{p}\right)$ is the Legendre symbol.

For the two first points, if $f(X)$ is irreducible in $\mathbb{F}_p[X]$, then $N(\alpha) = \alpha^{p+1} \equiv 1 \pmod p$.
Therefore :

$$\begin{array}{rcl} s_{p+1}(a) & \equiv & \alpha^{p+1} + \alpha^{-(p+1)} \\ & \equiv & N(\alpha) + N(\alpha^{-1}) \\ & = & 2 \\ & = & s_0(a) \pmod p \text{ and } \pi \text{ divides } p+1. \end{array}$$

If $f(X)$ is a primitive polynomial modulo p, then α generates the multiplicative group $(A^*, .)$ with cardinal order p^2-1 and π is exactly $\pi = p+1$.

Corollary 2.3 *For every integer e such that $gcd(e, \pi) = 1$, where the gcd is the greatest common divisor, the map :*

$$\begin{array}{rccc} Luc_e : & \mathbb{F}_p & \longrightarrow & \mathbb{F}_p \\ & a & \longmapsto & s_e(a) \end{array}$$

is a one-one correspondence.

Proof: Indeed, since $gcd(e, \pi) = 1$, then we can use the Euclid algorithm to calculate the inverse d of e modulo π. That means there exists an integer k such that $de = 1 + k\pi$. Hence :

$$\begin{array}{rcl} s_d(s_e(a)) & \equiv & s_{de}(a) \\ & \equiv & s_{1+k\pi}(a) \\ & \equiv & s_1(a) \\ & \equiv & a \pmod p. \end{array}$$

□

Lemma 2.4 *Let $e \in \mathbb{N}$ such that $gcd(e, \pi) = 1$ and let $c = s_e(a)$. Then p does not divide c^2-4 if and only if p does not divide a^2-4 and $\left(\frac{a^2-4}{p}\right) = \left(\frac{c^2-4}{p}\right)$.*

Proof: Since $a \equiv s_d(c) \pmod p$, where d is the inverse of e modulo π, it suffices to show that if $\left(\frac{a^2-4}{p}\right) = 1$, then $\left(\frac{c^2-4}{p}\right) = 1$ too.
So, assume that $\left(\frac{a^2-4}{p}\right) = 1$, then $\alpha \in \mathbb{F}_p$. Thus, $\alpha^e \in \mathbb{F}_p$ and $f_c(X) = X^2 - cX + 1$ splits in $\mathbb{F}_p[X]$; $f_c(X) \equiv (X-\alpha^e)(X-\alpha^{-e}) \pmod p$. Therefore, i.e., $\left(\frac{c^2-4}{p}\right) = 1$. □

2.2 Computation Method and analysis

Lemma 2.5 *For every integer n, set $s_n := s_n(a) \pmod p$. Then*

$$\left\{ \begin{array}{llcl} i) & s_{2n} & \equiv & s_n^2 - 2, \\ ii) & s_{2n+1} & \equiv & s_n s_{n+1} - a \end{array} \right.$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2018 doi:10.20944/preprints201810.0751.v1

Proof: Let n and m be two integers.

$$\begin{aligned} s_n s_m &\equiv (\alpha^n + \alpha^{-n})(\alpha^m + \alpha^{-m}) \pmod p \\ &\equiv (\alpha^{n+m} + \alpha^{-n-m}) + (\alpha^{n-m} + \alpha_2^{-n+m}) \\ &\equiv s_{n+m} + s_{n-m} \pmod p. \end{aligned}$$

Therefore,

$s_{n+m} \equiv s_n s_m - s_{n-m} \pmod p$.

In particular, we have i) and ii). □

Let $k = 2^r m$, where m is an odd integer. To compute s_k, first we compute s_m, then s_{2m} :

$$\begin{aligned} s_{2m} &\equiv s_m^2 - 2 \\ s_{4m} &\equiv s_{2m}^2 - 2 \\ &\cdots\cdots \\ s_k &\equiv s_{2^{r-1}m}^2 - 2. \end{aligned}$$

Then to compute s_k, we need r multiplications modulo p and we need s_m.

