

1 INTRODUCTION

Collaboration, as the act of working jointly, is present in many aspects of every day life. Most prevalent examples of collaboration in our lives include the activities in an industrial-scale workplace and team sports. In both these examples, groups of people work jointly towards the accomplishment of a well-defined goal. The operation of a group that collaborates towards a well-defined goal is called a *collaborative session*. In collaborative sessions members of the group follow certain rules (e.g. in an enterprise workplace collaboration, junior employees do not access top-secret company documents, and in football team members other than the goal keeper do not touch the ball with their hands). Moreover, externals to the group do not participate directly to the accomplishment of the goal of the collaboration (e.g. the owner of the restaurant opposite to the company's building is not allowed to access or contribute to the development of the upcoming product of the company, and the supporters of a football team are not allowed in the court during the game).

To ensure that the rules which govern a collaborative session are respected, a number of controls are associated with the operation of the corresponding group. One example of such controls is the case of a company that requires from its employees to carry their access badges whenever they enter the company building and to report to the reception the presence of a person in the company premises who does not carry an access badge. Another example of rules enforcement is the case of the referee in a football game who is responsible to supervise the game, spot violations of the game's rules and punish them according to the book.

Inevitably, collaboration has also received attention in computer systems and cooperative activities supported by computers have been a subject of research for the past two decades. However, the variety of means, tools and infrastructures that have been proposed for collaboration support are based on the assumptions of a central control point and a continuous connection of participants throughout the duration of a collaborative session. These assumptions do not apply in the case of *nomadic* collaborative sessions. Traditionally, the characteristics of nomadic computing include independence of location, motion, computing platform, communication device and bandwidth, and widespread access to remote systems and services (e.g. see [1]). For the purposes of this paper, the term *nomadic* is refined to refer to the lack of presence of a central control entity during a collaborative session and the uncertainty of services available in the physical context where the collaborative session happens. Examples of nomadic collaborative sessions are a working meeting regarding some international cooperation which takes place in a hotel's conference room and an ad hoc gaming session among pupils in the school bus.

This paper outlines our early work on the infrastructure support for nomadic collaboration management (or NCM for short). The prototype implementation of our approach assumes that a collaborative session takes place over a Bluetooth network. However, the design of the infrastructure support is not specific to Bluetooth and it can be applied to a variety of short range wireless connectivity means including the ANSI/IEEE Std 802.11 suite. Our approach is not based on IP security; it relies only on link level security for

the identification of the devices that are allowed in a collaborative session and for the establishment of a secure network among them. Based on the link level security, our approach provides end-to-end application level security for the participants of the collaborative session. More precisely, the security properties that are provided by the suggested infrastructure for nomadic collaboration management are:

- Authentication of the collaborative session participants
- Confidentiality of sensitive, application specific assets
- Integrity of sensitive, application specific assets

In brief, a candidate participant *prepares* his participation to the collaborative session by contacting the admission manager for that session. The admission manager is a well-known, trusted entity which authenticates the candidate and processes his request to participate to a given session. Provided that participation permission is granted, the candidate receives an admission ticket secured by a PKI system between the candidate and the admission manager. The admission ticket contains a session certificate which will be used by the candidate to enter the given session plus a number of encryption keys which will be used to secure the communication of sensitive data, classified in various access groups. The admission ticket contains the keys for the access groups to which the admitted candidate has access rights. The admission ticket may, or may not be time-bound depending on the policies associated to a given collaborative session.

The admitted candidate can exercise his right to join a session by contacting one participant of the session. After exchanging their session certificates for mutual authentication purposes, the admitted candidate becomes a participant of the session. Link level security mechanisms are used to ensure that the network links created for the purpose of a given session do not include any other entity except the authenticated participants. The protection of the sensitive application assets is ensured by classifying them into access groups and using the designated key for encrypting them before communicating them to other session members. This way, without exposing the access rights of session participants, we enable confidentiality of sensitive application assets.

The proposed infrastructure support for nomadic collaboration management (NCM) automates the supervision of a session security-wise. The end-user has to take only the following actions in order to enable the correct operation of the infrastructure:

- Provide his public key to the admission manager
- Specify the public keys of the other session participants
- Specify the access groups to which each sensitive application asset belongs

The remainder of this paper is structured as follows: a couple of generic scenarios exemplifying the utility of our approach and a concrete application used for demonstration purposes are presented in section 2. Section 3 presents the system model in terms of the assumptions made, the system architecture and the role of each identified architectural entity. Section 4 elaborates on the infrastructure support for the management of nomadic collaborative sessions. The paper concludes in section 5 with a summary of the presented approach, its current status and its future considerations, and a short discussion on its strong points and its limitations.

