Ciência e Natura
ISSN: 0100-8307
cienciaenaturarevista@gmail.com
Universidade Federal de Santa Maria
Brasil

Ciência e Natura, v. 37 Part 2 jun. 2015, p. 117−124

ISSN impressa: 0100-8307 ISSN on-line: 2179-460X

A New Secret Sharing Scheme With Priority in Order of Sharing and its Application in Multi Authority E-voting Systems

Seyyed Hamed Mousavi[1], Saeed Rahimi[2], Mousa Javidi Alsaadi[3], Foroogh Mousavi[4]

[1] Department of communication engineering, Shiraz university of technology, Shiraz, Iran.
[2] Department of Information Technology, Imam Hossein university, Tehran, Iran.
[3] Department of Motor Behavior, Shiraz university, Shiraz, Iran.
[4] Department of computer engineering, Fasa university, Fasa, Iran.

Abstract

Secret sharing caused a high level of security in encrypted systems. So, there are wide ranges of methods based on the secret sharing policies. Secret sharing schemes has 2 main aims. The first is determined to decrease the risks of attacks by adversaries which can be done by increasing the number of authorities. Second is to remove the dependence of protocol to an special part.

In this paper, the priority of parties to share the secret is important. Also different authorities may be given different type of part. We also propose some voting systems in order to justify suggested secret sharing protocol. Also we analyze theses protocols to show that this secret sharing protocol saves the security of E-voting system.

Keywords: *Priority in secret sharing, Multi-Authority, E-voting system, Elliptic curve, Composition of functions.*

Recebido: dia/mês/ano Aceito: dia/mês/ano

1 Introduction

Recently, secret sharing has been an active area in different cryptography systems. This is mainly because of the fact that it can make the security protocols more robust. So, there is an endless list of methods based on the secret sharing policies. Secret sharing has 2 main aims. First aim is to decrease the risks of attacks by an exterior person (like adversaries). The second one is to prevent system from depending just on one part. Advances in different schemes are rooted from two main ideas. The first one can be regarded as using different mathematical tools in order to find better ways to share a secret. Some example of mathematical tools can be interpolation of polynomials like in [Shamir, 1979], intersection of hyperplane like in [Blakley,1978], graph based schemes like in [Blundo,1995], using group theory like in [Liu,1998], designing by quantum cryptosystems like in [Cleve, 1999], [Hillery,1999]. Also [Beimel, 2011] is a good survey of different secret sharing methods. The second idea is to introduce some new applications which are suitable for different scenario. These applications results in some facilities. For example, Ingmarsen et all. in [Ingemarsson,1991] proposed a scenario in secret sharing without depending on a trusted center. Also there are some papers to investigate different conditions among authorities to have a fair sharing process. For example Tian et al. proposed a fair threshold secret sharing scheme in [Tian, 2013]. There are some works which are aimed at image secret sharing like in [Wang, 2007] and [Shyong, 2014]. There are some works which updates shares of members based on their interactions which is known as social secret sharing. For example in [Nojoumian, 2010] proposed an scheme to change the parties without changing the secret based on participants' cooperation.

One of the active application of secret sharing protocols in cryptosystems is E-voting systems. E-voting systems with multi-authority need secret sharing protocol to be more robust. Also it is better to not implement a trusted center. Some examples of multi-authority can be found in [Cramer, 1997], [Porkodi, 2011] and [Fouard, 2007]. Also [Weber, 2006] provided some typical ideas to illustrate the role of secret sharing in E-voting systems.

In this paper, we propose a new facility in secret sharing. This is rooted from a problem which arises in previous secret sharing protocols. The main contribution of this paper is to introduce a new facility of secret sharing in E-voting systems. In particular, we set priority in order of sharing parties by using composition of functions.

The rest of this paper is organized as follows. Section 2 is devoted to a brief review of some secret sharing schemes. We review a couple of E-voting schemes which are handy for us in this paper in section 3. Section 4 discussed two problems which have remained for this paper to solve. We propose a method of sharing to solve these problems in this section, too. In section 5, we design an E-voting system in order to indicate that proposed method is applicable. Section 6 is aimed at introducing our future works. Last section is the conclusion.

2 Review of some secret sharing protocols

2.1 What is secret sharing?

Threshold secret schemes are introduced by raising the following problem by Shamir in 1979.