Let $m = \sum_{i=0}^{l-1} k_i 2^{l-1-i}$. For every $0 \leq i < l-1$, let $f_{i+1} = 2f_i + k_{i+1}$ and $f_0 = k_0$. Then $f_{l-1} = k$. For $0 \leq i < l-1$ and assume that, $s_{f_{i-1}}$ and $s_{f_{i-1}+1}$ are computed. Then

$$\text{if } k_i = 0, \text{ then } \begin{cases} s_{f_i} \equiv s_{2f_{i-1}} s_{f_{i-1}}^2 - 2 \\ s_{f_i+1} \equiv s_{2f_{i-1}+1} \equiv s_{f_{i-1}}(a) s_{f_{i-1}+1} - a \end{cases}$$

$$\text{if } k_i = 1, \text{ then } \begin{cases} s_{f_i} \equiv s_{2f_{i-1}+1} \equiv s_{f_{i-1}} s_{f_{i-1}+1} - a \\ s_{f_i+1} \equiv s_{2(f_{i-1}+1)} \equiv s_{f_{i-1}+1}^2 - 2 \end{cases}$$

Computation Algorithm. This method warrant that s_k can be computed in about the same length of time as the k^{th} power is computed in the RSA method. But in the computation of s_m, having to compute *two numbers at each stage does slow* the computation down a little, but there are optimizations in the calculation which mean that the total amount of computation is only about *half less* than the amount needed for the RSA system. Therefore, to compute $s_k(a)$, the total number of multiplications modulo p is $log_2(k)$.

3 Main results

In this section we describe some applications of Lucas sequences, in more details : Lucas multi-receiver encryption scheme and Lucas Diffie-Hellman key exchange extended to a group of communications. Through this section, $n = pq$ is an RSA integer, $a \in \mathbb{Z}$ sush that $gcd(n, a) = 1$, $(s_n)_n$ is the Lucas sequence generated by $f = x^2 - ax + 1$, and f is a primitive polynomial over $\mathbb{F}_p$ and $\mathbb{F}_q$.

3.1 Multi-receiver encryption scheme

In this section, we propose a multi-receiver encryption scheme based on Lucas sequences and Chines Remainder Theorem. Our scheme consists of two procedures: Encryption procedure and decryption procedure.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2018 doi:10.20944/preprints201810.0751.v1

Algorithm 1: Computation Algorithm.

1. Input: $k = 2^r \sum_{i=0}^{l-1} k_i 2^i$ and a, where $k_0 \neq 0$ and $k_{l-1} \neq 0$.
2. Output: s_k.
3. $s_0 = 2$, $s_1 = a$.
4. for i from 0 to $l-1$ do
5. if $k_i = 0$ then $s_1 = s_1 s_0 - a$, $s_0 = s_0^2 - 2$
6. else then $s_0 = s_1 s_0 - a$, $s_1 = s_1^2 - 2$
7. End if
8. End for.
9. Return (s_0).
10. $s = s_0$.
11. for i from 1 to r do
12. $s = s^2 - 2$.
13. End for.
14. Return (s).