2 EXAMPLE SCENARIOS

In order to better comprehend the needs which fostered our work on the infrastructure support for nomadic collaboration management (NCM) presented in this paper, we few scenarios which show, from the end-user perspective, the benefits that our approach bears.

2.1 MEETING SCENARIO

A number of European enterprises form a cooperation initiative to promote their products and market goals. In the context of this cooperation, representatives from each partner company will meet for a kickoff meeting where they will present the plans of their company and they will try to create a technical/business roadmap for their cooperation. In order to facilitate the travel arrangements, the partners choose a convenient destination in Europe, easily reachable by all partners and they book a conference room in a hotel placed close to the airport. The common practice with such conference rooms is that they do not provide any facilities to support this kind of meetings, except maybe a video beamer which participants can connect to their laptops and use it to make their presentations. This falls far short from what they company representatives need for their meeting. To state the minimum, a number of files containing the presentations need to be distributed among the participants, let alone the files containing company brochures and the minutes of the meeting. The distribution of this material among the participants can happen either on the site using floppy disks and memory cards, or after the meeting by means of email. In any case, the sender will have to check the content of the material he is about to send out and verify that confidential information is not communicated to unintended recipients.

Using NCM the whole process is greatly simplified. When fixing the date and place for the meeting, the partners request an admission ticket from the NCM admission manager which may reside, for example, on a specific page of the web-site of the meeting organizer. Since the meeting date and duration is known in advance, the admission tickets provided for that meeting can be time bound. We assume that a partner from company X, in his request for an admission ticket, specifies three access groups: company-X private, company-X restricted and public. Assets classified in the first access group must not be accessed by participants of the meeting that do not work for company X, assets classified in the second access group can be accessed by other participants that work for affiliated companies and, finally, assets in the third group can be accessed by all participants. In principle, the partner who specifies these three groups, has also to specify who belongs to the company-X restricted group since the admission manager does not have any a priori knowledge about which companies are affiliated to company-X. For the sake of simplicity in this paper we do not address the issue of access group specification; rather, we assume that the admission manager possesses, or is able to retrieve, this information.

Using the NCM admission manager, all the partners who are invited to the kickoff meeting receive their admission tickets. The date of the meeting, the invited partners arrive at the agreed place with their laptops equipped with a wireless network card (e.g. a Bluetooth or a WLAN card). When the meeting starts, every partner has to provide to

the NCM running on his machines the public keys of the rest of the participants. This can be done either manually, i.e. by typing in or copy-and-pasting the keys to the NCM GUI or by selecting the participants from the NCM address book which contains their public keys. It is also fairly easy to automate this exchange of public keys by a separate utility which exchange electronic business cards. After this step, the NCM takes over and creates an interconnection network among the session participants (see §4.2 for more details). Provided that each participant has already classified the assets he is willing to share in this meeting into one of the three access group mentioned above (i.e. private, restricted and public) he can proceed by sharing them with the other participants without having to worry about sensitive information reaching the wrong hands.

2.2 GAMING SCENARIO

The software house which markets a certain game for portable devices with remote connectivity capabilities (either wired or wireless) for multi-player gaming, offers also the possibility to create teams, i.e. two or more players can form a team and play against other teams. Members of the same team may share game assets (e.g. energy pills, weapons, spells, game level keys, etc). Gaming sessions may happen at any place (e.g. some recreation room, in the school bus, at a cafeteria, etc). All players have to do is to connect to the software house web-site and request for an admission ticket for such sessions and choose the team in which they want to play, or create a new team.

In this case, the NCM admission manager resides at the web-site of the software house which provides the service of multi-player sessions for a given game. The admission ticket returned from the admission manager contains a certificate used to identify the player as participant to a given gaming session and, depending on the marketing policies of the company, may or may not be time bound. A player who receives such a ticket can use it in a recreation room to play a game with other players. When entering the room, the player exchange his public key with the other players present in the same room (similarly to the previous example, this step can be automated). Without having an explicit knowledge or interaction with the other players, the recently arrived player starts sharing game assets with the rest of the team to which he belongs. The NCM establishes the communication links with the rest of the game participants and ensures that the sharing of game assets happens strictly within the limits of formed teams.

2.3 COLLABORATIVE DRAWING SCENARIO

As a more concrete example of NCM employment consider we consider the case of a collaborative drawing application. The application consists of a shared canvas on which session participants can draw shapes like boxes, lines and text items and they can also associate comments to them. The creator of a shape is its owner and the one who specifies who of the other participants are allowed to view the comments associated to that shape. One end-user is responsible for organizing a collaborative drawing session by inviting a number of other end-users and communicating to the admission manager the list of invitees. All invitees must request from the admission manager an admission ticket and specify which other invitees belong to the trusted group which is allowed to see their comments that are not publicly visible.

