User Authentication in Multidatabase Systems

Eyk Hildebrandt Gunter Saake
ITI, Otto-von-Guericke-Universität Magdeburg
E-mail: {eykhilde|saake}@iti.cs.uni-magdeburg.de

Abstract

The aspect of security needs more consideration in the area of architectures for multidatabase systems. Particularly, the authentication of users which is a main prerequisite for a successful authorization is not considered sufficiently in current architectures. Due to the autonomy and heterogeneity of the component database systems, the problem of authentication in multidatabase systems is more complex than in traditional database systems. In this paper, we discuss the foundations and prerequisites for architectures of authentication in multidatabase systems. We present several approaches with respect to different degrees of autonomy and heterogeneity. Especially, we work out the authentication process and show the advantages compared with related approaches.

1. Introduction

Multidatabase systems (MDBS) provide access to heterogeneous and autonomous data sources [12]. In the literature, several architectures were presented dependent on the coupling mode of the participating systems and the area of application. These architectures describe the tasks and realization of commonly used components and the cooperation between the participating systems. But, there is no uniform architecture for MDBS because of the different requirements from applications.

Security of information systems proves as an important point because of the growing needs for secrecy and privacy. Therefore, security components are added to existing architectures or new architectures are developed around a security concept. The existing approaches for MDBS security systems are dedicated to: the search for the optimal global security model [9], the design of a federated security system [14] and access control and authorization specification [4]. However, in these projects the aspect of user authentication was not considered in detail. Only [6] provides three authentication schemata which are the starting basis of our work.

The correct authentication of users is an important task for database security in general because it is the necessary prerequisite for access control [11]. Despite that it is often solved through operating systems functionality or special hardware components, the database system is closely related to the task. The problem is more complex in multidatabase systems where three factors determine the area. First, the participating systems are autonomous to different degrees, which leads to different security requirements. Second, there can be many grades of heterogeneity because of the many existing authentication mechanisms. Third, in a MDBS the same user may have different identities and identifiers but has to be handled uniformly.

Before we present the approaches we discuss the necessary foundations and prerequisites including a user concept and a running example. For these approaches we work out the authentication process and discuss realization aspects. Also, we give a comparison with related work like authentication in distributed systems.

2. Background

In this section we work out the necessary background for the scope of this paper. Therefore, we give brief introductions to architectures of MDBS and to the foundations of user authentication in centralized systems.

Multidatabase systems are cooperations of autonomous and heterogeneous databases. They share information and functionality to an exactly determined degree and allow a more or less uniform access [8]. We distinguish two architectures which are representative for the whole spectrum.

Interoperating database systems cooperate without a global layer but with the full enforcement of autonomy. The participating systems communicate with each other through database adapters to overcome syntactic heterogeneity. A user can post access wishes from one system to all others.

In contrast, *federated database systems* (FDBS) use a global layer, the federated database management system (FDBMS), with integrated schemata to provide a uniform access and to overcome semantic heterogeneity. In FDBS the autonomy of the local systems can be restricted.

Figure 1 shows the principal but abstracted configurations for both architectures.

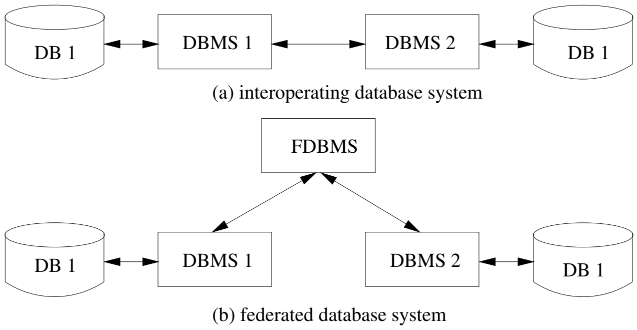

Figure 1. Multidatabase architectures

Before we describe user authentication in centralized systems, we introduce some often used notions:

Users are natural persons with the principal right to access a database system.

Identity is the unique name a user is attached to the system.

