International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 5(2013) pp. 581-588

Secure Auditing and e-Commerce

N. Antequera[1], R. García-Rubio[2] and J.A. Lopez-Ramos[3]

[1]Hanscan Spain, S.A.,
Avenida de la Vega 1, Alcobendas, Madrid 28108, Spain
nicolas.antequera@gmail.com

[2]University of Salamanca, Departamento de Administración y Economía de la Empresa,
Campus Miguel de Unamuno s/n, Salmanca 37007, Spain
rgr@usal.es

[3]University of Almeria, Department of Algebra and Analysis,
Carretera de Sacaramento s/n, Almeria 04120, Spain
jlopez@ual.es

Abstract: **This paper deals with the necessity of protecting information in the auditing process mainly in the actual e-society characterized for the continuous information exchange carried out nowadays by companies and accounters. We introduce a system that allows this information exchange, applicable for other e-systems, in an efficient and secure manner.**
Keywords: Information protection, authentication, auditing, multicast.

I. Introduction

Information society has produced new concepts in human relationships that give raise to e-systems that organize or promote wider relationships and activities such as e-commerce, e-learning or e-governance. These e-systems are characterized by integration, automation of processes and controls. Our e-society increases reliability of (and dependence on) systems what is also accompanied by an increased need for accurate and timely information [9]. This creates the need for new approaches in audit procedures and auditing architectures in all these e-systems, but especially in the largescale socioeconomic systems.

Remote access software and client/server technology are in the core of any e-activity, but they show to be essential for accountants to keep pace with the business. Technology has a great impact at every phase of the audit process and provides efficiency and effectiveness. Computer-generated audit programs or audit software capable of testing the entire population of the client's data helps undoubtedly the accountant with his task and this can use remote access software to run programs on a computer at another location, download files, or discuss issues with another accountant at the remote computer ([4]). The use of information technology have influenced undoubtedly in the accounting and auditing process and its benefits have been also studied by some authors (cf. [6] for example).

Nowadays every investment-banking firm and many big companies use packages that provide management and accounting with a consolidated view of risk across business lines and helping to make strategic decisions. An example of the contribution of accounting information systems (AIS) to knowledge management and strategic role of the organization can be found in [19]. Auditing plays a central role in preventing fraud. Beasley, et al. [3] conducted an archival study of fraudulent financial reporting. A 2001 study by Church et al., [5] reveals that internal auditors are able to identify certain risk factors for fraudulent financial reporting. Accounting scandals of the early decade have led to increased awareness of both the regulators and organizations (public and private) for internal control. Thus we are at a stage of maturity of the organizations and improving efficiency and control of their activities has become one of the basic needs. Some authors claim that within the various activities involved in internal control strategy of the organizations, control over the management of information systems every day becomes more relevant ([8], [15]). At the time when organizations become aware of the need to increase the level of control over the management of information systems, there is the next concern. Are data integrity available to the auditing firm? There exist many risks associated with the computer-to-computer exchange of business information and transactions, particularly in ad-hoc networks (cf. for instance [10]) due to the massive use of smart devices, and that obviously also affect auditing such as unauthorized access, virus contamination, information vandalism, theft of information, confidentiality and security of transactions, and hacking by outsiders. Supporting of financial auditor, the computer auditor treats the accounts, but questions the reliability thereof too. New control objectives, such as control over information access, management of authorization and registration mechanisms of activity on that information start being considered. Therefore the information processed by information systems must have an adequate security level to its value and the associated risks to its use. Clear examples of this fact are [1] where the author analyzes possible threats that any accounting information system based on modern information

technologies faces off, or [16] where ubiquitous computing, nowadays commonly used to provide services, some of them required to be carried out in a secure manner as pointed out above for e-auditing, is analyzed from the security point of view.