The request of the admission ticket and its reception by the invitees happens in separate steps. First, in a registration phase, the invitees must contact the admission manager and request the ticket before a predefined time interval from the beginning of the collaborative session. When this time interval is passed, the registration phase is closed and those that have already requested a ticket must contact the admission manager again to receive it. This assures that the admission manager has complete knowledge of the session participants in order to create the appropriate number of keys for the exchange of application sensitive assets (i.e. the comments that are not publicly visible). Since the session takes place on a give date and time and for a predefined duration, the admission ticket is time bound to this information. End-users who have not been invited (they are not in the invitee list communicated by the session organizer to the admission manager) or have not registered in time do not receive an admission ticket when they contact the admission manager.

When the collaborative drawing session takes place, the invitees first exchange their public keys and then their certificates in order to mutually authenticate each other. Then, all information created during the session (i.e. shapes and comments) is multicast to the group of participants; shapes and publicly visible comments are sent without encryption and comments which are not publicly visible are encrypted with the owner's key. Only the members of the trusted group of participants for the given end-user have the key to decrypt this information and hence to view these comments.

3 SYSTEM MODEL

The security properties provided by NCM, namely authentication of session participants and confidentiality and integrity of sensitive application assets, are based on a fundamental assumption we make in this paper: the end-user and his operation on a portable device are non-separable. If this assumption does not hold, then it is possible that a legitimate candidate to a session receives on a portable device a valid admission ticket but the device and the admission ticket are used by another end-user who is not entitled to enter the given session. This assumption is not restrictive in common practice since portable devices are either personal devices like mobile phones and Personal Digital Assistants or PDAs (hence not shared among end-users) or they have enough resources and power to employ a user access mechanism and protect the assets stored on the device by non-legitimate users.

Another assumption we made is the existence of an underlying software layer which provides link level security, i.e. the establishment of an interconnection network where only authorized devices participate and can communicate in a secure way with respect to non-authorized device that may listen to the same physical bearer (e.g. devices connected to the same Ethernet or devices in the same Bluetooth neighborhood). Neither this assumption is restrictive for NCM since most wireless protocols provide link level security (e.g. see Bluetooth security in [6]) and there exist well-established approaches for achieving additional link security guarantees (e.g. see [2]). Also, we assume that the interaction between the admission manager and the candidates is authenticated and it guarantees confidentiality (e.g. by means of PKI services).

Finally, our last assumption regards the application and end-user involvement in NCM. In brief, the end-user or the application must provide to the local NCM the public keys of all other entities with which it will interact (i.e. admission manager and other candidates). In addition, the application must instruct the NCM about the sensitivity of the assets that will be communicated to other session participants, in order to NCM to employ the appropriate encryption.

3.1 SYSTEM ARCHITECTURE

Based on the above assumptions, a clear, three-layered system architecture is outlined. At the bottom, the network layer providing connectivity among session participants and link security guarantees forms the foundation of the system architecture. Above it, the NCM layer ensures end-to-end application security in terms of authentication of session participants and confidentiality and integrity of sensitive application assets. At the same level, though in a different container that lies outside the scope of NCM, a typical set of PKI services support NCM. On top of the ICM layer, the application layer contains the front-end of the application as well as the interface with the ICM layer which is responsible to specify the different groups of sensitive application assets and the assets themselves and to supply NCM with the public keys of the entities with which a portable device will interact in the context of a collaborative session (i.e. the admission manager and the devices representing the other session participants). Figure 1 provides a graphical illustration of this system architecture.