Identifier is an attribute that can be uniquely assigned to a determined user and proves his identity via the system.

User profile contains all information about a user.

Identification is the assertion of a user to possess a determined identity.

Authentication is the process of proving that a user possesses the asserted identity.

For centralized database system two kinds of authentication components can be distinguished [2]. First, the task is done by the operating system or a hardware component outside the control of the DBMS with accepted and trustworthy mechanisms. Hereby, the database system has no access to user identifiers and shares the response for security with the operating system. Second, the database system realizes and controls an own component for this task. This allows an easier coordination with the other parts of access control.

Finally, we present mechanisms for user authentication which are employed for or in actual database systems:

Password–based systems: A user is identified by a string known only by himself and the system. This by no means trustworthy but simple to realize method is used in the majority of commercial database systems.

Hardware–based systems: By these methods information (e.g. on a magnetic card) owned by a user or characteristics (e.g. a fingerprint) of a user is used to prove the identity. They have a high trustworthiness, but they are also expensive to realize. They are often employed in security–critical or military information systems.

3. Preliminary Considerations

In this section we firstly turn towards the problems and tasks of authentication in MDBS. After that, we introduce a user concept and present some policies about the granting and the withdrawal of identities. We conclude this section with the introduction of a running example.

The authentication of users in multidatabase systems is more complex than in traditional database systems. The main reasons and the resulting tasks are:

Heterogeneity: The local authentication components can base on the variety explained in section 2. As a result users have to pass through all these different procedures to gain access. Besides, the identity of a user can vary from system to system. The task is to overcome this heterogeneity without a decrease of security. That means, each user should be authenticated once but correct to all relevant participating systems per session.

Autonomy: The local authentication components decide dependent on the delivery of the correct identifier about access wishes. The maintenance of autonomy is necessary to secure the trust of the local systems.

Population control: A user can operate in an MDBS with different identities and identifiers. But, a user should be handled as a single subject independent with which identity he logs in. The authentication of users without local identities has to be solved in some environments.

The heterogeneity and the autonomy vary in the different application environments. Resulting, we need several approaches to handle this process.

Now we introduce a simple user concept. We assume that each local system associates each of its users with exact one identity and one identifier. We do not consider the authentication of user groups and roles or processes. In federated systems, the FDBMS grants a so-called *global identity* to each of its users. In MDBS we can recognize the following three classes of users:

- local users with one identity per affiliated system,
- global users with only a global identity,
- federated users with local and global identities.

The following events cause dynamic transitions between these classes:

- acquisition of a global identity,
- acquisition of one or more local identities,
- dispossession or return of local or global identities.

The granting and withdrawal of identities for users regulates the principal access to a determined system. This is the task of the responsible administrators of the system. In multidatabase systems we need therefore some coordinated policies to prevent unauthorized access. For the granting of global identities we present the following principles:

- The FDBMS grants after a check a global identity to each local user who have to use the federated system.
- The FDBMS grants a global identity to persons with no access to local systems if they fulfil at least one of the following conditions:
  - prospect to receive one or more local identities,
  - access to global data should be granted,
  - access to determined local data should be granted without the required local identity.

The last condition can occur if the enrolment of new users in a local system is prevented by a limited capacity or a determined policy. We discuss different realization possibilities for this with the corresponding approaches. Otherwise, a local identity should be granted to a global user, if there are enough information about him and the necessary trust. The withdrawal of a relevant identity should be propagated to all other participating systems where the user belongs.

Finally, we present a running example to elucidate the authentication processes in the particular approaches. Also, this example simplifies the possibility to compare the different architectures. We assume the following facts:

- There are two database management systems $DBMS_1$ and $DBMS_2$ which directly or indirectly control (via the operating system) components LA_1 and LA_2 for the authentication of their users.
- There is an FDBMS with or without an authentication component GA. It possesses the identity *FDBMS* and the identifier **** in both local systems.
- A user with the name *John Doe* possesses access to $DBMS_1$ with the identity *John* and the identifier *** (assumed as secret password) and also to $DBMS_2$ with *Doe* and **. In stated cases he possesses the global identity *John Doe* and the identifier *****.