Modern technologies allows to consider new methods for authentication and secure access and/or distribution of the information, as the method considered in [17]. The aim of this paper is to propose a system for accessing the distributed information for accounting based on multicast communications satisfying the main requirements of any system implemented for a secure treatment of information: confidentiality, the information can be accessed just for authorized users; authentication, source and destination of information are authenticated; integrity, users can ensure that information has not been altered during the communication process and finally, but not less important, non-revoking, i.e., source of the distributed information cannot be denied. Multicast communications allow a host to simultaneously send information to a set of other hosts, avoiding the establishment of point-to-point connections with all of them ([18]). Key management is a main issue in secure multicast. Efficiency in the key management, including key storage and refreshment and perfect forward and backward secrecy are required, i.e., a new user should not be able to decrypt the contents before joining the multicast group and an old user should not access the encrypted information after leaving, trying to minimize key storage and communication overcomes, although sometimes the interest on secrecy of contents may expire. The system we are introducing in this paper makes use of the protocols presented at [2] after the cryptoanalysis developed in [14] of a group of algorithms proposed in [13]. The system is quite resilient against faulty networks due to the fact that users can derive all the rekeying information from their previously predistributed private information and so they are of particular interest in the above mentioned situations. We also include a method to authenticate the rekeying information, in order to avoid external non-desirable intrusions that can steal the distributed auditing information and also a method to authenticate users that distribute the information to be audited and therefore, we avoid possible forging of this information, enhancing the auditing process. Both former algorithms do not pretend to substitute traditional digital signatures and certificates respectively, but may constitute an alternative to be used in some light devices that cannot admit them or slow down the communicating process due to computational requirements.
The paper is structured as follows. In the first section we propose the architecture to be used by the auditing system. Then we introduce and analyze security of the rekeying algorithm, the authentication algorithm for the rekeying information and the authentication protocol between users.

II. The Architecture

In Figure 1 we can observe the architecture that corresponds to an entity that makes public the information to be audited. The entity sends the information through the channel and this is received by every member in the multicast group and resent from this to the other members interested in receiving such an information. Note that in this figure we consider just one entity, sending the information to a plurality of users that can be either other entities interested in the information by commercial purposes or accounting to carry on an auditing on this information. The system would be composed by a set of these situations where communication takes place from many to many and not only from one to many.

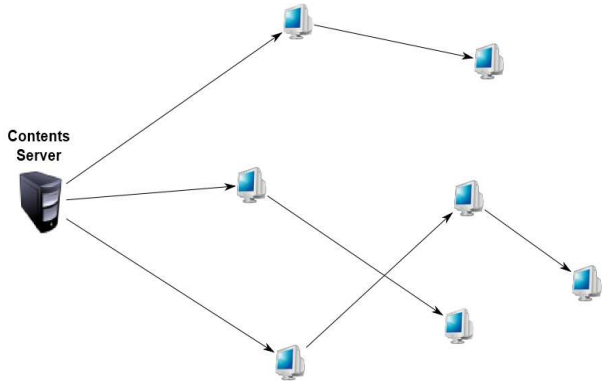

Figure. 1: Multicast: one-to-many communication

In Figure VI we include a trust entity that provides the session key to encrypt the information to be distributed. Communication of this session key is carried out similarly by every user as in Figure 1 and only those users holding some private information used to protect this session key will be able to retrieve it. Then the session key is used to encrypt the information to be audited and sent, once it is encrypted with the session key.

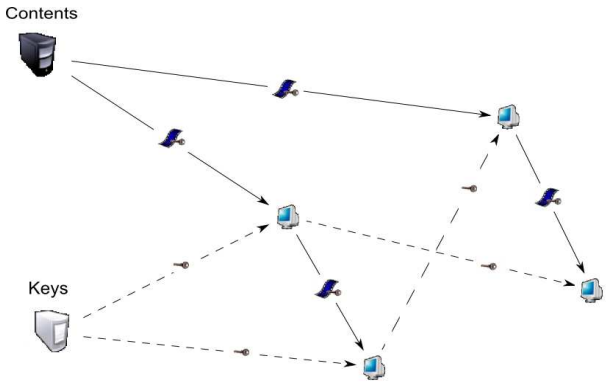

Figure. 2: Secure Multicast Centralized

Thus we could summarized the scenario as follows:

- Audience is composed by a set of clients.
- Communications are encrypted with a session key.
- A Key Server distributes and refreshes the session key by means of one-to-many communications.
- Communications among clients may be either one-to-many or many-to-many.
- Clients can enter/leave the system at any time.
- We require perfect forward (and backward) secrecy.