1. **Keys generation :** First, assume that there are $r+1$ users. Any user selects $s \leq r$ authorized receivers $u_1, \cdots, u_s$ such that each one has one public key pair (n_i, e_i), where every $n_i = p_i q_i$ is an RSA integer, e_i is coprime to $(p_i \mp 1)(q_i \mp 1)$ and $n_1, \cdots, n_r, n_{r+1}$ are pairwise coprime.
2. **Encryption procedure :** A sender uses all s receivers public keys to encrypt the message and then broadcasts the ciphertext to the s authorized receivers as follows :
   (a) Assume that he needs to send a message $0 < m < N$ to the s legitimate receivers, where N is the minimum of all n_i.
   (b) Then for every $i = 1, \ldots, s$, he calculates $c_i = s_{e_i}(m)$. As $n_1, \cdots, n_s$ are public and pairwise coprime.
   (c) He uses the Chines Remainder Theorem to calculate the integer c such that for every $i = 1, \ldots, s$, $c \equiv c_i \pmod{n_i}$.
   (d) Then He broadcasts the ciphertext c to the s receivers.
3. **Decryption procedure :** When a autorized receiver i receives the ciphertext c, he will compute his private key d_i as follows :
   (a) First he calculates $c_i \equiv c \pmod{n_i}$, $\epsilon_{p_i} = \left(\frac{c_i^2-4}{p_i}\right)$ and $\epsilon_{q_i} = \left(\frac{c_i^2-4}{q_i}\right)$.
   (b) He calculates his private key d_i :

$$\text{the inverse of } e_i \text{ modulo } \pi = (p_i - \epsilon_{p_i})(q_i - \epsilon_{q_i}).$$

   (c) Finally, he calculates $m \equiv s_{d_i}(c_i) \pmod{n_i}$.

Proof :

Encryption :

Indeed, in order to calculate the ciphertext c, for every $i = 1, \cdots, s$, the sender will calculate

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2018 doi:10.20944/preprints201810.0751.v1

$M_i = \prod_{j=1, j\neq i}^{s} n_j$, y_i its inverse modulo n_i and $c = \sum_{i=1}^{s} c_i y_i M_i$.
As $y_i M_i \equiv 1 \pmod{n_i}$ and for every $j \neq i$, $y_i M_i \equiv 0 \pmod{n_j}$ (since n_j divides M_i), for every $i = 1, \ldots, s$, $c \equiv c_i \pmod{n_i}$.

Decryption :

For the decryption phase, as $c_i \equiv s_{e_i}(m) \pmod{n_i}$ and $gcd(e, \pi) = 1$, $\left(\frac{c_i^2-4}{p}\right) = \left(\frac{m^2-4}{p}\right)$ and $\left(\frac{c_i^2-4}{q}\right) = \left(\frac{m^2-4}{q}\right)$.
Thus,

$$\begin{aligned} s_{d_i}(c_i) &\equiv s_{d_i}(s_{e_i}(m) \\ &\equiv s_{d_i e_i}(m) \\ &\equiv m \pmod{n_i}. \end{aligned}$$

Analysis :

Now, we will discuss security, computation cost and the anonymity of receivers of the proposed scheme. The anonymity means that the identities of receivers can be protected. We will compare the computation cost of our scheme and other existing schemes.

3.1.1 Security

Since this scheme uses "LUC" cryptosystem, the security of the scheme is the same as mono-cast "LUC" cryptosystem. Namely, if Oscar wants to decrypt the encrypted message c, first of all he needs to break the LUC cryptosystem, which is cryptographically equivalent to the security of RSA [17, 2].

3.1.2 Anonymity of Receivers

When a sender broadcasts the ciphertext nobody in the group except the sender knows who is the receiving end.

3.2 Diffie-Hellmane Key exchange extended to a group of communication

The standard Diffie-Hellman key exchange was first proposed in 1976 [20]. After that there have been efforts to extend its simplicity to a group setting. The main motivating factor is the increasing popularity of various types of group applications and the need of doing it securely. Since key distribution is the cornerstone of secure group communication, it has naturally received a lot of attention. Notable solutions have been proposed but some of theme are only of theoretical interest, while the security of some others remains unproven (See for example [12, 8, 3, 13]). In this paper, a natural extension of Diffie-Hellman key exchange, based on Lucas sequences is proposed, see Figure 1.

Let $n = pq$ be an RSA integer, a be an integer such that $X^2 - aX + 1$ is a primitive polynomial modulo p and modulo q too.

