In: *C. Farkas, P. Samarati (Eds.): Research Directions in Data and Applications Security XVIII; IFIP TC11/WG11.3 Eighteenth Annual Conference on Data and Application Security; Sitges; Catalonia; Spain* pp. 265–280, Kluwer, Juli 2004

EFFICIENT VERIFICATION OF DELEGATION IN DISTRIBUTED GROUP MEMBERSHIP MANAGEMENT

Ladislav Huraj
Department of Computer Science,
Faculty of Natural Sciences, Matthias Bel University, Slovak Republic
huraj@fpv.umb.sk

Helmut Reiser
Munich Network Management Team,
Department of Informatics, Ludwig Maximilian University Munich
helmut.reiser@nm.ifi.lmu.de

Abstract In ad-hoc networks and other highly distributed and decentralized environments, authorization certificates can be used to control access. Moreover, it is possible to delegate rights listed in the certificate to another users. Several such subsequent delegations build a chain of certificates. Chains of delegation certificates can improve the capability and manageability of systems.

Distributed group membership management, i.e. managing groups of users in a distributed environment, utilizes the efficiency of certificate chains. Adding, removing as well as authentication of users is managed by chains of delegation certificates. On the other hand, the size of certificate chains could be too long to be usable.

In this paper we take a look at distributed group membership management based on delegation certificates. Then we present a new kind of certificate, the implanted chain certificate, and its characteristics. With this new kind of certificate the verification time of a certificate chain can be decreased without losing delegation information. Finally, we compare our verification method to reduce the time of verification.

Keywords: Authorization, Delegation, Chain of certificates, Verification

1. Introduction

Access control mechanisms in networks are used to avoid unauthorized users to access data or services. In traditional networks, e.g. an Access Control List (ACL) was used for this purpose. ACL involves access control information of the users. In decentralized, distributed systems such as ad-hoc networks, new users can be added or removed anytime, which disallows predefined ACLs. Consequently, new methods of authorization, such as delegation of rights are applied. In our paper we deal with a group membership management based on authorization certificates and a solution reducing verification time of delegation is presented. Moreover, our scenario is founded on ad-hoc networks.

A wireless ad-hoc network is a network where two or more devices communicate with each other using wireless transmission without the required intervention of any centralized access point or existing infrastructure. The topology in ad-hoc networks can change rapidly as nodes move in and out of each other's range. Ad-hoc networks are suitable for many applications, rescue, emergency and civil defense operations, team working applications, military systems, virtual classrooms or even local area networks.

Authorization certificates are used to control access. They grant access permissions to an entity, the entity is trustworthy for the issuer granting the permission rights. Moreover, every other entity, which obtained a certificate from a trustworthy entity, can be trustworthy for the issuer as well. These types of certificates which delegate authority from one entity to another are called *delegation certificates*. Chains of delegation certificates can improve the capability and manageability of the authorization process, the responsibility is distributed among several users and a user does not have to manage the authorization of each entity itself. However, the size of certificate chains in delegation systems could be extremely high.

For example, we consider a meteorological office (MeteO) which gathers weather, pollution, and other environmental phenomena from the landscape. MeteO consists of a center, but also of many particular static or mobile stations which can work independent of MeteO center, as well as individual scientists in the field and distributed environmental sensors. All these parts of the meteorological office are called MeteO members.

MeteO members can communicate with each other, share their results and exchange their particular field measurements. Because members can change their position in a landscape area, they use an ad-hoc network for communication. The MeteO membership is delegated through chains of delegation certificates, e.g. the MeteO center does not delegate the membership directly to each sensor, but authorizes a mobile service team which manages the sensors, and the sensors obtain the membership from these members.

Moreover, cooperative institutions or cooperative research partners can be allowed to utilize the measure results as well as exchange particular measurements with MeteO members during their mutual project. They obtain the rights either directly from the MeteO center or from a cooperation station, eventually from individual scientists which head a project, and they become MeteO members during the project. Cooperative partners can also delegate the membership to their partners for the period of the project, these to their partners, etc. In our approach cooperative institutions are considered as subgroups.