III. Protecting the Information

Through this section we will describe in detail the process of protecting the information and how the session key is refreshed, which is depicted in Figure 2. We will also show that the protocol is resistant against passive and active attacks.

We recall that when an entity wishes to make public any information this will be encrypted and sent as shown in Figure 1 in an efficient way (cf. [18]). The point now is how to refresh the session key also in an efficient and secure manner.

Every user in the system is assigned a prime number x_i that we will denote a ticket. This ticket is given during the first connection of the client to the Key Server and it is communicated in a secure way. The client will not have to access the Key Server again to this end during the life-time of the ticket, i.e., we can assign a caducity to every ticket and after it expires, the client should ask for a new ticket to keep accessing the distributed information.
The set up process is carried out in a very simple an efficient way. The Key Server selects

- m and p, large prime numbers, such that $m-1 = p \cdot q$.
- k and δ, such that $\delta = k + p$ and $\delta < x_i$, for every $i = 1, \ldots, n$.
- g that verifies $g^p = 1 \ mod \ m$.
- The session key to be distributed will be $g^k \ mod \ m$.

However, the session key, as noted above above should be distributed in such a way that only authorized users can access it and thus, they will be able to retrieve the information encrypted with such a key. To this end, the Key Serve just makes two simple computations:

- First, it calculates the product of all the valid tickets in a determined session, $L = \prod_{i=1}^{n} x_i$. This will be kept private at the Key Server..
- Secondly, it computes two integers u, v, by means of the Extended Euclidean Algorithm (cf. [11]), and such that

$$u \cdot \delta + v \cdot L = 1$$

Then the Key Server broadcasts u, as well as the public information g and m. Recovering the key is straightforward for every user. Assume that a company holds the ticket x_i. Then it calculates $u^{-1} \ mod \ x_i = \delta$ and $g^{\delta} \ mod \ m = g^k$ since $g^p = 1 \ mod \ m$. Then using this session key that can be a symmetric key, any information is encrypted and securely distributed. When an accountant receives the information, he uses his own ticket to derive the secret key, and then gets the information. It is important to note at this end that everyone holding a valid ticket x_j gets the same δ and so, the same session key. The reason is that δ is the only solution of the congruence system $u \cdot x = 1 \ mod \ x_j$ for every j in the range $[0, \ L-1]$. Thus, using this efficient way of distributing information we avoid unauthorized access since only those users holding a valid ticket during a session will be able to decrypt it. The consequences are that it will be impossible to theft of information, as we will show below, and of course, confidentiality and security of transactions.

In [13] the authors pointed out that every user holding a valid ticket is able to compute δ and therefore it is possible to get the number $u \cdot \delta - 1$, which is a multiple of the product of every ticket, L, since x_i is a factor of L that divides $u \cdot \delta - 1$. Thus, if this user is able to factorize this number, then he would get all the valid ticket at the session given by u. But this may involve the factorization of a huge number if the primes x_i are selected large enough.

A very well-known attack to any key exchange is the so-called "man in the middle" attack. In that case a forger impersonates the Key Server identity and sends false messages containing the session key. Then, information sent by any user could only be decrypted by this forger and use the information maliciously.
In [14, Section 3], the authors propose a "man in the middle" attack using the above mentioned multiple of L. The forger, as noted before, possibly a previously legal user h generates a new value $\delta' < \delta$, and computes u' and v' applying the Extended Euclidean algorithm, such that

$$u'\delta' + v'(vL) = 1$$

Then member h sends g, m and u' to the rest of users. Those members will obtain the new value $\delta' = u'^{-1} \ mod \ x_i$, and compute the refreshed key as in the proposed protocol.
It can be seen that, as usual with this kind of attacks, this issue is easily solved with any authentication associated protocol, i.e., the Key Server should send some attached information allowing the users to authenticate the source of such rekeying messages. This can be carried out by using typical digital signatures ([11]), although we will also propose an authentication method that can be derived from the rekeying method.
However this is not the only menace that a key distribution scheme face off when considering a group of users. Another possibility is that any legal user could try to get some other user's private information. Thus the attacker could still get all the information sent through the network even after his own ticket's caducity, this time using the private information of someone else. However this is not the only problem. Our system aims to be a way where entities can send securely their information to other entities or accountants in charge to carry out an auditing. In our system, as we will propose later there is a way to authenticate legal users and therefore, if someone is able to get the private information of anyone else, then the forger could impersonate this user and sent fake information with the disastrous consequences that this could be imply for an entity that it is being audited, in case this information shows worse results than the real one or, on the other hand, if the fake information shows better results than the real, then when the truth came up, this entity's credibility would be questioned by the accountants, but specially, by customers and investors.