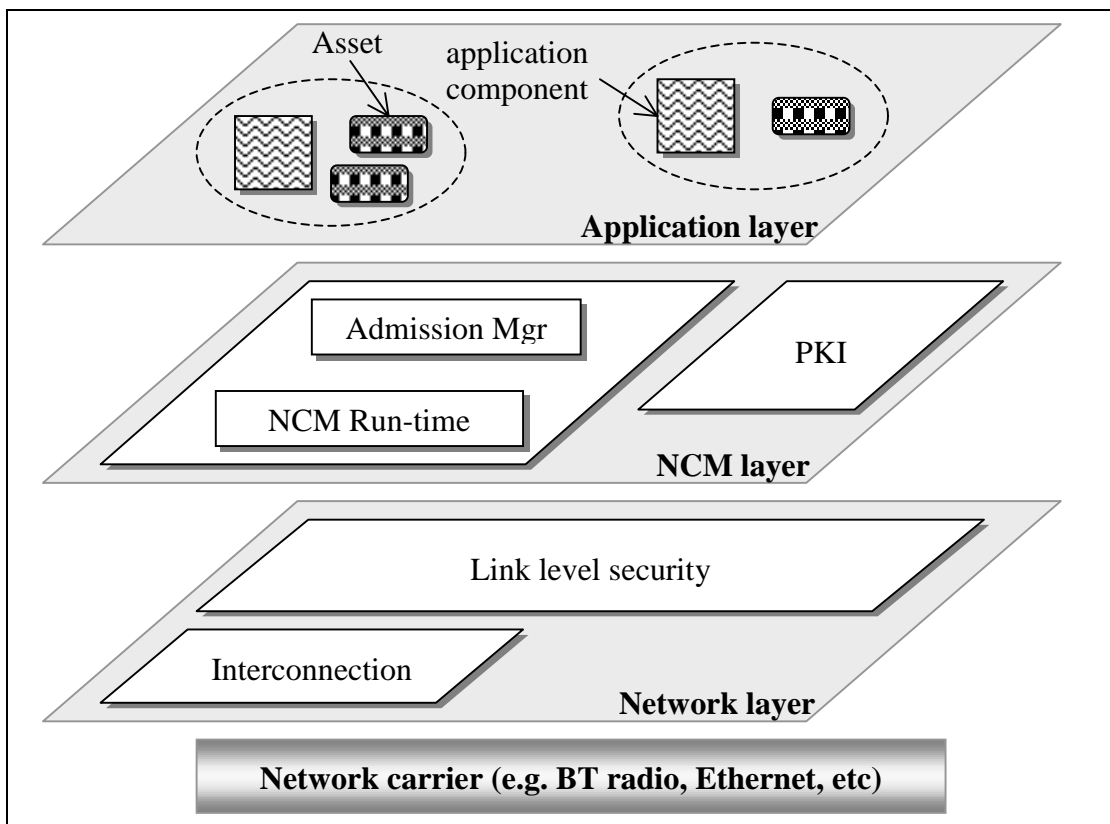


Figure 1 The system architecture of the NCM.

3.2 NOMADIC COLLABORATION INFRASTRUCTURE

The management of the collaborative session identifies three types of entities: the admission manager, the session participants and third parties which are not allowed in a session. As previously explained, provided that a typical set of PKI services is in place, the management of the collaborative session takes place in two steps. First, a third entity must request the right to enter a given session from the admission manager (step 1a in Figure 2). At this point, the third entity is labeled as *candidate*. If the right to enter the session is granted, the admission manager replies with an admission ticket (step 1b in Figure 2) and the candidate participant becomes a *legitimate participant*. A legitimate participant can exercise the right to enter a given session (steps 2, 2' and 2'' in Figure 2) and thus to become an *active participant* or simply *participant*.