How one can share a secret among n people such that each t one can find the secret and each set of $t-1$one can't gain anything.

There are a couple of conditions which each threshold secret sharing has to contain them. These conditions are as follows.

Correctness: Each t one can find the secret.

Privacy: Each $t-1$one can't find anything about the secret.

We propose two schemes briefly (For more schemes, see [Beimel, 2011]).

Recebido: dia/mês/ano Aceito: dia/mês/ano

2.2 Shamir Secret Sharing Scheme

Shamir used interpolation of polynomials to find an answer for mentioned problem. He supposed that $f(x) \in F_p[x]$ is a polynomial with degree $t-1$. He assumed that $s = f(0)$ is the secret and shared $s_i = f(i)$ as the parties for each one. One can see that if t out of n one share their secret parties, they can find f and particularly s. So correctness is satisfied.

Also it is easy to see that there are p different polynomials satisfying in each $t-1$ points. So if p is big enough, privacy is guaranteed.

2.3 Jointly Secret Sharing Scheme

One of the main questions which arises in Shamir scheme is its need to a trusted center in order to choose the polynomial. However, it is not very desired, since a trusted center is usually hard to find. Also sharing one's piece of parties is needed sometimes. If everyone can share his/her parties, then he/she can share it to all of his/her trusted people. In this way, if he/she is absent and all of his/her trusted people are present in sharing process, they can find his/her part. Jointly secret sharing scheme is proposed to solve these problems.

In this method, first i th one choose a polynomial with degree $t-1$ and send it with private channel. Then the polynomial $f = \sum_i f_i$ is calculated and $s = f(0) = f_1(0) + \cdots + f_n(0)$ is considered as the secret. Also $s_i = f(i)$ is chosen as the i th one's secret part. Then i th one with t_i trusted people chooses a polynomial g_i with degree $t_i - 1$ such that $s_i = g_i(0)$. Then i th one send $g_i(j)$ with private channel to his/her j th trusted one.

If t out of n one share their parties, they can find f and s. Also if one participant is absent, but his/her trusted ones share their parties which he/she entrusted them, they can find part of their absent friend. This methods satisfies correctness and privacy same as Shamir scheme and does not need any trusted center. Also each one can share his/her part to his/her trusted friend.

3 Secret sharing in E-voting protocols

3.1 Elliptic Curves

Every elliptic curve can be considered as an algebraic curve with the following equation (See [Koblitz, 1987]).

$$y^2 = x^3 + ax + b \tag{1}$$

The set of points on the curve with the following operation, make a group. Addition of two point $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ is

$$P_1 + P_2 = \begin{cases} 0 & \text{if} x_1 = x_2 \, \& y_1 = -y_2 \\ (x_3, y_3) & \text{Otherwise} \end{cases} \tag{2}$$

and

$$(x_3, y_3) = (\lambda^2 - x_1 - x_2, \lambda(x_1 - x_3) - y_1) \tag{3}$$

Where

$$\lambda = \begin{cases} \dfrac{3x_1^2 + \alpha}{2y_1} & \text{if} \quad P_1 = P_2 \\ \dfrac{y_2 - y_1}{x_2 - x_1} & \text{Otherwise} \end{cases} \tag{4}$$

Getting k from kP is known as discrete logarithm problem which is very hard to solve. In fact, if $NP \neq P$, this problem does not have any solution with polynomial order. So it can be used for cryptosystem.

3.2 Secret Sharing For Voters and Authorities

There are two kinds of groups in almost all of voting protocols.

Voters: People who vote. It is clear that these people should not share their secret to others, if the system is anonymous. So secret sharing is useless among voters.

Authorities: People who make process on votes. There are two kinds of systems based on number of voters: with one authority or multi authority. It is possible to exploit secret sharing schemes in multi authority protocols.

It is possible to share a secret in multi authority systems in different ways. For example in [Cramer, 1997] and [Porkodi, 2011], Shamir

Recebido: dia/mês/ano Aceito: dia/mês/ano

secret sharing scheme is used. Also protocol in [Weber, 2006] uses jointly secret sharing scheme.