A member from MeteO group or from MeteO subgroups, wanting to obtain direct information for example from a sensor, has to prove an authority, confirming it can do this, i.e. to prove its MeteO membership. For this, the member has to bring forward its certificate chain to prove its membership, the chain must start with the MeteO center. Generally, the chains of certificates are used not only to confirm membership but also to document who authorized the membership.

Our scheme is built upon distributed group membership management.

In this paper a new kind of certificate will be presented to reduce the time of the verification process of delegation chains. The verification time will be reduced by decreasing the number of expensive cryptographic operations. In Section 2, we will describe the distributed group membership management as well as related works. Section 3 will present the new kind of certificate. Section 4 will outline the role of the issuer of this certificate. In Section 5 the proposed scheme will be compared with previous results. Finally, the Section 6 gives conclusions and future work.

2. Distributed group membership management – State of the Art

This section is a short extract about the idea of distributed group membership management for ad-hoc networks from [AM01], [MAH00]. A group within group membership management is a set of members, persons or other physical or logical entities that may collectively be given access rights [AM01]. Some extraordinary members are leaders and have the right to make decisions about membership within the group.

Membership management is based on public-key certificates. A classical identity certificate (e.g. X.509 and PGP) binds a public key to the name of certificate owner and all the members in the group possess a unique name. This approach is identity-oriented. A contrary approach is key-oriented where each member is represented by its public key which is unique. Consequently, when a certificate is issued, it is issued directly to the public key of the member. A format of key-oriented certificates is used for example in the SPKI certificate theory [EFL+99], the main purpose of which is authorization rather than authentication and which defines a straight authorization mapping: authorization to a key. The SPKI certificate format is flexible. It is possible to include various contents as well as various rights into SPKI certificate.

The group membership management in [AM01] assumes that when a new group is established, a new key pair, the group key, is generated identifying the group. Each member of the group is identified by its respective public keys and obtains a certificate signed by the private key of the group key to certify membership of the group. Verification of the certificate is performed with the public group key. The group-key owner can certify either an ordinary member or a leader. The leader possesses the same authority as the group-key owner, i.e. the leader can certify a new member (Certificate 2, Fig. 1) as well as to appoint other leaders. A new leader obtains a leader certificate which was issued to its public key, the leader key. Leaders appointed directly by the group-key owner, are called top-level leaders (Certificate 1, Fig. 1). When a new member or leader is certified, it acquires its member/leader certificate as well as all certificates proving its status in the group starting with a certificate signed by the group key. The certificates create a chain of delegation certificates and a member can prove its membership by presenting its certificate chain and by using its own private key.

The verification process of delegation chain passes along the whole chain. A verifier has to check whether the first certificate in the chain was signed by the private group key, the second certificate by the private key of the key included in the first certificate, the third certificate by the private key of the key included in the second certificate, etc. Moreover, the verifier has to check whether each member in the certificate chain, except the last one, possesses the authority to delegate membership, in our case whether each member in the chain is a leader in the group. Also, the verifier has to compute a validity period of the chain as intersection of validity periods of all certificates in the chain. Such verification of a certificate chain is called *classical verification of a certificate chain*. The intersection of validity periods of *n* certificates is defined as: Let $X^i = (X^i_{min}, X^i_{max})$, $1 \leq i \leq n$, be validity dates of a certificate, where X_{min} is the not-before date and X_{max} is the not-after date. The date range intersection $V = \bigcap_{i=1..n} \{X^i\}$ is $V_{min} = \max_{i=1..n}(X^i_{min})$, $V_{max} = \min_{i=1..n}(X^i_{max})$. If $V_{min} > V_{max}$, then the intersection failed and the chain of certificates is not valid [EFL+99]. Moreover, expiration of a certificate in a chain signifies that all the following certificates in the chain are not necessary to be dealt with.

As mentioned above, groups may also have subgroups. A subgroup is a set of members and there is a sub-group leader with a sub-group key as well. The relation between groups which are called supergroup and subgroup is bound through a subgroup certificate. The subgroup certificate, as a member or leader certificate, contains a validity period, signature and group identifier and a subgroup identifier is added there. With such a certificate, it is possible to admit all the members of a group to another group. All the members of a subgroup are also members of the supergroup. A supergroup leader is the leader in all subgroups as well, however, a subgroup leader is not the leader in a supergroup automatically [MAH00].