So let us show consequences and simple ways to avoid two different attacks considered in [14, Section 3].
Firstly consider that a rekeying has taken place and the components of the group of users, entities and accountants has

not changed. So as noted above, any legal user has computed a multiple of the product of L, namely $v \cdot L$. previously to the rekeying. After rekeying, a new u' has been distributed and then, the same legal user has access to a new multiple of L, $v' \cdot L$. Then the attacker can compute the greatest common divisor of these two multiples, and if v and v' are coprime, then this will reveal L. If the group of participants is stable, then this strategy could be carried out for a period until the divisor obtained appears a determined number of times, what will imply that the factor appearing is precisely L. However this does not constitute a problem since, as we noted above, if the attacker pretends to get other users' tickets, then he will have to factorize L, that is formed by primes large enough to avoid the more simple case when any user tries to factorizes simply the multiple $v \cdot L$.

A second case involves a sequence of different rekeying messages where the group of users is changing, entering or leaving the system. The attacker can try to apply successively Euclid's algorithm. The strategy here is similar, but in this case, the attacker applies the algorithm until a ticket is revealed, i.e., the algorithm is applied until the output is precisely a prime number, that would correspond to some user's ticket. This has an easy solution by just considering as a factor of the product L a ticket that is not revealed to anybody and kept secret by the Key Server. This fake ticket can be changed after a determined number of refreshments and thus, in the worst case, the output of Euclid's algorithm would give a integer that is a product of two large primes and again, a factorization attack will not be feasible.

IV. Authenticating Rekeying Messages

As we mentioned in the previous section any key distribution scheme is susceptible of an attack by forging the Key Server's identity. The way to avoid this is to attach some additional information that allows authenticating the source of the received data. This is a very simple way to avoid viruses and their terrible consequences as vandalism or theft of information. In this section we give an authentication scheme that do not pretend to be a substitute of digital signatures, but an alternative to this in case of use light devices that may not have the computing requirements to do so many times efficiently as can be the case in real time communications without desynchronizing and the corresponding loss of information.
Let us recall first the message authentication associated to the key distribution scheme proposed in [13]. Assume that the Key Server aims to distributes the session key $g^k \ mod \, m$. Then the Key Server carries out the following actions:

- It computes $s = (g^k)^{-1} \ mod \ L$ by means of the Extended Euclidean Algorithm,
- Then it chooses a random number a, such that $a < x_i$, for every x_i.
- Finally, it broadcasts the pair $\{a \cdot s, h(a)\}$. $h(a)$ is the output of a hash operation on a.

Now, when a member i receives the authentication message, he computes $h(a \cdot s \cdot K_{pub} \ mod \ x_i)$, which should be equal to the value $h(a)$ received in case x_i is a factor of L, i.e., only users holding a valid ticket will be able not only to get the session key, but also to be sure that this session key was created by someone that knows their tickets, the Key Server. Therefore, it is convenient that the authentication message is attached to the refreshment message so authenticity can be verified upon reception.

In [14, Section 5] the authors give a way to break the message authentication protocol associated to the key distribution scheme. The attack to forge Key Server's identity proposed in [14] is as follows. Assume that a user has computed a multiple of L, say $v \cdot L$. Then he generates new values for the parameters of the system, that is, m', p', g', δ' and k'. Then, he computes u', v' applying the extended Euclid's algorithm to δ' and $v \cdot L$. In this point, the member h begins with the authentication protocol described above, but using $v \cdot L$ instead of L, that is, he generates $s' = (g'^{k'})^{-1} \ mod \ v \cdot L$, and chooses a' at random. Then, he broadcasts the forged rekeying message $(m', \ g', \ u')$ and the attached authentication message $(a' \cdot s', h(a'))$.