3.3 An E-voting system with shamir secret sharing

Cramer proposed a homomorphic E-voting protocol based on ElGamel with Shamir secret sharing scheme in [Cramer, 1997]. Then Porkodi improved this protocol by using ECC based cryptosystem in [Weber, 2006].

We review the protocol with two candidates which was designed by Porkodi. First, center chooses point P, field GF_p, secret key s and polynomial f. Then center sends $s_i = f(i)$ to each authority as his/her private key. Next, $h = sP, h_i = s_i P$ is announced in bulletin board by center as public keys of center. Then each voter encrypts his/her vote $v_i \in \{-1,+1\}$ by ElGamel system as follows and send it with a zero knowledge proof of authentication in bulletin board.

$$c_i = (c_{i,1}, c_{i,2}) = (\alpha_i P, \alpha_i h + v_i P) \qquad 5)$$

(

After the end of voting, authorities compute summation of all votes $c = (c_1, c_2)$ and i th authority announces $s_i c$ with his/her proof of knowledge in bulletin board. If a subset of t honest authorities (name this subset J) share their result, then one can see the following by Shamir secret sharing scheme.

$$s = \sum_{j \in J} \prod_{k \in J, k \neq j} \left(\frac{k}{k-j}\right) s_j \qquad (6)$$

so

$$sc_1 = \sum_{j \in J} \prod_{k \in J, k \neq j} \left(\frac{k}{k-j}\right) s_j c_1 \qquad (7)$$

and finally

$$c_2 - sc_1 = \qquad (8)$$

$$(\sum \alpha_i) sP + dP - s(\sum \alpha_i) P = dP$$

It is enough to form the following table and find d (result of voting) thanks to dP table checking.

$$\{-NP, -(N-1)P, \cdots, -P, 0, P, \cdots, NP\} \qquad (9)$$

3.4 An E-voting system with jointly secret sharing

Weber in [Weber, 2006] exploits jointly secret sharing to propose a coercion-resistant E-voting protocol. His protocol resists against threats for voters because of the fact that each voters can vote more than once and he used a part of mixers, blind signature and homomorphic idea in his proposed voting system. He could find some ways to speed up this method, too. There are four steps in his protocol. We review them in very brief way, but we omit mixing process. This is because of the fact that we want to show the effect of secret sharing in more explicit way. The following protocol is not coercion-resistant.

First step: Setting up

Each authority like A_i chooses $x_i \in F_p$ and polynomial f_i with degree $t-1$ such that $f_i(0) = x_i$ in order to choose a secret key for the protocol. Then he/she sends $f_i(j)$ to his/her j th trusted person by private channel. He/she sends g^{x_i} by public channel. Also the public form of secret key is announced in bulletin board as follows.

$$pk_A = \prod_{i=1}^{n} g^{x_i} \qquad (9)$$

Then each authority checks other authorities' public key. Finally the hash tables and keys (h_i, z_i) are designed. These tables are designed for zero knowledge proofs.

Second step: Registering Voters

Each authority sends secret key σ_i to voters. This key is used for proof of authentication.

Third step: Vote casting

Each voter chooses a random number k. He/she sends his/her vote as follows with a zero knowledge proof of authentication.

$$(x, y) = (g^k, pk_A{}^k v_j) \qquad (10)$$

Fourth step: Tallying

First, each authority checks proofs of authentication. Then they tally the votes as

Recebido: dia/mês/ano Aceito: dia/mês/ano

follows. If there exists t honest authorities who share their results, one can find all f_i, x_i for each i using jointly secret sharing.

Finally, g^d is computed (public form of result) by multiplying all ballots and d can be found by forming the following table.

$$\{g^{-n}, \cdots, g^{-1}, 1, g, \cdots, g^n\} \qquad (11)$$

4 A Problem in Reviewed Secret Sharing Scheme

Above mentioned protocols had not paid attention to the difference role of people in sharing process. There are some methods which paid attention to the people in different categories which can be found in [Benaloh, 1990]. But it is not possible to stop sharing protocol by a specific category and there is not priority in order of sharing. To put it simply, there is no difference that A shares his/her parties first or B does it first. The difference among authorities which was considered in past research goes back to increase the number of parties of one person to increase his/her role in sharing. In this way, one member is more important than others. But in general there is no protocol which consider the importance of priority in sharing. This causes some problems. To illustrate these problems in application, take the following example.