Aims and related work

The membership management model uses chains of certificates. Within a tree structure formed by the certificates, the chains can become too long to be practical. The deeper the tree, the higher the verification costs. Moreover, if a delegation information in chains of certificates cannot be lost, each certificate of the chain is available for further audit, then it is not possible to use classical solutions such as reduction of chain of certificates. Furthermore, the reduction requires cooperation of the first key in the reduced part of the chain, which is not always possible. Considering a computational limitation of mobile devices or sensors an improvement of verification time would be very helpful.

The aim of this paper is to reduce the time of the verification process of delegation chains. Therefore the following requirements must be fulfilled:

- The solution need not lose delegation information included in chains of certificates.
- The solution should assume that the private group key could be erased after certain time (e.g. to prevent a compromise of this key).
- The solution should be adaptable in the mobile environment with size of hundreds or thousands of nodes.

Existing Solutions As mentioned earlier, a natural approach to improve the verification time is a reduction of the certificate chain. For example, an SPKI scheme uses a certificate result certificate (CRC) which is defined as a single certificate of computation, what the owner of a certificate chain is allowed to do [EFL+99]. Because the delegation information cannot be lost, the delegation chain, through which the member have obtained the membership, must be collected by the nodes that reduce this chain [AE00]. In our scenario, this can be attained by letting the members obtain redundant certificates directly from top-level leaders and use them instead of the original certificates given when the members first joined the group [MAH00]. This solution requires that each top-leader has to establish a database to store the information about each member, whose certificate was reduced and requires a search who and where is the respective top-level leader. Since the scenario is based on ad-hoc networks, the achievability of top-level leaders as well as the collection of data is insecure. Moreover, a sharing or a distribution of such databases is problematical.

By using nested certification and the corresponding subject verification methods [LC00], it is possible to have efficiently verifiable certificate paths. In this approach, a nested certificate guarantees correctness of another subject certificate. The subject verification method only compares a content in the nested certificate with a content in the subject certificate. The subject verification does not use a cryptographical operation, therefore the verification process is faster. The whole chain of certificates can be transformed to a chain of subject certificates requiring cryptographical operation only for the first certificate in chain. It is a timesaving solution. However, issuing, managing and storing one nested certificate for each certificate is a burden in the system.

Path validation in classical systems based on X.509 certificates and its improvements [LKS+01] require a centralized model and can only be used when a special server is available anytime. This server can give a variety of information about a certificate, a certificate chain or can afford a simple "Yes/No" statement about the certification path validity. However, it requires communication delay and the service is centralized.

The approach of Keoh and Lupu in [KL02] for group management in mobile ad-hoc networks is similar to PGP [Zim95]. It uses signed assertions which allow authorization decisions. Assertions are obtained from nodes in order to introduce a user's identity, membership in groups or other attributes. While in PGP the trustworthiness associated with a key determines to what extent the user is trusted for authentication, in [KL02], the trustworthiness associated with the key determines to what extent the user is trusted to sign assertions. Using of assertions from the nodes lessens the number of cryptographic verification operations. However, the trust policy expects a trust relationship between participating users. Also the PGP web-of-trust philosophy is different from our straight-delegating situation.

In comparison with our situation, all previous solutions require particular changes or different requirements in the scenario such as a strictly centralized server, high overhead of certificates, storing of reduced information with top-level leaders or different delegation policy.

3. Implanted Chain Certificate

In this section we show a new kind of certificate, its structure and the method of verification.

Description

Implanted Chain Certificates (IChC) are used to guarantee integrity and correctness of a chain of certificates. IChC can be imagined as a certificate for a chain of certificates.

For example, certificates 1 and 2 in Figure 1 are standard delegation certificates. Certificate 1 is issued by a group-key owner to delegate leadership in the group, Certificates 2 by another leader to delegate membership. Certificate 3 is issued by an IChC issuer to certify a whole chain of certificates. We can classify certificates in our paper as:

1 classical identity certificate certifies binding between public key and identity of its owner;
2 standard delegation certificate certifies the delegation of rights;
3 implanted chain certificate certifies a chain of certificates.