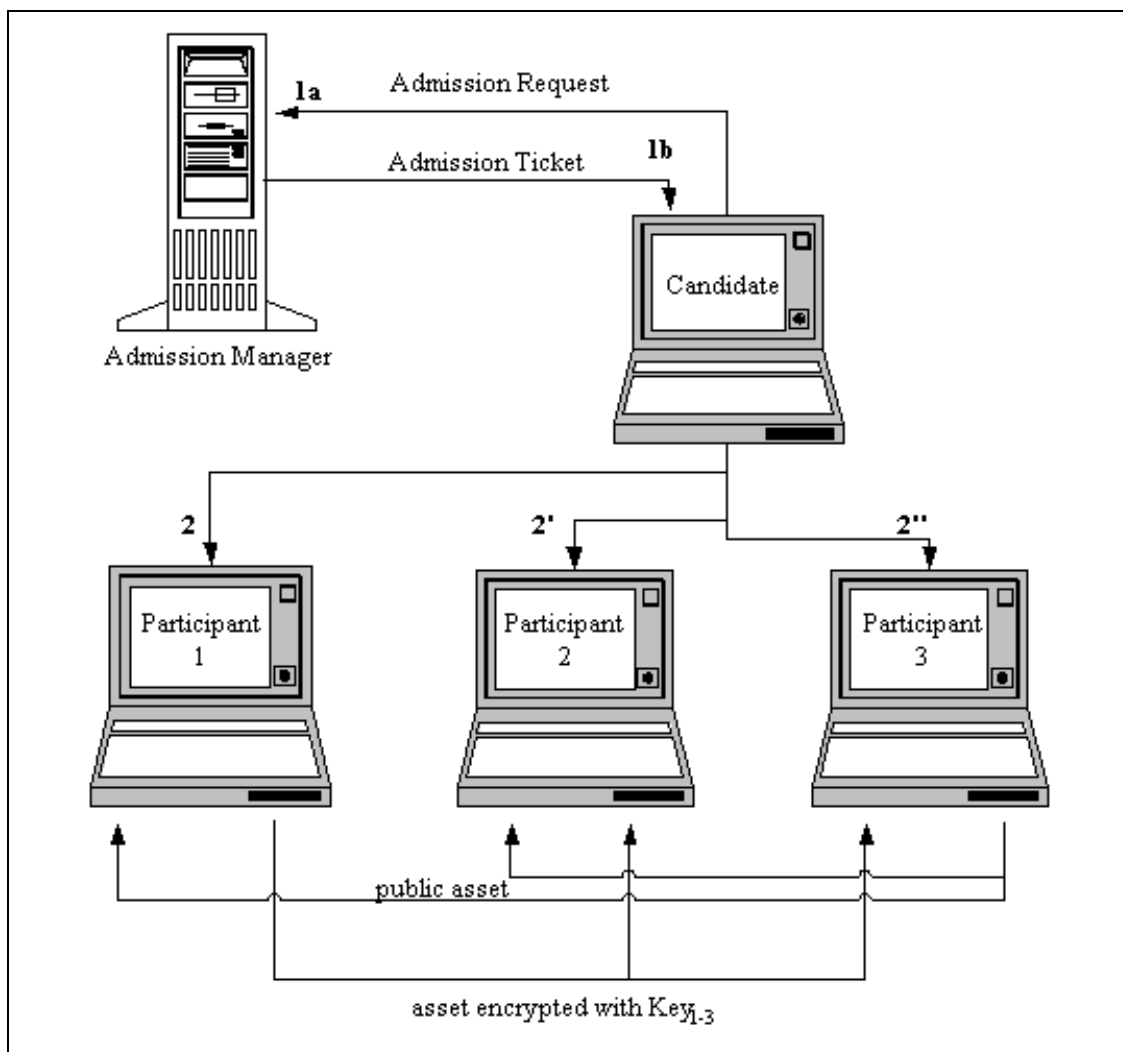


Figure 2 An illustrative example of NCM in action.

The first step of this process necessitates an authentication mechanism between the candidate and the admission manager and communication confidentiality guarantees, which can be anything from a Kerberos system coupled with a private keys encryption scheme

to a PGP-based PKI scheme. Also the connection between the candidate and the admission manager may or may not be secured by software means, as long as the confidentiality of the admission ticket is guaranteed (e.g. the interaction with the admission manager may happen only by physically plugging the personal device to the computer hosting the admission manager by RS232 or USB cable). In the remainder of this paper we are not addressing the security of this interaction since it does not belong per se to the category of collaborative session interactions.

Notice that the action of entering a session (steps 2, 2' and 2'' in Figure 2) requires the interaction of the entering party with all the current participants of the given session. The legitimate participant which is about to enter the session and the active participant which the former contacts exchange the session certificates which they have previously encrypted using the key assigned by the admission manager for this purpose. This is in order to ensure the confidentiality of the certificate they exchange. Unless both parties have received a valid admission ticket, they will not be able to mutually authenticate each other and the entrance of the legitimate participant to the session fails. This should not happen unless one of the two parties has not received an admission ticket from the admission manager and attempts to enter the given session illegally.

The brute-force way for completing the entrance of a legitimate participant to a given session is the following. Initially the entering party identifies one of the participants and after their mutual authentication it receives from the latter the network addresses of all the other participants of the session in order to proceed by performing a mutual authentication with each one of them. In practice, under the assumption of distributed trust (e.g. see [3]) among the participants of the session, only the first authentication process is necessary while the authentication with the rest of the session participants can be reduced to the communication of the network address of the newly joint participant by the session participant which has authenticated him.

Once the participant has completed his entrance in the session, he can communicate to the other participant of the session all application assets he wishes to share provided that he has previously specified to NCM the access group to which each asset belongs. For non-public assets, NCM is responsible for using the designated key (which was part of the admission ticket) to encrypt them before multicasting them to other participants. Only participants that belong to the specified access group possess the key to decrypt the sensitive application assets. For example, in Figure 2 the application assets communicated by participant 1 to participants 2 and 3 belong to the access group called 1-3 and which contain only participants 1 and 3 as members. Hence, from the two recipients of this message, only participant 3 has the key to decrypt the content of the message and to access the contained application asset. This fact matches the intentions of the sender and confirms the claim that NCM provides communication confidentiality and integrity guarantees based on access rights control.