4. Direct authentication

This approach bases upon the fact that in some MDBS the enforcement of autonomy is more important than the overcoming of the heterogeneity. The main reasons are:

- high local autonomy and security requirements,
- low trust between the participating systems,
- invincible heterogeneity, e.g., through local hardware based authentication components.

Therefore, each user has to be authenticated by all participating systems he wishes to access.

The first architecture for this approach has no global authentication component. The users operate only with their local identities. For the realization a simple program is necessary on every possible login system to establish a connection to the authentication components of the other systems. However, the user must also direct log on at all systems where a hardware based component is used. Association tables for the different local identities of the users can be used to handle them as single subjects. Therefore, the local systems have to share the relevant information with the others or must propagate them to the global system. Figure 2 shows our example with two interoperating databases,

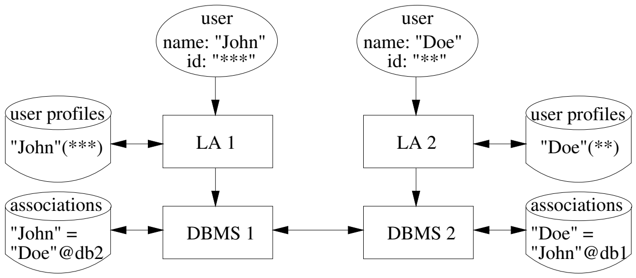

Figure 2. Direct authentication I

where the user directly connects to both. Altogether, this architecture is suitable for pure interoperable systems with or without global mediators [1].

The second architecture uses a global component to authenticate the global identity of the users. The task of full population control can be done with the help of association tables. With it, each user is attached to his other *protection subjects* (groups, roles etc.). In figure 3 we show the arrangement of the components adequate to our example. Moreover, we subsequently show the positive passing authentication process for our federated user:

1. user: global identification (*John Doe*, *****)
2. GA: authentication of the global identity through comparison with the stored user profiles
3. user: selection of the local systems to be accessed (DBS_1 and DBS_2)
4. FDBMS: connecting to the selected systems
5. user: local identification (*John*, ***) and (*Doe*, **)
6. LA_1 and LA_2: authentication of the local identities,
7. $DBMS_1$ and $DBMS_2$: notification to the FDBMS
8. FDBMS: granting of access to the user; association of local and global identities; activation of the relevant protection subjects

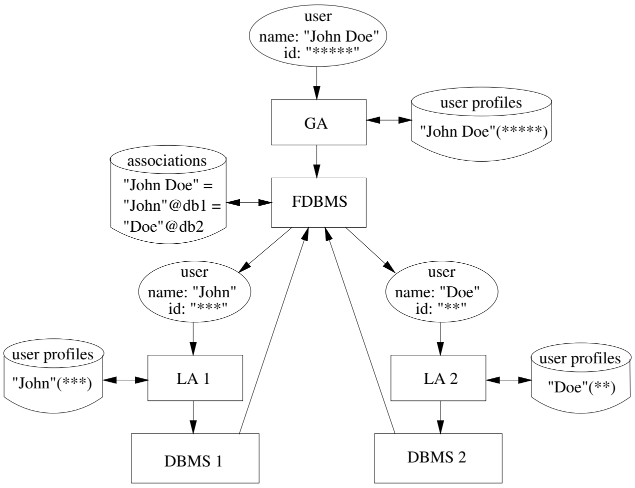

Figure 3. Direct authentication II

This architecture allows that a user logs in first at a local system and receives then access to the others through the global component. However, a user should not be allowed to operate simultaneous at different participating systems without global authentication. Beside a successful passing process we recognize the following negative decisions of the participating components and their consequences:

- incorrect global identity or identifier $\Longrightarrow$ no access to the multidatabase system
- incorrect local identity or identifier $\Longrightarrow$ no access to the local system, global access limitations
- no correct subject association possible $\Longrightarrow$ global and local access limitations

The limitations and the eventual exclusion of users should be defined in a coordinated policy by the global and local administrators. They have also the task to maintain the correctness of the association table. In this approach global users can only receive local access through the acquisition of local identities.