Suppose that one of mentioned secret sharing protocols is applied in a university. The ones who can share the secret are as follows. There are 5 members in security part, 4, 6 and 6 members in education, financial and research part, 2 professors of science and 2 professors of Engineering department. So there are 25 keys and the secret sharing scheme which is implemented is (25, 14) Shamir threshold secret sharing.

Assume that there is a top secret document which is aimed at education in engineering department. Now suppose that 6 financial and 6 research expertise and 2 professors from science are willing to share their secret. These 14 people can find document, while no one from engineering department is not in sharing process. Also there is no expert in educational matters. In this situation, there is not even someone from security part to prevent or control the sharing process and decides how to take care of security proceedings after reading the secret.

As one can see, these problems are remained to solve. In the sequel, we propose a new secret sharing scheme which can solve these problems. The main idea of our method can be applied in a lot of secret sharing methods. We implement it on Shamir secret sharing.

5 Proposed method in jointly secret sharing based on Shamir scheme

We assume that there are some layers with some parts in each of them in this method. Also we mean that person or part A has priority according to person or part B in secret sharing, if B cannot share his/her secret when A has not do it. The main purpose is to make priority among layers. Also there is not priority among people in parts of same layer.

There is one polynomial for each part. To do this, we use multiplication of polynomials for each part and composition of polynomials for each layer. If one has all of these polynomials, he/she can find the main polynomial and consequently the secret key of the protocol.

Let there are L layers, M_i parts for i th layer and $n_{i,j}$ one in j th part of i th layer. Assume that one needs $m_{i,j}$ one to find the polynomial of j th part of i th layer with the condition $m_{i,j} > 1$. So it is enough to choose a polynomial $f_{i,j}$ with degree $m_{i,j}$. Then k th one in j th part of i th layer is received $f_{i,j}$ as his secret parties.

If there are $t_{i,j}$ member (name them J_1) in j th part of i th layer, then $s_{i,j}$ can be represented as follows.

$$s_{i,j} = \sum_{j \in J_l} \Big(\prod_{k \in J_l, k \neq j} \big(\frac{k}{k-j}\big)\Big) \qquad (12)$$

Now $S_i = \prod s_{i,j}$ is considered as the key of i th layer. If one has f_i in each layer and considering the following equation, one can find $s = f(0)$ which is chosen as the key of protocol.

$$f = f_1 \circ \ldots \circ f_L \qquad (13)$$

(Function f_i is equal to $\prod f_{i,j}$).

Recebido: dia/mês/ano Aceito: dia/mês/ano

According to threshold condition, each part needs to have some members who are willing to share their parties in order to reveal the polynomial of each part. So if we consider the first (or last) function for security part, then it is easy to stop the sharing protocol, if security part detects any adversary or traitor or danger. Also if he/she finds out that the secret document needs some more measures to reveal, he can prepare the condition. Moreover, for each secret, there are some relevant experts in sharing protocols. Generally this protocol can be applied to joint secret sharing. However, in voting application, these two process cannot be applied at the same time. This is mainly because of existence of public key. It requires to do more research for some special cases when one needs to have a jointly secret sharing with priority in E-voting protocol. We introduce a voting protocol which is based on the protocol in [Porkodi, 2011].

6 Proposed secret sharing scheme in E-voting based on ECC and Shamir secret sharing

We propose a voting system based on protocol in [Porkodi, 2011] and our method in secret sharing. This system has the following parts.

1- Trusted center
2- L layers for authorities and n_i one in i th layer.
3- M Voters
4- Two candidates
5- Verifiers

Our purpose is that if there are m_i authorities in i th layer, then one can find the polynomial of that part. There are 3 steps in our proposed protocol.

First step: Setting up

First, center chooses keys s_i, prime numbers p, q, elliptic curve E_p, polynomials $f_{i,j}$ with degree $m_{i,j}$ and point P on elliptic curve.

Also $s_i = f_i(0)$ is considered as the secret key of i th layer. Finally polynomial f_i is as follows

$$f = f_1 \circ \cdots f_L \tag{15}$$

The total secret key $s = f(0)$ (secret key of center) is assumed. Next, center sends $s_{i,k} = f_i(k)$ for k th authority in i th layer by private channel. Also he/she computes $h_{i,k} = s_{i,k}P$. Then he/she announces P, q, p, E_p and $h_{i,k}$ in bulletin board.