Assume that r users $u_1, \cdots, u_r$, whose have access to a public key data (n, a) and each one has his private key x_i, want to agree on a shared secret key. Then :

— The first user u_1 makes public $y_1^{(1)} = s_{x_1}(a)$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2018 doi:10.20944/preprints201810.0751.v1

— The second one u_2, makes public $(y_1^{(2)}, y_2^{(2)}, y_3^{(2)})$, where :

$$\begin{aligned} y_1^{(2)} &= s_{x_2}(a), \\ y_2^{(2)} &= s_{x_1}(a), \\ y_3^{(2)} &= s_{x_2}(y_1). \end{aligned} \tag{3}$$

The notation y_i^j means that this quantity is published by the j^{th} user.

— The third makes public $(y_1^{(3)}, y_2^{(3)}, y_3^{(3)}, y_4^{(3)})$, where :

$$\begin{aligned} y_1^{(3)} &= s_{x_3}(y_1^2), \\ y_2^{(3)} &= s_{x_3}(y_2^2), \\ y_3^{(3)} &= y_3^2, \\ y_4^{(3)} &= s_{x_3}(y_3^2). \end{aligned} \tag{4}$$

— Once the i^{th} user makes public the tuple $(y_1^{(i)}, \cdots, y_{i+1}^{(i)})$.
The $(i+1)^{th}$ makes public $(y_1^{(i+1)}, \cdots, y_{i+2}^{(i+1)})$, where for every $j = 1, \cdots i$, :

$$\begin{aligned} y_j^{(i+1)} &= s_{x_{i+1}}(y_j^{(i)}), \\ y_{i+1}^{(i+1)} &= y_{i+1}^{(i)}, \\ y_{i+2}^{(i+1)} &= s_{x_{i+1}}(y_{i+1}^{(i)}). \end{aligned} \tag{5}$$

— Finally, the last user u_n makes public $(y_1^{(n)}, \cdots, y_n^{(n)})$ and keeps secret the common private key $y = y_{n+1}^{(n)} = s_{x_i}(y_i^{(n)})$.

— Each user u_i, $i = 1, \cdots n-1$ can calculates the same value $y = s_{x_i}(y_i^{(n)})$.

The common private key is $y = y_{n+1}^{(n)}$. Indeed,

$$\begin{aligned} y &= s_{x_n}(y_n^{(n-1)}) \\ &= s_{x_n}(s_{x_{n-1}}(...(s_{x_1}(a))...)) \\ &= s_{x_n x_{n-1} .... x_1}(a). \end{aligned}$$

Computation analysis:

We compare the performance monocast "LUC" encryption and L. Harn et al. scheme [11] with the presented scheme. From Table 1 one can see that proposed scheme is more computationally efficient than the scheme proposed in [11], see Table 1.

4 Conclusion

In this paper, we introduced a new multi-receiver encryption scheme which provides the anonymity of receivers. Our scheme is more efficient than prevailing schemes in terms of computational cost. The security of this scheme is of equal strength as monocast LUC encryption. The comparison with others results and the security analyses clearly show that the proposed method provides anonymity of receivers and a good computational cost. Each receiver only needs to keep one private key and uses this private key to decrypt the ciphertext.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through General Research Project under grant number (G.R.P-86-38).

References

[1] A. Fiat and M. Naor, "Broadcast encryption," in Proceeding of Advances Cryptology - Crypto, LNCS 839, pp. 480-491, Springer-Verlag, California, USA, Aug. 1994.

[2] Chi-Sung Laih, Fu-Kuan Tu, Wen-Chun Tai, On the security of the Lucas function, Information Processing Letters 53(1995), pp 243-247.

[3] Chin Chen Chang, Tzong Chen Wu, and C.P. Chen. The Design Of A Conference Key Distribution System. In Advances in Cryptology -AUSCRYPT'92, Lecture Notes in Computer Science, pages 467-474. Springer- Verlag, Berlin Germany, December 1992.