A chain of certificates must be verified to issue an IChC. In order to verify the certificate chain, the issuer of IChC needs the group public key. Since the group key is the group identifier, everyone doing business with the group will automatically know it. Moreover, the whole public group key could be included in each delegation certificate. After the issuer makes sure of integrity and legitimation of the chain, it implants the whole chain as content of IChC and signs over the IChC content with its digital signature.

Position and role of issuer of implanted chain certificate is given.

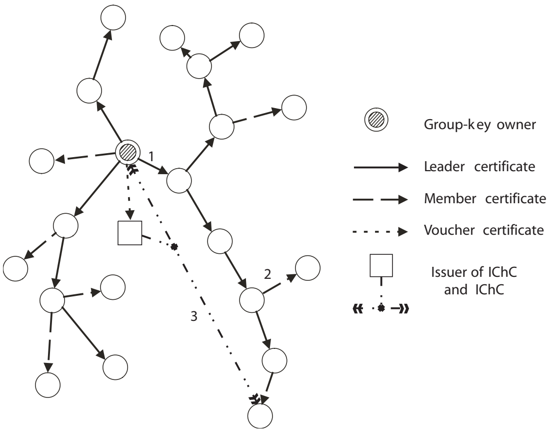

Figure 1. A group with members, leaders and an issuer of IChC

It should be noted, that an implanted chain certificate vouches correctness of a chain of certificates as well as that the first certificate in a certificate chain is signed by group key. However, an implanted chain certificate does not guarantee that several certificates in chain of certificates have not been revoked. In this way, an implanted chain certificate becomes independent of the revocation policy inside the group.

For example, in [AM01] the revocation policy relies on a propagation of the revocation list from member to member. Revocation data are signed only by a leader and the leader's chain of certificates needs to be attached. In [LL00] the revocation of nodes is based on Maintaining CRLs, where each user collects information about its neighbouring nodes. If a user's list contains k ($k > 0$) or more legitimate accusers of a node, the node is marked as convicted and determined as misbehaviouring. When using IChC, the verifier stays responsible for revocation control of certificates. But, in both previous policies of revocation, the verifier possesses the information about revocated nodes and it doesn't need to perform cryptographical operations to check the validity of nodes in IChC.

Structure of implanted chain certificate

An IChC is issued to a chain of certificates and it states that the chain of certificates is correct. An implanted chain certificate contains the following information:

- the group identifier
- chain of certificates
- a signature signed by the issuer of IChC.

The group identifier field is a group identifier of the IChC issuer, it doesn't matter how many subgroups are included in the chain. We describe the question of subgroups in Section 4.

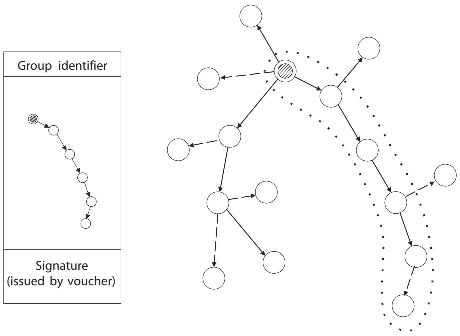

Figure 2. Structure of IChC

It is possible to add a validity period in an IChC optionally. The validity period of IChC depends on the shortest validity period of all certificates in the chain of certificates. The optional validity field in IChC makes it possible to detect directly if the chain of certificates is expired or not. Because of flexibility of our proposal, described in Section 4 – Subchains of IChC, we don't use this field as mandatory in the IChC.

Except the alternative restriction of validity, the lifetime of IChC is not limited. However, in case that the group key is changed, for example if the group is reconstructed, the validity of IChC expires.

The reconstruction of group key may be done periodically or when there has been enough changes in the group membership [MAH00].

Note that the size of the whole structure compared with the original size of the certificate chain increases only for one signature and group identifier, e.g. the group key.

Method of IChC verification

A verification process of a node certificate through implanted chain certificate consists of three steps. A verifier has to:

(i) check if the key of the verified node is included in the chain of certificates (the certificate containing the key can be placed anywhere in the chain, it need not be at the end)

(ii) verify the signature over the implanted chain certificate cryptographically

(iii) verify the certificate of the IChC issuer (more about IChC issuer in Section 4)

(iv) compute a validity period of IChC which is the intersection of all certificates in the chain from the beginning to the key of the verified node (it can be done automatically when the key of verified node is checked).