4 COLLABORATIVE SESSION MANAGEMENT

After the quick outline of NCM in terms of design assumptions, system architecture and infrastructure support, this section provides an elaborated description of the operations that take place for managing a collaborative session using NCM.

4.1 SESSION PREPARATION

The first part of the collaborative session management is the preparation of the conditions that will ensure the secure interactions during the actual collaborative session, and includes the following steps

- The specification of the admission manager for a given session
- The update of the policies at the admission manager site
- The requests from admission to the given session by the candidates, and
- The admission manager verdict for each request which, if positive, is translated to an admission ticket.

The first two steps are not always necessary. For example, in the gaming scenario presented in §2.2 it is highly probable that the admission manager has a set of predefined policies (e.g. regarding the teams that can be formed and hence the different access groups allows in a collaborative session) as well as access to the database of players who have bought the service of multi-player gaming and are entitled to participate to gaming sessions. In other cases, the first two steps must be taken when there is an indication about a collaborative session happening in the future. For example, in the meeting scenario presented in §2.1 when the organizer of the meeting sends the invitation to the invitees list, he has the responsibility specify in the invitation the address of the admission manager. It is also his responsibility to update the admission policies to the session which, besides the IDs of the candidates which should be admitted, also includes the policies regarding the access groups that should be created. The latter may vary from a predefined list of access groups and potential participant pre-assigned to them to a flexible scheme where each candidate can define his own access groups and assign other potential participants to those. In any case, the admission manager is not able to provide admission tickets before it has a complete picture of access groups that the admitted candidates have specified.

The latter two steps sketched above (i.e. admission request to the manager and ticket returned to the admitted candidates) are always carried out for the session preparation. Let's assume for simplicity purposes that a PKI scheme exists and is used by the admission manager and the candidates (the former possesses the latter's public keys and vice versa). Using the PKI to ensure the confidentiality of the communication, each candidate authenticates itself with the admission manager (e.g. by sending its credentials) and then issues a request for an admission ticket. When the admission manager receives an admission request check with its policies and decides whether or not to admit the candidate. If the candidate is not admitted, he received the rejection immediately. However, if the candidate is admitted, it will receive the admission ticket only after the admission manager has the complete information regarding the access groups that may exist in a given session. This is necessary in order to create pairs of encryption keys for each group which will be used to ensure the access right control on application assets that will be assigned by the participant in different access groups. In addition to these key pairs, one for each access group, the admission manager creates one key pair common for all participants which is used for mutual authentication purposes and a session certificate common for all participants too.

The last step, i.e. admission ticket sent to the admitted candidates, is created separately for each admitted candidate in the following way. First the session certificate is packed, followed by the common session keys used for mutual authentication purposes. Then, for each of the access group that the admitted candidate is a member, the group ID followed by pair of keys for access control purposes in that group is packed. The structure of the admission ticket is graphically illustrated in Figure 3. Finally, the admission ticket for each admitted candidate is encrypted using the public key of the given candidate and it is subsequently sent to him.

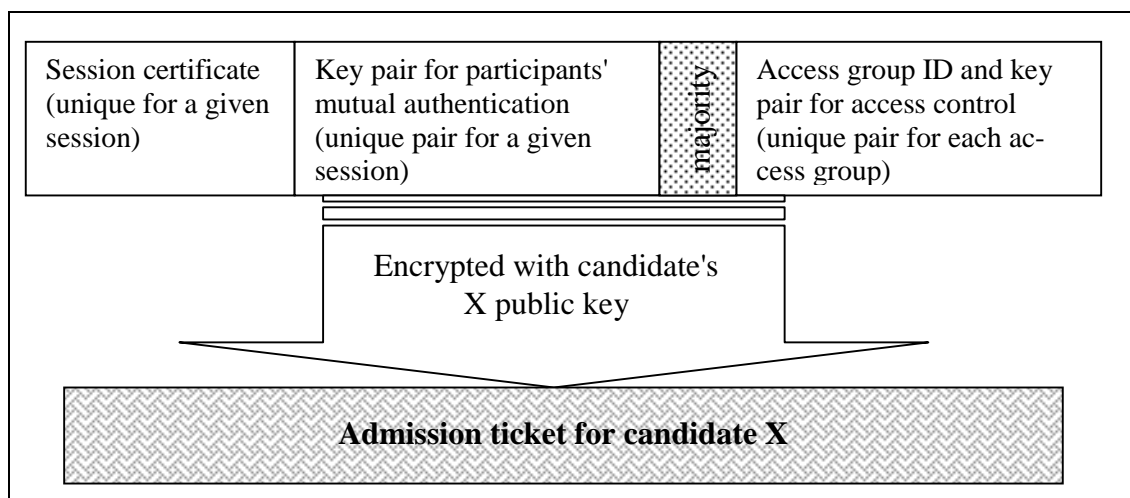


Figure 3 The structure of the admission ticket.