[4] C. I. Fan, L. Huang, and P. Ho, "Anonymous multi-receiver identity-based encryption," IEEE Transations on Computers, vol. 59, no. 9, pp. 1239-1249, Sep. 2010.

[5] D. Bleichenbacherl, W. Bosma, and K. Lenstra, Some Remarks on Lucas-Based Cryptosystems, Advances in Cryptology - CRYPT0 '95, LNCS 963, pp. 386-396, 1995

[6] Douglas R. Stinson, Cryptography Theory and Practice, Third edition 2006, Chapman, Hall/CRC, Taylor and Francis Group.

[7] G. H. Chiou and W. T. Chen, "Secure broadcasting using the secure lock" IEEE Transations on Software Engineering, vol. 15, no. 8, pp. 929-934, Aug. 1989.

[8] Hugh Harney, Carl Muckenhirn, and Thomas Rivers. Group Key Management Protocol (GKMP) Architecture. INTERNET-DRAFT, September 1994.

[9] H. Ghodosi, J. Pieprzyk, and R. Safavi-Naini, "Dynamic threshold cryptosystems: a new scheme in group oriented cryptography," in Proceeding of Pragocrypt'96: the 1st International conference on the Theory and Applications of Cryptology, pp. 370-379, Prague, Czech Republic, Sep. 1996.

[10] L. El Fadil, A Public-Key Cryptosystem Based on Lucas Sequences, Palestine Journal of Mathematics Vol. 1(2) (2012) 148-152.

[11] L. HARN, C.- C. CHANG, and H.-L. WU, "An Anonymous Multi-Receiver Encryption Based on RSA", Journal of Network Security, Vol.15, No.4, PP.307-312, July 2013.

[12] M. Steiner, G. Tsudik and M. Waidner, Diffie-Hellman Key Disrtribution Extended to Group Communication, Proceding of the 3rd ACM conference on computer and communications security, March 14-16 (1996), new Delhi.

[13] M. Burmester and Y. Desmedt. A Secure And Efficient Conference Key Distribution System. In I.B. Damgard, editor, Advances in Cryptology, EUROCRYPT'94, Lecture Notes in Computer Science. Springer-Verlag, Berlin Germany, 1994.

[14] M. Bellare, A. Boldyreva, and S. Micali, "Public-key encryption in a multi-user setting: security proofs and improvements," in Proceeding of Advances Cryptology - Eurocrypt 2000, LNCS 1807, pp.259-274, Springer-Verlag, Bruges, Belgium, May 2000.

[15] P. Ribenboim, *The Little Book of Big Primes,* Springer-Verlag, 1991.

[16] P. Smith. *LUC Public Key System : A secure Alternative to RSA.* Dr. Dobb's Journal, January 1993.

[17] P. Smith and M. J. J. Lennon, LUC : A new public key system. In Proc. of the Ninth IFIP Int. Symp. on Computer Security, p. 103-117, 1993.

 doi:10.20944/preprints201810.0751.v1

[18] S. Berkovits, "How to broadcast a secret," in Proceeding of Advances Cryptology - Eurocrypt'91, LNCS 547, pp. 535-541, Springer-Verlag, Brighton, UK, Apr.1991.

[19] S. Berkovits, "How to broadcast a secret," in Proceeding of Advances Cryptology - Eurocrypt'91, LNCS 547, pp. 535-541, Springer-Verlag, Brighton, UK, Apr.1991.

[20] W. Diffie and M. E. Hellman, New directions in cryptography, IEEE Trans. Inform. Theory, vol. IT-22, (1976), pp. 644-654.

[21] X. Du, Y. Wang, J. Ge, and Y. M. Wang, "An ID-based broadcast encryption scheme for key distribution," IEEE Transations Broadcast, vol. 51, no. 2, pp. 264-266, June 2005.