When a user receives the message, he uses his ticket to compute $\delta' = u'^{-1} \ mod \ x_i$, and $r' = g'^{\delta'} \ mod \ m'$. Then he authenticates r' by computing $h(a' \cdot s' \cdot r' \ mod \ x_i)$.

It is easily observed that the authentication process works since $h(a' \cdot s' \cdot r' \ mod \, x_i) = h(a')$ but only for those x_i such that $a' \leq x_i$. Thus the authentication would not work for those users whose tickets x_j are lower than a'. Therefore an entity holding such a ticket could inform the Key Server that someone is sending fake rekeying messages, trying to forge its identity.
We could increase the difficulty for developing such an attack by simply considering authentication messages $(a \cdot s, h(a))$ corresponding to a determined rekeying message such that a is above a determined bound of the same order of the least ticket, these tickets are distributed uniformly and there is not a short distance between two consecutive tickets. The attacker could then choose a' less than a determined ticket, say x_h, and therefore the fake message succeeds just for the users holding a ticket that is larger than x_h.

A. An enhanced authentication scheme

As we have shown above, the attack that is proposed in [14] succeeds for some users of the system, but there is always somebody that could detect the fake message and denounce it to the Key Server. We note that there are systems of similar functionality, used mainly in multimedia transmission in streaming as the well-known system proposed in [12] or more recently in [7], where some trusted entities or users, usually those with a longer stay at the communicating group, that may help the Key Server to maintain a vigilance on the distributed messages. If we assign this entities the lowest tickets, then they would detect this attack. Thus once the fake message is detected, then this detached user would inform the Key Server and this should track the origin of this message and detect the forger. However, while detecting the fake message and its origin some kind of fraud has possibly taken place as theft of information or distribution of fake in-

formation to be audited. Thus, we have to give an alternative in order to avoid this difficult but possible attack.

The solution that we are introducing in this section will not need the existence of this detached trusted entities and will be shown to be secure. So let us assume a system as described in the previous section where each client holds a ticket that is used for decrypting and authenticating.
Now every user holds a ticket x_i as before for decrypting the rekeying messages and another one b_i in the rank $[0, x_i - 1]$ $i = 1, \ldots, n$ that will be used for authenticating the rekeying messages. Note that we are assuming that the system is composed by n entities. If, as above, $r = g^k \ mod \ m$ is the session key to be distributed, then the Key Server carries out the following actions to build the corresponding authentication information:

- Firstly, the Key Server selects a such that $a < x_i$ for every $i = 1, \ldots, n$.
- Then it calculates $h(a)$ for h a hash function.
- Finally, the Key Server computes $s = (g^k)^{-1} \ mod \ L$, where $L = \prod_{i=1}^{n} x_i$ and solves the system of congruences $x = as + b_i \ mod \ x_i$, $i = 1, \ldots, n$, getting a solution S.

Then the information that allows authenticating the session key r is $(S, h(a))$.

Let us show now that every user is able to authenticate positively the rekeying message $r = g^k \ mod \ m$: Firstly the user computes $S_i = S - b_i \ mod \ x_i$. Then, he computes $h(S_i \cdot r \ mod \ x_i)$ and verifies that equals $h(a)$ since $S = as + b_i \ mod \ x_i$ for every $i = 1, \ldots, n$ and $s\dot{r} \ mod \ x_i = 1$.
It is clear now that although a forger could try to impersonate the Key Server by using a multiple of the product of all the tickets, L, as proposed in [14], even if she is able to factorize this product, then to forge the authentication message, he needs to know all the secret values b_i, $i = 1, \ldots, n$.