4.2 SESSION CREATION

The second part of the collaborative session management is the creation of a single working group where session participants interact with each other. Ideally, a single group is created by the first two legitimate participants that come together and legitimate participants that arrive subsequently enter the already created collaborative session in the way outlined in §3.2 where the support provided by the nomadic collaboration infrastructure is described. Basically, this ideal case consists of the following steps:

- The legitimate participant is mutually authenticated with an active participant X of the session
- In case of a distributed trust scheme in the collaborative session, other participants trust participant X and his judgment, and start communicating with the newcomer after receiving by X the latter's address.
- In the absence of a distributed trust scheme, the newcomer receives from X the addresses of all other active participants which he contacts one by one and performs the same mutual authentication process in order to become a fully integrated active participant himself.

In practice however, it is not very unlikely that distinct sets of legitimate participants will come together simultaneously and they will create disjoint working groups for the same collaborative session. Then, provided that the collaborative session is physically confined to the same interconnection network (e.g. the same Bluetooth neighborhood of

Bluetooth neighborhoods that belong to the same scatternet) member of such disjoint groups will eventually contact each other. At that point, the disjoint groups should merge with ultimate goal the creation of a single group of participants for a given collaborative session.

NCM provides a straightforward mechanism for merging disjoint groups and providing support for application level consistency after the merging, which is based on the a priori knowledge of the number of participants in a given session (mechanism clearly does not apply in certain cases, e.g. ad hoc multi-player gaming sessions). It is possible to include in the admission ticket, the number of member that a group of participants must contain before the collaborative session can be activated. When this number is set to the absolute majority of the participants that are admitted in a given session, it is guaranteed that the session can be active only in a single group of participants. Hence, other groups of participants will not be able to produce any application data (collaborative session is not active in these groups) before they merge into a group which collectively contains the majority of admitted session participants.

In any case, the creation of a collaborative session bears at minimum the overhead of establishing network connections among participants and performing mutual authentication at least once for each participant (in case of a distributed trust scheme among session participants). This initialization overhead is the price participants have to pay in order to ensure that only admitted parties are present in the interconnection network of a given collaborative session.

4.3 SESSION OPERATION

Once a collaborative session is active, NCM ensures a predefined access control policy on sensitive application assets which are specifically assigned to one of the access groups known to the admission manager. NCM provides a network abstraction for every participant, which invokes NCM methods to communicate with other session participants. When invoking the NCM methods, the participant specifies the application asset to be communicated to the session as well as the access group to which the given asset belongs. Based on this information, NCM creates the message to be sent to session participants as follows:

- If the access group is "public" then the asset is not encrypted. The message to be communicated to the session consists of the ID of the "public" group followed by the plain asset and it is multicast to all the participants of the given collaborative session.
- If the access group is other than public, then NCM retrieves from the admission ticket the encryption key for that group and uses it to encrypt the application asset. The message to be communicated to the session consists of the ID of the specific group followed by the encrypted asset and it is multicast to all the participants of the given collaborative session.

When receiving a message, NCM first reads the ID of the group to which the message belongs and tries to match it with the group IDs it has received with the admission ticket. If no match is found then the given participant does not belong to the access group which qualifies the received message and has no rights to access the application

asset included in the message. Hence, NCM will immediately discard the message. On the other hand, if NCM is maliciously treated at the receiving end in order to read application assets to which the receiving participant has no access rights, the lack of the access group key to decrypt the data will prevent it from doing so.

By default, the communication in collaborative sessions is penalized with the overhead of multicasting (individual message transmission to each recipient in the multicast group), unless an efficient multicasting mechanism is put in place at the network level. NCM is no exception to this rule. However, messages that contain application assets allowed to be accessed only by a small subset of the session participant come with the same multicast penalty. This is the price that NCM has to pay for ensuring access control without requiring the exchange of access rights credentials among the session participants. Finally, in addition to the multicast overhead associated with every communication in a collaborative session, the communication of sensitive application data has to pay the penalty of encryption and decryption of data, which may increase considerably the duration of end-to-end communication of application data depending on the size of the encryption key and the resources available at the participant's device.