The correctness of the above mentioned four steps implies that the information given in the IChC is correct. The verification process requires only two cryptographic operations and is independent of the length of a chain.

When a member wants to prove its membership, it has to present the IChC certificate, attach the certificate of IChC issuer and proves its ownership of the key for example with a signed request.

Characteristics of IChC

Our new kind of certificate guarantees:

(i) integrity of all delegation certificates in the certificate chain, i.e. the content of certificate has not been accidentally or maliciously modified

(ii) all signatures of all certificates in the certificate chain are legitimated

(iii) all certificates in the chain were signed in right order respectively starting with the private group key.

It is the IChC issuer's goal to control the certificate chain, whether the chain is suitable for characteristics (i)-(iii). It will be done through a classical verification of the certificate chain, as mentioned in Section 4.

4. The Voucher – Issuer of Implanted Chain Certificate

For issuing the implanted chain certificate, a new special member of the group is needed. We call the IChC issuer a *voucher*. Its goal is to verify a chain of certificates and to issue a signed implanted chain certificate.

The voucher is a member of the group, it possess its membership certificate. A voucher certificate is a special form of member certificate. The status field in a membership certificate refers the three possibilities (member, leader, voucher). A voucher is established by a leader and it is on the leader's decision, when and why to establish a new voucher. For example, the leader can do this when the cost of verification of its delegation chain is too high or a high number of members occurres in the area requiring an optimization of the verification process.

For issuing a voucher certificate only one additional key pair is used, the private and public Common Voucher Key (CVK) and each voucher certificate is signed by the private CVK. In the beginning, the CVK is generated by a group-key owner, the group-key owner establishes several vouchers and signs their voucher certificates with the private CVK.

For managing vouchers and its keys, we compare two approaches which follow the philosophy of mobile networks and of distributed management.

A naive approach assumes each voucher possesses the secret key of CVK. The vouchers established by the group-key owner obtained the CVK directly from the group-key owner.

When a new voucher is established by a leader, voucher's own pair of public and private key is generated and the voucher obtains a voucher certificate and a certificate chain from the leader to prove its group membership. The process of voucher confirmation is not finished, because the voucher certificate is not signed by CVK, but by the leader's key.

To confirm its vouchership the voucher needs to contact an existing voucher. The existing voucher issues a new single voucher certificate, signed by CVK, to the voucher key. If we need to hold on the information who established a voucher, then the existing voucher has to issue an IChC to the voucher certificate chain signed by the private CVK. Finally, the existing voucher passes the private CVK to new voucher through secure channel (e.g. encrypted with the voucher's public key). Now, the new voucher is able to issue an IChC to other members and prove its authority by a single certificate as well as to confirm a vouchership to new vouchers. To sum up, a new voucher needs to reach only one existing voucher to certify its vouchership.

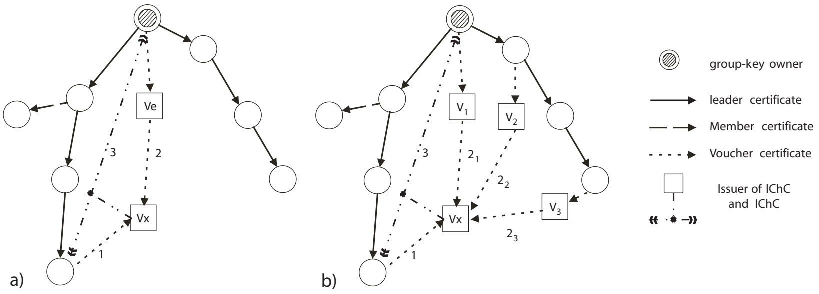

Figure 3. Establishing a new voucher

For instance, in Figure 3.a) a leader issued the voucher certificate 1 to a voucher V_x. Then the voucher V_x connected an existing voucher V_e and the new V_x voucher certificate was issued by V_e signing it with the private CVK. Next the voucher V_x is able to issue an IChC, the certificate 3, for a chain of certificates.

This naive approach is highly vulnerable, since an adversary only needs to compromise one voucher to acquire the private CVK. With the private CVK the adversary is able to break the whole voucher as well as IChC system.