Let us remark finally that length of the authenticating information can be shortened. If as above $r = g^k \ mod \ m$ is the rekeying information to be distributed between n users, the Key Server gives every user a second ticket, namely, y_i that we will call the authentication ticket. Every authentication ticket y_i will be a prime number satisfying that $m \leq y_i$ for every $i = 1, \ldots, n$. Then the new authentication protocol is developed by using y_i instead of x_i as in the previous section. We observe that y_i's only need to be coprime in order to get the desired result given by the Chinese Remainder Theorem. Then new authentication message is of the order of the product of all the tickets y_i's, and, since there is not any possible factorization attack as in the distribution protocol, then y_i could be considered shorter, simply greater than m.
Let us show the consequences in a real system we aim to introduce in this work. Assume that our system is composed 10 entities including accountants and companies to be audited and consider that the authentication tickets are given by primes of 64 bits length. Then as we noted above the authenticating information is of the order of the product of 10 primes of 64 bits, which gives that our message will a little over 640 bits (recall that we are also sending $h(a)$ of some few bits more). Thus the authenticating information is quite shorter than a typical X.509 certificate used in standard secure communications.

V. Authentication between users

As we have mentioned several times through this work, a crucial point in any auditing process is to authenticate the source and destination of the information in order to prevent a wrong auditing with its corresponding consequences and the theft of information. Thus it would be convenient that when establishing contact with any source, the communicating parties could authenticate each other in order to be sure that both of them are legal users holding tickets given by the Key Server.
Thus assume that an accountant, that we may denote by user i, demands some information from a determined company, denoted by user j. Firstly the accountant wishes to be sure that user j corresponds to a company that has been authenticated by the Key Server and it is not a forger trying to distribute fake information on this company. If this authenticating process succeeds then the communication will be established. We will show that authentication is carried out with no disclosure of any private nor sensible information.
Let us suppose first that the session key has already been distributed, and that the public information is constituted by the prime number m and the generator g of the cyclic group $\mathbb{Z}_m$. Then the authentication of the company, user j by the accountant, user i is given by the following protocol:

- User i chooses a random integer t such that $1 < t < m$ and sends it to the Key Server.
- The Key Server computes $inv = t^{-1} \ mod \ L$ and sends it back to user i.
- Then user i sends $\{inv, \ g^{x_i} \ mod \ m\}$ to user j.
- User j calculates $t_j = inv^{-1} \ mod \ x_j$, $\beta_j = t_j \cdot (g^{x_i})^{x_j}$ and sends back $\{\beta_j, \ g^{x_j}\}$ to user i.
- User i computes $\beta_i = t \cdot (g^{x_j})^{x_i}$, which should be equal to β_j.

If $\beta_i = \beta_j$ then it is clear that j owns a valid ticket x_j. Otherwise user i should warn the Key Server and thus security measures can be taken against user j.

As it can be observed, a force brute attack to try to derive any information on the tickets x_i or x_j involves solving the discrete logarithm ([11]).
However, in [14, Section 7], introduce an attack where the attacker tries to use the above protocol to discover the ticket of the users that supposedly is going to be authenticated. The attack is as follows: firstly user i computes $dh_i = g^{x_i} \ mod \ m$, chooses inv at random and sends the pair (dh_i, inv) to user j. Then user j computes $r_j = inv^{-1} \ mod \ x_j$, $dh_j = g^{x_j} \ mod \ m$, and $\beta_j = r_j \cdot (dh_i)^{x_j} \ mod \ m$ and sends back to user i the pair (dh_j, β_j). Now peer i computes $dh = (dh_j)^{x_i} \ mod \ m$ and recovers the value $r_j \ mod \ m = \beta_j \cdot dh^{-1} \ mod \ m$. If m is greater or equal than x_j, then

$r_j = r_j \; mod \, m$ and $(inv \cdot r_j - 1)$ is a multiple of x_j. Therefore, x_j could be computed as $x_j = gcd(inv \cdot r_j - 1, v \cdot L)$ with high probability, where $v \cdot L$ is the multiple that any user is able to compute as we pointed out in the first section.

We point out that this reasoning works for those values x_j such that verifies either the prime m is greater or equal than x_j or r_j is in the rank $[0, m]$, since otherwise $r_j \neq r_j mod \, m$. To avoid this situation we could consider only those authentication messages verifying $m < r_j < x_j$. Otherwise, user j will refuse the authentication message.

We can also observe in the above protocol that every time a user wishes to authenticate another one, the Key Server is demanded a number. This could constitute a problem in systems with a big number of users since these requests could collapse the Key Server, which gives the opportunity to an attacker of a denial by service attack, i.e., the Key Server is not able attend the demands and broadcast new rekeying messages, which can be used by the attacker to forge identities in the meanwhile with the consequences previously commented.