4.4 SESSION TERMINATION

The termination of a session does not present any particular interest from the NCM viewpoint. A participant may quit a session without notifying any other participant. This will have an impact to the communication time until the interconnection network realizes the change in its state and adjust to the new configuration. However, the sudden disappearance of a participant does not place a threat to the security guarantees provided by NCM.

5 CONCLUSION

This paper presented an approach to the management of nomadic collaborations, where the term is defined to signify collaborative sessions which take place without the need of a central control point to ensure the correct operation of the session. The focus of the paper has been the presentation of NCM, a software infrastructure support for the admission control and access rights management in nomadic collaborations. The presented approach is based on the separation of the collaboration management in two phases: the admission to a given session after a request issued to the designated admission manager for that session, and the enforcement of the access rights on sensitive application access as they have been communicated to the admission manager and they have been set on each application by its owner. The cornerstone of the presented approach is the use a session certification and a number of key pairs used for the mutual authentication of the participants and for the enforcement of the access rights policies. The security guarantees offered by NCM are authentication of session participants and confidentiality and integrity of sensitive application data. These guarantees are provided to the applications using NCM under the following assumptions: the end-user and his operation on the portable device that host a session participant are non-separable, and there exists a mechanism that provides authentication between the admission manager and each candidate and confidentiality of their communications. Other issues besides security con-

cerns which relate to the management of nomadic collaborations have been addressed in our previous work [5].

Among the strong points of NCM is the fact that manages the operation of nomadic collaborations in a completely distributed way (the admission to a session is not part of the operation of a collaborative session; rather it is part of its preparation). Another strong point of our approach is that it does not necessitate knowledge about other participants access rights, as long as the admission server has enough information to create key pairs for all access group in a given session and distribute them among admitted candidates. On the other hand, NCM does not provide any better solution regarding the communication overhead in collaborative sessions. In addition, it penalizes the end-to-end application level communication in the case of transferring sensitive application assets by the overhead of their encryption at the sender side and decryption at the receiver side. Compared to techniques and mechanisms well-established in the field of computer security for ensuring authentication and communication confidentiality and integrity (e.g. see [7]), our approach does not contribute any innovation. However, its originality lies in the employment of simple and comprehensive security techniques for providing security guarantees for nomadic collaborations in a completely distributed way and without the need to share any credentials regarding access rights of a session participants.

In the context of the VIVIAN project [4], we have developed NCM as a service specific to the domain of collaborative applications on top of an implementation of the wireless CORBA specifications over Bluetooth. However, NCM itself is developed as a COBRA service and hence it is not specific to the wireless CORBA implementation over Bluetooth; we have performed a few experiments of NCM on a traditional IIOP-based ORB which ran on top of a WLAN network . However, the GIOP engine in the wireless CORBA implementation on top of Bluetooth has much smaller overhead than the IIOP engine, which provides a more sensitive environment for measuring the overhead caused by the encryption activities and for assessing the size of the encryption keys that is practically useful with today's technology. The front-end of the NCM demonstrator is an implementation of the collaborative drawing scenario described in §2.3. The integration of NCM with the middleware support for mobile collaboration [5] and the collaborative drawing application into a full fledged demonstrator for the VIVIAN project is expected to finish by the end of the summer 2002. By that time, NCM will be extended to provide protection against denial of service attacks which would prevent the smooth operation of a participant in a collaborative session.

6 REFERENCES

- [1] R. Bagrodia, W. W. Chu, L. Kleinrock and G. Popek. *Vision, Issues, and Architecture for Nomadic Computing*. IEEE Personal Communications, 2(6):14-27, December 1995.
- [2] A. Aziz and W. Diffie. *Privacy and Authentication for Wireless Local Area Networks*. IEEE Personal Communications, 1(1):25-31, 1st Quarter 1994.
- [3] L. Kagal, T. Finin and A. Joshi. *Trust-Based Security in Pervasive Computing Environments*. IEEE Computer, 34(12):154-157, December 2001.

- [4] The VIVIAN Consortium. *Opening Mobile Platforms for the Development of Component-based Applications*. VIVIAN¹ / ITEA² 99040.
- [5] P. Rust, G. Ferrari and T. Saridakis. Architecting Mobile Collaboration at the Middleware Level. In the *Proceedings of the 3rd International Conference on Information Reuse and Integration*, November 2001.
- [6] Bluetooth. *Specification of the Bluetooth System (v1.1)*. 2001.
- [7] D. Gollmann. *Computer Security*. John Wiley & Sons, 1999.

¹ www-nrc.nokia.com/Vivian

² www.itea-office.org