This approach respects the full autonomy of the local systems. The security requirements are guaranteed by the local authentication components. A simple password–based system and the implementation and protection of the association tables are sufficient to realize the global component. The remaining heterogeneity with the resulting complex process for the users and the limitations for global users are the main disadvantages. This approach is suitable for all kinds of MDBS where the explained reasons are valid.

5. Indirect Authentication

With this second approach we introduce an architecture that is more practicable for the users than the direct approach. Hereby, the central point is to overcome the heterogeneity. Therefore, we take some restrictions of the local autonomy into account. The main idea is to deliver the relevant user information for the local authentication indirectly from a special component and not directly from the user.

Foremost, we present this solution for interoperating databases, where only local to local connections are concerned. There are two changes opposite to the analogous direct approach described in section 4. First, we do not store only the relevant identities in the association table of each participating system but also the identifiers. Second, with each interoperating session a user starts or with each query he submits to another system the stored information is delivered to authenticate him at the participating systems. This leads to a configuration shown in figure 4, where a hexagon represents a delivered message. Apart from the

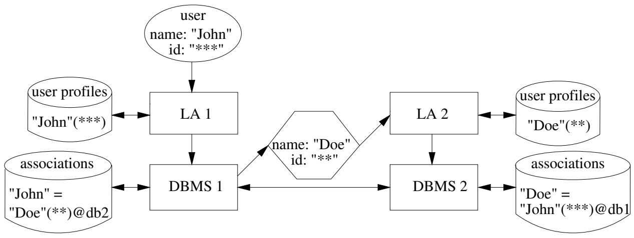

Figure 4. Indirect authentication I

fact that this is only suitable if the identifiers are software–based, the main problem is to secure the trust between the systems to enable the mutual storing of identifiers. Therefore, the participating systems should support the same security standards. In particular, each system has to ensure that the identifiers are only released for authentication. The employment of cryptographic protocols is also necessary.

Now, we show the application of this approach for federated databases. Here we have again a component to authenticate the global identity. As part of the FDBMS there is also a management component that regulates the access to the association table and executes the delivery of authentication information as shown in figure 5. Some remarks to the implications of this approach follows:

- The global management of local identifiers needs the trust of the local systems.
- The propagation of local identifiers and their delivery for local authentication leads to high requirements for a secure data transfer.
- The association of identities needs the agreement of the user and of the relevant local administrators.

This approach offers the possibility to grant access to global users without local identities. Therefore, the local systems provide some identities to the FDBMS which are

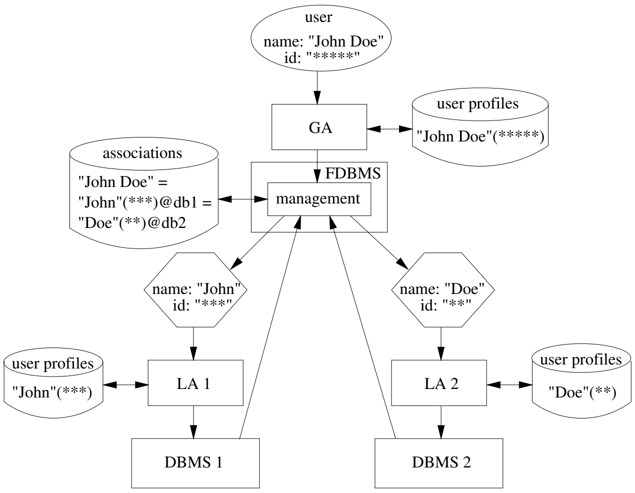

Figure 5. Indirect authentication II

attached with different but exactly defined access rights. The FDBMS can now associate these identities with global users without releasing them. So, global users receive limited but no direct access to local data. Nevertheless, the local administrators have the right to participate in the decision which user receives access. There are two possibilities to ensure this. First, the local administrators could establish a list with persons they want not as users which the FDBMS has to follow. Second, the FDBMS gets the local approval for each decision.