Also center computes

$$h^r(x) = (h_1^r, h_2^r) = f_r(x)P.$$

After that center ignores all coefficients of $1, x^2, x^3, \cdots$. So the remaining part is a polynomial in F_p like follows.

$$l^r(x) = (l_1^r, l_2^r) = (xe_1^r, xe_2^r)$$

Then center finds

$$W^{i-1} = h_1^{i-1}(l^i)$$

and then find l^{i-1}.

Where $W^r = h_1^r(0)$ and for each $i \in \{1, 2, \cdots, r\}$

$$h^{i-1}(x) = (h_1^{i-1}, h_2^{i-1}) = f_{i-1}(x)P$$

Then center announced $h = l^0$ in bulletin board. Finally voters are registered and are given a_i by center.

Second step: Vote Casting

Every voter sends his/her vote $v_i \in \{-1, +1\}$ with zero knowledge proof of authentication as follows.

$$B^i = (a_iP, a_ih + v_iP) \tag{17}$$

where a_i is a random number.

Third step : Tally computing

Each authority computes the vector $B = \sum B^i$ where each B^i is an honest ballot. Then he/she blinds first entry of B by computing $B_1 s_{i,k} = u_{i,k}$. Then he/she sends it to bulletin board. If there exists m_i ones in i th layer (Name this set of participants as J_i), then $B_1 s_r$ can be found as follows. First, the r th layer computes

$$f_r(x)B_1 = \sum_{l \in J_{ri}} (\prod_{\substack{l \in J_{ri}, \\ k \neq l}} (\frac{k-x}{k-l}) s_{i,k}) B_1 \qquad (18)$$

Then this layer finds y^r by ignoring constant and coefficients with degree more than one. If we assume $B = (\alpha P, \alpha h + dP)$, then it is easy to see that

$$y^r = \alpha l^r$$

Then layer $r-1$ computes y^{r-1} as follows. First, They find $f_{r-1}(x)B_1$. Then he ignores constant and coefficients with degree more than one and finds y^{r-1}. It is obvious that

$$y^{r-1} = \alpha l^{r-1}$$

Note that it is impossible to find α, since one has to solve a discrete logarithm problem which is impossible. If we continue this way, it is easy to prove that by induction that

$$y^i = \alpha l^{i-1}$$

So each layer like i computes y^i and sends it to $i-1$th layer. In this way, $y^0 = \alpha l^0 = \alpha h$ can be computed finally. Then $dP = B_2 - \alpha h$ is computed where d is the result of voting. Finally one can find d by forming a table.

7 Analysis of the secret sharing protocol and Future works

Our proposed protocol gives priority in secret sharing which is a little harder to be attacked. This is mainly because of two reasons. First is that it is hard to find a specific number in each part to cheat them or get their keys by force. Also betraying of a part (no matter how big is this part) cannot be enough to find the key. But in the meanwhile, one layer can be broken easier than before. Since secret of each layer is dependent of less people.

Here we investigate privacy and correctness in this method. In order to demonstrate the correctness, first we should show that correctness is true in each layer. This is because of the fact that sharing in each layer is a threshold shamir secret sharing and this results in the correctness in each layer satisfies. Hence one can find the polynomial of each layer, if the threshold condition of that layer is satisfied. So because of equations (14) and correctness of each layer, correctness of protocol is proved.

To prove privacy, suppose that less than number of threshold participants want to share their secret. So threshold condition of at least one layer is not satisfied. Assuming privacy of threshold shamir secret sharing, it is impossible to find the polynomial of that layer. So because of (14) it is impossible to find the polynomial of protocol. So privacy is satisfied.

Moreover, we discuss 2 factors fairness and robustness of proposed E-voting system to justify our claim about security in an application of mentioned secret sharing protocol. The condition of fairness is satisfied, if and only if the result is announced exactly after sharing process of the last layer of authorities. Also there is impossible that one traitor layer can announced the results before the voting is finished. Because even if it has the keys to compute its result form (y^i), it cannot has the keys of next layers, unless all of layers are willing to betray which is not our assumption. Moreover, during the vote casting phase, finding the result of voting in each moment is even harder than voting based on ordinary secret sharing schemes. This results from existence of more than one layer which has to be attacked. Also it is more unlikely that there exists more betrayers than threshold of that layer. So proposed E-voting scheme is more robust. It is more desired that threshold of each layer be more than one authority.