We may address this two matters with the following easy solution. If a peer wants to get a "challenge", inv as denoted in the above protocol, to authenticate other users can generate a list of numbers and send them to the Key Server. Then the Key Server selects some of them among those numbers whose inverse verifies what was noted before. To do so, the Key Server selects inverses such that are in the rank of m and the highest ticket (including possible false tickets as proposed in Section 1). In this way the attack proposed in [14] is not applicable. Moreover, the user may have a list of authenticating messages (of single use for security reasons) that may be an alternative to standard certificates expedited by an authority. Figure 3 describes the authentication of users protocol.

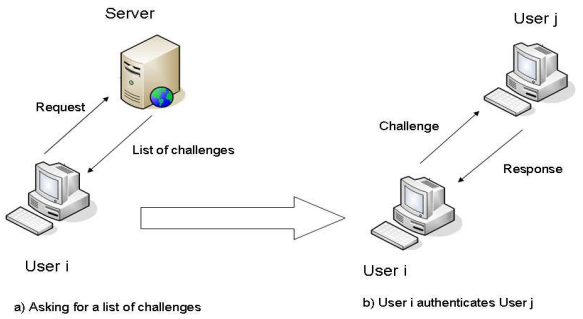

Figure. 3: Secure Multicast Centralized

We could increase the difficulty to attack this protocol by including some information in the Key Server message (second step of the authentication protocol) to allow authenticating that part of the information that was created by the Key Server and not by a forger.

VI. Conclusions

In this paper we have introduced a secure system for auditing in order to get the doubtless benefits that an auditing information system based on the modern information technologies can provide. We get the advantages of adding secure protocols to the auditing process such as unauthorized access, virus contamination, information vandalism, theft of information, etc. We give a complete security system consisting on an efficient algorithm to distribute information in a secure way, based on an efficient algorithm to compute multiplicative inverses in a ring which is a product of finite rings. We provide confidentiality of the distributed information and authentication of the source such an information. We also give a protocol to authenticate legal users, i.e. the final destination of the distributed information, auditors, accountants, investors, customers, etc. Our method also provides non-revoking, that avoids denying of possible fake accounting distributed information. Moreover, we can ensure integrity of the distributed data and thus we can ensure that the original distributed information is exactly what auditors receive in real time. In other words, we are preventing fraud, one of the main objectives of accounting. We have also shown that the cryptoanalysis on the above mentioned algorithm carried out by some authors in a previous work and that may affect different parts of our system does not compromise security, confidentiality of information, identity and secret information hold by users or source of information, nor its efficient functioning, i.e. we are foreseeing attacks by denial of service, avoiding, in this way that the accounting process is carried out in real time. We also provide alternatives to those situations where the cryptanalysis may succeed showing now its fortress against them and that can be used combined or in a single way as an alternative to other traditional cryptographic methods.

Acknowledgments

This work has been supported by the Spanish Ministry of Science and Innovation (TEC2009-13763-C02-02) and Junta de Andalucía (FQM 0211).

References

[1] A.A. Abu-Musa. The Perceived Threats to the Security of Computerized Accounting Information Systems. *T*he Journal of American Academy of Bussines, Cambridge, USA, 3(1), pp. 9-20, 2003.

[2] N. Antequera, J.A. Lopez-Ramos. Remarks and countermeasures on a cryptanalysis of a secure multicast protocol. *Proceedings of 7th International Conference on Next Generation Web Services Practices, Salamanca 2011*, pp. 201-205, 2011.

[3] M. Beasley, J.V. Carcello, D.R. Hermanson. Fraudulent Financial Reporting: 1987-1997, *An Analysis of U.S. Public Companies*. New York, NY: AICPA, 1999.

[4] J.L. Bierstaker, P. Burnaby, J. Thibodeau. The impact of information technology on the audit process: an assessment of the state of the art and implications for the

future, *Managerial Auditing Journal*, 16(3), pp. 159-164, 2001.