We now present a policy for the association of local and global user identities and the access of these informations. Such a policy is needed to prevent that the identifiers are used for prohibited access to the local systems. Therefore, we state the following principles:

- The association of user identities is done jointly by administrators from the relevant local and the global system regarding the approval of the user. This holds also for updates and removals.
- The propagation of local user profiles as well as updates and deletions are the task of the local administrators. The global ones receive the right to read identities but not identifiers. The same holds for the global user profiles but conversely.
- The transmission of a user profile for local authentication has to be strongly bound to the global authentication process and needs the agreement of the user.

The main advantages of this approach are the practicability for the users, the reduction of the heterogeneity and the possibilities for the FDBMS to grant access to users without local identities. The prizes are a loss of autonomy and high efforts for realization and communication. Some remarks for this aspect are to follow:

- The component for the authentication of global identities should consist of mechanisms which are reliable and trustworthy for the local systems. Therefore, it should support standards that are at least as high as the ones of the local components.
- The management component has to assure that the association tables are only accessed accordingly to the presented policy. The implementation of the component with the interfaces to the local and global authentication components and administrators has to follow high security and quality standards. Cryptographic mechanisms should be used for the storing and the transmission of identifiers.
- To prevent possible misuse by the administrators some of the information integrity principles [3, 10] should be installed. For this case that means, that for all necessary management actions there are *wellformed transactions* and no administrator can deliver local identifiers without the users agreement.

The primary field of application for this approach are tightly coupled FDBS where a certain degree of trust is available.

6. Global Authentication

In this last approach one participating system takes the full control on the entire process. In a federated system this is the task of the FDBMS. This system authenticates the global (or local) identities of the users. All other systems grant special local identities and identifiers to this system which will be authenticated at determined time points. These can range from once at the beginning of a connection till each query dependent on the security requirements. Hereby, we distinguish two cases dependent whether the local user profiles are used for the final decision or not.

In the first case we consider the local access rights of the federated users. Therefore, we realize either an association table like in the direct approach or a global management component like in the indirect approach. We have the following possibilities to secure the trust of the other systems:

- If there is no invincible heterogeneity the global component can be required to direct authenticate all relevant identities and identifiers of a user.
- An independent dedicated authentication server takes the task of the authentication for all users in the multidatabase environment. The components of the participating systems have to be dissolved.

In figure 6 we show a configuration with the first possibility.

In the second and opposite case we renounce the local user profiles. The controlling system decides according to

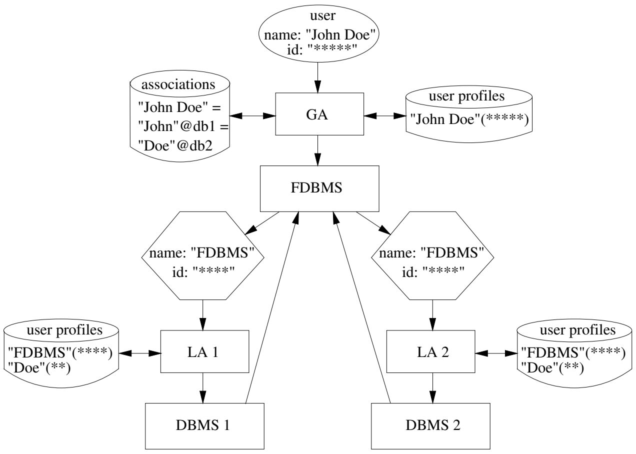

Figure 6. Global authentication

its own judgement about a users access wish. This judgement should be based on the authorization policy of the MDBS. The following reasons with different consequences can lead to the application of this case in federated systems:

- There are no local authentication components. A local login of the FDBMS is not required.
- There are no federated users. The authentication process is strictly separated into global authentication for global users and local authentication for local users.

It is obvious that this approach with its different variants is suitable for environments with special requirements.