In the future works, we aimed at analyzing this protocol and improve it for different kinds of attacks. Also we will try to find an E-voting protocol which can exploit both jointly sharing and priority at the same time.

8 Conclusion

We reviewed some secret sharing schemes and E-voting protocol in this paper. Also we proposed a new method of secret sharing which provides priority in sharing. To do this, we exploited composition of functions which is a non-commutative operation. Then we implemented this contribution in voting system [Porkodi, 2011] to justify proposed sharing method. Finally we analyzed proposed secret sharing protocol.

Recebido: dia/mês/ano Aceito: dia/mês/ano

Acknowledgment

This paper is a result of a research about E-voting protocols at Iran Electronic Industries Corporation. We thank Amin Kaffashzadeh for their useful comments and Farzad Padashnik for his help to prepare this paper.

References

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. Em: Proceedings of the 2nd International Symposium on Information Theory, pp. 267–281.

Shamir, A. (1979). How to Share a Secret. Communications of the ACM, 22(11), 612-613.

Blakley, G.R. (1978). Safeguarding Cryptographic Keys .Managing Requirements Knowledge, International Workshop on (pp. 313-313). IEEE Computer Society.

Blundo, C. De Santis, A. Stinson, D. R. Vaccaro, U. (1995). Graph Decompositions and Secret Sharing Schemes. Journal of Cryptology, 8(1), 39- 64.

Liu, M. and Zhou, Z. (1998). Ideal Homomorphic Secret Sharing Schemes over Cyclic Groups.Science in China Series E: Technological Sciences,41(6), 650-660.

Cleve, R. Gottesman, D. and Lo, H. K. (1999). How to sShare a Quantum Secret. Physical Review Letters, 83(3), 648.

Hillery, M. Bužek, V. and Berthiaume, A. (1999). Quantum Secret Sharing.Physical Review A, 59(3), 1829.

Beimel, A. (2011). Secret-Sharing Schemes: A Survey. Coding and Cryptology. Springer Berlin Heidelberg, (pp. 11-46).

Ingemarsson, I. and Simmons, G. J. (1991). A Protocol to Set up Shared Secret Schemes without the Assistance of a Mutually Trusted Party. Advances in Cryptology Eurocrypt'90, Springer Berlin Heidelberg. (pp. 266- 282).

Cramer, R. Gennaro, R. Schoenmakers, B. (1997). A Secure and Optimally Efficient Multi-Authority Election Scheme. European Transactions on Telecommunications, 8(5), 481490.

Tian, Y, Jian. F, Changgen. P, Qi. J, (2013). Fair (t, n) threshold secret sharing scheme. Information Security, IET 7.2:106-112.

Shyu, S.J. (2014). "Visual secret sharing with meaningful shares." Science and Information Conference (SAI).

Wang, R, Shyong-Jian S. (2007). Scalable secret image sharing. Signal Processing: Image Communication 22.4: 363-373

Nojoumian, M, Douglas R.S, Morgan G. (2010). Unconditionally secure social secret sharing scheme."IET information security 4.4: 202-211.

Cramer, R. Gennaro, R. and Schoenmakers, B. (1997). A Secure and Optimally Efficient Multi-Authority Election Scheme. European Transactions on Telecommunications, 8(5), 481490.

Porkodi, C. Arumuganathan, R. and Vidya, K. (2011). Multi-Authority Electronic Voting Scheme Based on Elliptic Curves. IJ Network Security,12(2), 84-91.

Fouard, L. Duclos, M. Lafourcade, P. (2007). Survey on Electronic Voting Schemes. Supported by the ANR Project AVOTÉ.

Weber, S. (2006). A Coercion-Resistant Cryptographic Voting Protocol-Evaluation and Prototype Implementation. Darmstadt University of Technology, http://www. cdc. informatik.tudarmstadt.de/reports/reports/StefanWeber.diplom. pdf.

Benaloh, J. C., Leichter, J.(1990) Generalized secret sharing and monotone functions. Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 27–35.

Recebido: dia/mês/ano Aceito: dia/mês/ano