The cooperative approach is based on threshold secret sharing and threshold multi-signature protocol. The concept of threshold secret sharing is to distribute secret information, in our case the private CVK, among *n* members through their secret shares. The aim is to allow any subset of *k* members to reconstruct the complete secret. This recovery of the secret is impossible for less than *k* members. Moreover, a new member can obtain its new secret share which is computed directly from *k* secret shares. In a threshold multi-signature protocol, *k* members posses their shares of the secret. They must cooperate to generate a valid signature of message M that can be verified by anyone using the appropriate public key, in our case the public CVK.

The structure of vouchers can be built in a satisfactory way by using a scheme described in [KZL+01]. The scheme is built upon Sharmir's threshold secret sharing [Sha79], established on Lagrange interpolation.

In the beginning, the group-key owner establishes at least *k* vouchers and signs their voucher certificates. Next the group-key owner distributes to vouchers their secret shares of CVK through a secure channel (e.g. encrypted with each voucher's public key). Thereafter, the group-key owner can erase the private CVK, because apart from the initialization phase, the CVK is never used in whole. Cooperation of *k* vouchers is needed to use the private CVK. Therefore, one possibility to compromise CVK is to compromise *k* vouchers and then to recover the whole private CVK.

Like in the naive approach, a new voucher has to be established by a leader, a voucher's own pair of public and private key is generated and the voucher obtains a voucher certificate and a certificate chain from the leader to prove its group membership. Next the voucher has to contact *k* existing vouchers to resign its voucher certificate (or to issue an IChC) and to obtain its new secret share. This is possible with a multi-signature protocol. For details of multi-signature protocol and computation of new secret share see [KZL+01].

When a member wants to prove its membership, it presents its implanted chain certificate and the voucher certificate. Consequently, every voucher needs a direct voucher certificate signed by the private CVK to prove its vouchership. The public CVK can be known for everyone like in case of group key, or the group-key owner can issue a leader certificate to this key.

For example, in Figure 3.b) a leader issued the voucher certificate 1 to a voucher V_x. Then the voucher V_x connected *k* (*k*=3) existing vouchers V_1, V_2, . . . , V_k and the new V_x voucher certificate 2 was signed by *k* partial secret shares of V_1, V_2, . . . , V_k, i.e. the voucher obtained a voucher certificate signed by private CVK. Next the voucher V_x is able to issue an IChC, the certificate 3, for a chain of certificates.

Implementation with large *k* can resist more powerful adversaries but the service availability degrades. Otherwise, small *k* increases the availability but the system is more vulnerable to attacks. Designing a key management framework that satisfies availability, vulnerability as well as fault tolerance is not an easy problem. Moreover, once *k* has been chosen and the system is deployed, it is expensive to change *k*. Therefore *k* depends on security policies within the system. A good value of *k* in our scheme is a question to be investigated by ad-hoc simulation e.g. by ns-2 network simulator [NS2]. But it is clear that no value of *k* will fit all requirements. The applicability of threshold schemes in ad-hoc networks has been shown, e.g. in [YK03] where a user has to reach at least k Mobile Certificate Authorities (MOCA) based on threshold cryptography to obtain a certificate service.

It must be emphasized that it is in the user's interest to obtain IChC from the voucher to reduce verification time of its certificate chain. Moreover, the IChC for the user is issued only once, but used several times.

Note that each leader could be a voucher as well, only the voucher certificate has to be issued by the leader to itself and consequently, the confirmation of vouchership has to be made.

Method of IChC issuing

If a voucher wants to issue an implanted chain certificate for a group member, the voucher must obtain the chain of certificates proving the membership of the member. After that the voucher executes:

(i) assurance that the first certificate in the certificate chain was signed by the group key

(ii) revision of integrity of all delegation certificates in the certificate chain through a hash algorithm and respectively public keys

(iii) revision of legitimation of all signatures over the several certificates in the certificate chain

(iv) revision of all certificates in the chain, whether they were signed in right order respectively

(v) revision of correct delegation of all the certificates in the chain, i.e. whether everyone in the certificate chain had the authority to delegate the rights.

For this, the voucher has to perform a classical verification of certificate chain. If the verification is successful, the voucher issues the IChC which guarantees correctness of the certificate chain. The IChC with a voucher's membership certificate is sent to the member. In short, the method of IChC issuing can be called a *pre-verification* of the certificate chain.