[5] B.K. Church, J. J. McMillan, A. Schneider. Factors affecting internal auditors consideration of fraudulent financial reporting during analytical procedures, *Auditing: A Journal of Practice and Theory*, 20(1), pp. 65-80, 2001.

[6] I. Dalci, V.N. Tanis. Benefits of Computerized Accounting Information Systems on the JIT Production Systems. *Review of Social, Economic & Business Studies*, 2, pp. 45-64, 2002.

[7] L. Garces-Erice, E. W. Biersack, K. W. Ross, P. A. Felber, G. Urvoy-Keller. Hierchichal p2p systems. *Proceedings of ACM/IFIP International Conference on Parallel and Distributed Computing (Euro-Par)* Harald Kosch, Laszlo Boszormenyi and Herman Hellwagner, editors, LNCS 2790, 1230-1239, 2003.

[8] S.M. Glover. The next generation software, *Internal Auditor*, 55(4), 1998

[9] M. Janakova. Database Technology Analysis for the Support or Instant Information Retrieval, *International Journal of Computer Information Systems and Industrial Management Applications*, 3, pp. 586-593, 2011.

[10] R. Maulik, N. Chaki. A Study on Wormhole Attacks in MANET. *International Journal of Computer Information Systems and Industrial Management Applications*, 3, pp. 271-279, 2011.

[11] A.J. Menezes, P.C. Van Oorschot, S.A. Vanstone. *Handbook of Applied Cryptograhpy*, CRC Press, 1996.

[12] S. Mittra. Iolus: A framework for scalable secure multicasting. *Proceedings of the ACM SIGCOMM (New York, Sept.)* ACM 27(4), New York, pp. 277288, 1997.

[13] J.A.M. Naranjo, N. Antequera, L.G. Casado and J.A. Lopez-Ramos. A suite of algorithms for key distribution and authentication in centralized secure multicast environments. To appear in *Journal of Computer Applied Mathematics*, 236(12), pp. 3042-3051, 2012.

[14] A. Peinado, A. Ortiz. Cryptanalysis of Multicast Protocols with Key Refreshment Based on the Extended Euclidean Algorithm. *Proceedings of CISIS 2011, Lecture Notes in Computer Sciences,* 6694, pp. 177-182 2011.

[15] F. Pons Ortega. Auditoría informática. Una aproximación a la mejora del control interno. *Auditoría pública, revista de los órganos autonómicos de control externo,* 41, pp. 97-100 2007

[16] A. Ukil. Context Protecting Privacy Preservation in Ubiquitous Computing. *International Journal of Computer Information Systems and Industrial Management Applications*, 3, pp. 228-235, 2011.

[17] M. Vashek, R. Zdenek. Security of Biometric Authentication Systems. *International Journal of Computer Information Systems and Industrial Management Applications*, 3, pp. 174-184, 2011.

[18] S. Zhu, S. Jajodia. Scalable group key management for secure multicast: A taxonomy and new directions, *Network Security*, Springer, pp. 57-75, 2010.

[19] M.S. Zulkarnain. Accounting Information Systems (AIS) and Knowledge Management: A Case Study. American Journal of Scientific Research, 4, pp. 36-44, 2009.

Author Biographies

First Author Nicolas Antequera is technical director of Hanscan Spain, company dedicated to development of authentication and encryption systems and e-commerce. He got a Master Degree in Physics and Engineering at University Complutense of Madrid in 2007. He has also worked as an engineer at the European Space Agency and actually his research interest includes Applied Cryptography and Biometrics.

Second Author Raquel Garcia-Rubio is an Assistant Professor of Accounting and Business Administration in the Faculty of Economics at University of Salamanca, Spain. She received a PhD from the University of Alicante, Spain in 2009. Nowadays she teaches in the Faculty of Economics at the University of Salamanca, Spain. Her research interests include Corporate Social Responsibility, Environmental Accounting and Auditing.

Third Author Juan Antonio Lopez-Ramos is an Associate Professor at University of Almeria, Spain. He got a PhD in Mathematical Sciences at the University of Almeria in 1998. He is author of more than thirty peer-reviewed papers and a book on Homological Algebra. His research interest includes Algebra and Applications to Information Theory, particularly Coding Theory and Cryptography.