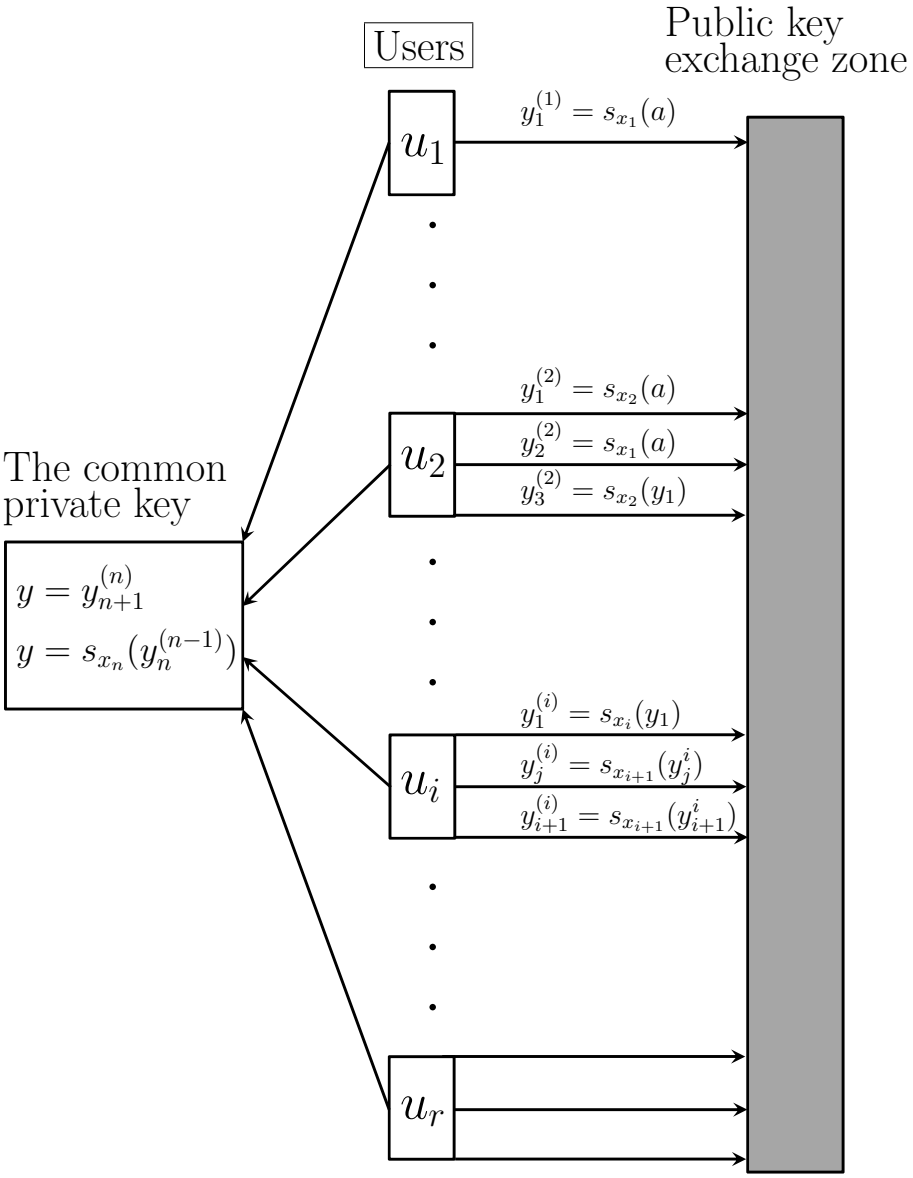

Figure 1: Diffie-Hellmane Key exchange extended to a group of communication.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2018 doi:10.20944/preprints201810.0751.v1

	Monocast LUC encryption	L. Harn, et al. scheme	Our scheme
Sender	$n/2 \times T_{E\_1024}$	$n \times T_{E\_1024}$	$n/2 \times T_{E\_1024}$
Receiver	T_D	T_D	T_D
Rounds of transmission	n	1	1
Anonymity of receivers	No	Yes	Yes

Table 1: Comparison results among different schemes.