Revocation and compromise

If a voucher's own private key has been compromised or revokated, implanted chain certificates issued by this voucher are no longer valid. Considering that every IChC includes a whole chain of certificates, the classical verification of certificate chain is further feasible.

Subgroups and IChC

If a certificate chain consists certificates of two groups, a supergroup and a subgroup, i.e. one subgroup certificate is in the path, an implanted chain certificate could be issued, too. The IChC is issued by a voucher in a supergroup and a member uses the IChC when it wants to prove its supergroup membership (Fig. 4.a).

The role of a voucher is not transitive. A voucher of a subgroup does not automatically become a voucher in a supergroup. The subgroup voucher, es-

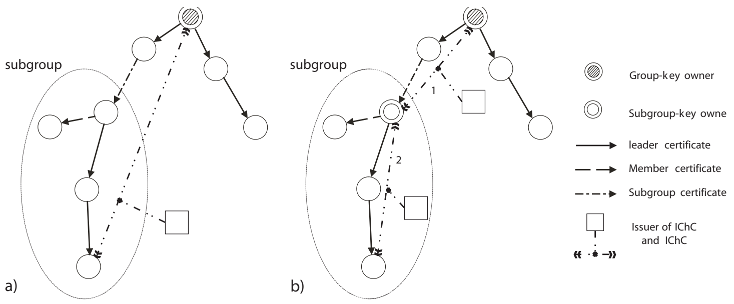

Figure 4. An IChC in a subgroup and a chain of IChCs

tablished for issuing of IChC in a subgroup, cannot issue an IChC for a member in a supergroup. Such an IChC is not trustworthy in supergroup, the certificate of subgroup voucher can not be verified in the supergroup because the supergroup has a different CVK. On the other hand, the same will happen with an IChC of a supergroup voucher in a subgroup. To solve this situation, a supergroup/subgroup voucher certificate can be issued to a subgroup/supergroup voucher.

Furthermore, a combination of IChCs is possible. If there is an IChC of the subgroup-key owner in a supergroup and an IChC in a subgroup is issued, the combination of these two IChCs is possible. Then we get a *chain of implanted chain certificates*. In Figure 4.b) an IChC chain consists of two IChCs, the IChC 1 in a supergroup and the IChC 2 in a subgroup. Using of chains of IChCs is a topic of our further work. Also the combination of a standard certificate and an IChC, e.g. when a leader possessing IChC delegates the voucher rights to itself, as well as the combination of several IChCs and IChC subchains, will be researched.

Subchains of IChC

An IChC is issued to guarantee correctness of chain of certificates from a group-key owner to a certified member. Moreover, there is a special behaviour of IChC: if IChC certified the correctness of the whole certificate chain, it certified the correctness of any subchain of the chain as well.

The behaviour could be used to reduce the amount of issued IChCs eliminating the time delay necessary to issue a new certificate.

Supposing that an IChC is issued to a bottom-level member, i.e. member, whose certificate chain includes plenty of members. Each of previous members included in a certificate chain of the bottom-level member consequently in the IChC, can use the IChC to prove their membership. When a bottom-level obtained an IChC from a voucher, it can offer the IChC to a previous leader, this leader to a previous leader, etc. Since no additional effort is required, members are likely to offer this service to another members. Remind that the method of IChC verification has been constructed in the way that the key of the certified member needs not be at the end of the certificate chain.

5. Results

In a full verification of a chain of certificates, the end users have to execute all the verification operations by themselves, which requires *O(n)* verification of certificates, where *n* is the number of nodes in the certificate chain and verification operations are expensive cryptographic operations. The reduction of a certificate chain reduces the whole chain to a single direct certificate, so computation by end users is decreased to *O(1)*.

In our proposal, cryptographic verification of an IChC is needed as well as cryptographic verification of either one voucher certificate if the CVK is known for everyone or two certificates, a leader certificate for the CVK and than a voucher certificate. The time necessary for passing along the chain and checking whether the public key of member is included in the chain is negligible in comparison with cryptographic operations. That implies that our solution requires *O(1)* verification of certificates and is comparable to reduction of certificate chain.

Note, that in the worst case, e.g. a compromise of the voucher system, there is always a possibility to use the classical verification of the certificate chain.