7. Conclusions

In this paper we have shown that user authentication in MDBS is more complex than in traditional systems. Nevertheless, we have presented several approaches to resolve the explained problems with respect to different architectures and requirements. As a foundation we introduced a user concept for MDBS as well as a policy for the granting and withdrawal of identities in such systems.

The presented approaches are developed on the basis of authentication schemes for FDBS from [6]. Therefore, we want to discuss some advancements and advantages of our approaches. The first advancement was to expand the approaches also to the scope of interoperating database systems. Second, we developed them to full architectures and made some necessary enhancements. The indirect approach requires in opposite to the related schema (with medium autonomy) that local identifiers are delivered and authenticated. Third, our approaches respect different degrees of autonomy and heterogeneity. Additionally, they provide also some possibilities to grant access to global users without local identities. We also showed some mechanisms to secure the necessary trust from local systems in a global authentication component. Last, we gave some hints for the possible realization of the approaches.

Alternatively, multidatabase systems could use authentication mechanisms provided for distributed systems [7]. Independent of whether a authentication server like Kerberos [13] or a credential system [5] is used, the main advantages are the full overcoming of heterogeneity and the existence of accepted and fit for service components. But on the other hand, not many MDBS can tolerate the therefore necessary restrictions regarding autonomy.

References

[1] K. S. Candan, S. Jajodia, and V. S. Subrahmanian. Secure Mediated Databases. In S. Y. W. Su, editor, *Proc. of the 12th IEEE Int. Conf. on Data Engineering ICDE'96*, pages 28–37. IEEE Computer Society Press, 1996.

[2] S. Castano, M. G. Fugini, G. Martella, and P. Samarati. *Database Security*. ACM Press, Addison-Wesley, 1995.

[3] D. D. Clark and D. R. Wilson. A Comparison of Commercial and Military Computer Security Policies. In *Proc. of the 1987 IEEE Symp. on Security and Privacy, Washington D.C.*, pages 184–194. IEEE Computer Society Press, 1987.

[4] S. De Capitani di Vimercati and P. Samarati. Authorization specification and enforcement in federated database systems. *Journal of Computer Security*, 5(2):155–188, 1997.

[5] V. E. Jones, N. Ching, and M. Winslett. Credentials for Privacy and Interoperation. In *Proc. of the 1995 New Security Paradigms Workshop*. IEEE Computer Society Press, 1995.

[6] D. Jonscher and K. Dittrich. An Approach For Building Secure Database Federations. In J. B. Bocca, M. Jarke, and C. Zaniolo, editors, *Proc. of the 20th Int. Conf. on VLDB*, pages 24–35. Morgan Kaufmann Publishers, 1994.

[7] B. Lampson, M. Abadi, M. Burrows, and E. Wobber. Authentication in Distributed Systems: Theory and Practice. *ACM Transactions on Comp. Systems*, 10(4):265–310, 1992.

[8] W. Litwin and M. C. Shan. *Introduction to Interoperable Multidatabase Systems*. Prentice Hall, 1994.

[9] G. Pernul. Canonical Security Modeling for Federated Databases. In D. K. Hsiao, E. J. Neuhold, and R. Sacks-Davis, editors, *Interoperable Database Systems (DS-5)*, pages 207–222. North-Holland, 1993.

[10] R. Sandhu and S. Jajodia. Integrity Mechanisms in Database Management Systems. In *Proc. of 13th NIST–NCSC National Computer Security Conference*, pages 526–540, 1990.

[11] R. Sandhu and P. Samarati. Authentication, access control and audit. *ACM Computing Surveys*, 28(1):241–243, 1996.

[12] A. P. Sheth and J. A. Larson. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3):183–236, 1990.

[13] J. G. Steiner, C. B. Neumann, and J. I. Schiller. Kerberos: An Authentication Service for Open Network Systems. In *Winter 1988 USENIX Conference*, pages 191–201, 1988.

[14] Z. Tari and G. Fernandez. Security Enforcement in the DOK Federated Database System. In P. Samarati and R. Sandhu, editors, *Database Security, X: Status and Prospects*, pages 23–42. Chapman&Hall, London, 1997.