Remind, that the IChC is issued only once and that the size of the whole structure, compared with original the size of the certificate chain, increases only for one signature and one public key. Moreover, an IChC issued to a bottom-level member can be used for all previous members in a certificate chain. This is different to the method of reduction of certificate chain, where every user has to obtain a respective reduced certificate.

6. Conclusion and future work

In this paper we presented a way of improving the verification of a chain of certificates. Our approach is based on a new type of certificate called Implanted Chain Certificate. Using of an IChC is comparable to using a reduction of a certificate chain. Moreover, if an IChC is issued to a bottom-level member, the IChC can be used with previous members in a certificate chain as well. We described also the task and position of an IChC issuer, the voucher.

Implementing of this new certificate into infrastructures improves the efficiency of verification of delegation certificates when delegating rights among the members as well as when accelerating the verification process.

In future work, we will focus on IChC chains and intersections of such chains. We would also like to realize a prototype of an IChC based on SPKI certificate standard as well as to implement our scheme in an ad-hoc network simulator to present the efficiency of the application.

Acknowledgments

The authors would like to thank the members of the Munich Network Management (MNM) Team for helpful discussions and valuable comments on earlier drafts of this paper. The MNM Team directed by Prof. Dr. Heinz-Gerd Hegering is a group of researchers at the Ludwig Maximilian University Munich, the Munich University of Technology, and the Leibniz Supercomputing Center of the Bavarian Academy of Sciences. Its webserver is located at `http://wwwmnmteam.informatik.uni-muenchen.de`

The first author would like to acknowledge German Academic Exchange Service (DAAD) funding his study stay at Ludwig Maximilian University Munich.

References

[AE00] T. Aura, C. Ellison. "Privacy and Accountability in Certificate Systems." Research Report A61, Laboratory for Theoretical Computer Science, Helsinki University of Technology, Espoo, Finland, April 2000.

[AM01] T. Aura, S. Maki. "Towards a survivable security architecture for ad-hoc networks." In Proc. Security Protocols, 9th International Workshop, LNCS No. 2467, pp. 63-79, Cambridge, UK, April 2001

[EFL+99] C. Ellison, B. Franz, B. Lampson, R. Rivest, B. M. Thomas, T. Ylönen. "SPKI certificate theory." RFC 2693, IETF Network Working Group, September 1999.

[KL02] S. L. Keoh, E. Lupu. "Towards Flexible Credential Verification in Mobile Ad-hoc Networks." Proceedings of the Second SIGACT International Workshop on Principle of Mobile Computing (POMC 02), Toulouse, France, October 2002.

[KZL+01] J. Kong, P. Zerfos, H. Luo, S. Lu, L. Zhang. "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," International Conference on Network Protocols (ICNP), pp. 251–260, 2001.

[LC00] A. Levi, M. U. Caglayan. "An Efficient, Dynamic and Trust Preserving Public Key Infrastructure", Proceedings of 2000 IEEE Symposium on Security and Privacy, pp. 203–214, Oakland, CA, USA, May 2000.

[LKS+01] B. Lee, K. Kim, M. Seo, W. Huh. "Efficient Offline Path Validation", First International Workshop for Asian Public Key Infrastructure (IWAP2001), pp. 117-125, October 2001

[LL00] H. Luo, S. Lu. "Ubiquitous and Robust Authentication Services for Ad Hoc Wireless Networks." Technical Report TR-200030, Dept. of Computer Science, UCLA, 2000.

[MAH00] S. Maki, T. Aura, M. Hietalahti. "Robust Membership Management for Ad-hoc Groups." Proceedings of the 5th Nordic Workshop on Secure IT Systems (NORDSEC 2000).

[NS2] The Network Simulator – ns-2. Available at http://www.isi.edu/nsnam/ns/.

[Sha79] A. Shamir. "How to Share a Secret." Communications of the ACM, 22(11), pp. 612-613, November 1979.

[YK03] S. Yi, R. Kravets. "MOCA: Mobile Certificate Authority for Wireless Ad Hoc Networks", 2nd Annual PKI Research Workshop Program (PKI 03), Gaithersburg, Maryland, April, 2003.

[Zim95] P. R. Zimmermann, "The Official PGP User's Guide", MIT Press, 1995