Research and Implementation of a Four-Dimensional Access Control Model

YunQi CHEN

China Coal Technology Engineering Group Chongqing Research Institute, Chongqing, China

chenqee@yeah.net

Keywords: management information system; permission management; data access control; role-based access control; metadata; function view

Abstract. Based on the Role Based Access Control (RBAC) model, this paper presents a Four-Dimensional Access Control (FDAC) model which contains four scopes: user, role, function and metadata. The FDAC model realizes the permission control of function and data, and it completes the whole process through meta control and function view. This new model is in favor of modular development and design, and reduces the repetition of coding and complexity of its business logic. It is applicable to most information systems under multiple platforms and has better scalability.

Introduction

In recent years, with the rapid development of information technology and the Internet, management information systems have widely used in various industries which has not only given great convenience of people's life and work, but also promoted the continuous progress of the whole society. Meanwhile, due to the increasing requirements and the changing of software development technology, users of management information systems are more and more, its range involves more widely, the scale and complexity of those systems are growing too, the security has got much more attention. Therefore, how to ensure the security of the management information system while offering friendly operating experience and taking into account features of it, such as resources openness, sharing and so on, is a hot topic in current [1].

Traditional access control policies were varied, such as operating system level that you could get all operations of the software after logged on to the operating system. Some software that you could use it after input correct username and password, and the other systems had simple different user groups, like administrator and general user while the scope of permission are different [2]. All of the above, the granularity of those access control strategies had obvious shortcomings, such as coarse granularity, complex processing logic, difficult maintenance and etc.

Role-Based Access Control (RBAC) model was first proposed by David Ferraiolo and Rick Kuhn in 1992 [3]. And then, Ravi Sandhu put forward the most famous RBAC96 model with his colleagues. Role hierarchy and assign constraints were added to the improved model [4]. The RBAC model has been widely used because it realized the separation of user and permission that makes the access control to become more flexible.

RBAC Model and its shortages

The basic principle of RBAC and its improved models is the presented concepts of Role and Permission. In those models, some permissions have assigned to different roles, and then user could get those permissions who played those roles, it could operate all functions including in those permissions. Those models contains four elements: user, role, permission and session, as shown in the Fig. 1.

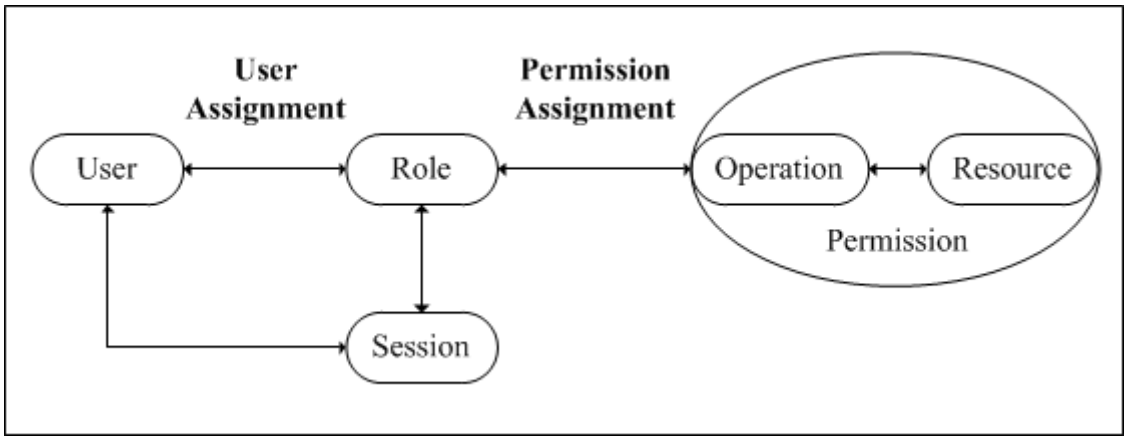

Fig. 1 RBAC Model

Permission refers that some resources are allowed to operate, such as buttons, pages, menus, forms and so on, and operation means some actions, like add, edit, delete, query and retrieve. User who wants to get those permissions must login to information system and then initializes the session which stores roles and their permission list of the current user.

The RBAC model is well realized the separation of user and its rights with the help of role highlighted as the principal part. A user's role determines its rights, and it is consistent with the real world, so it has much applicability. And then, the RBAC model is widely used in various types of C/S and B/S systems because it is easy to implement through many different programming languages. Finally, this model is in favor of permission assignment, both roles of user and role's permissions could modify after the system released, so it is conducive to usage and maintenance than ever before.

However, as the progress of information system and the evolution of development mode, limitations in the RBAC model are becoming more and more obvious in the following two aspects:

The first element is lack of the standard of granularity of permission, the definition of resource and operation are too broader then lead to increase the difficulty of system design and implementation. For example, a page contains many buttons, and those buttons will assign to different roles, this condition should cause confusion and get complicated logic and worse maintenance.

Another disadvantage of the model is lack of data access control strategy. In fact, controlling of data is also a key process in rights management, because of the restriction of functions is aimed at limiting the data scope that user could operate. In most cases, if you restrict the range of data that user can access, and naturally also limits the permissions.

Four-dimensional access control model and its framework

Based on the RBAC model, we get an extract definition of function, and makes a clear division of right size, and then propose a four-dimensional access control which consists of four scopes: metadata, user, role and function. It not only achieves to functional authorization but also realizes to control data rights. The new model could reduce the coupling of information system and promote the flexibility and maintainability of it.

Metadata and meta control. Metadata is the data that describes other data. It is the foundation of data constituting information systems, and it is the main basis to achieve access control. Common metadata includes public basic data, such as area information, department information, various types' data, range data, etc. The most familiar forms of metadata are data collection and data list.

For example, a national exam registration management information system, level of examination, examinees' grade, as well as the provinces, cities and other regional information are all belong to metadata. If it is different permission levels, different operational metadata purview. For instance, the range of an administrator of Ministry of Education authorized should be a collection of all the country's regional information including all provinces and cities while a province-level

administrator can only be limited to browse the data under the jurisdiction of the province. There is a big distinction between the data scope of the two administrators. The difference is illustrated in Fig. 2.

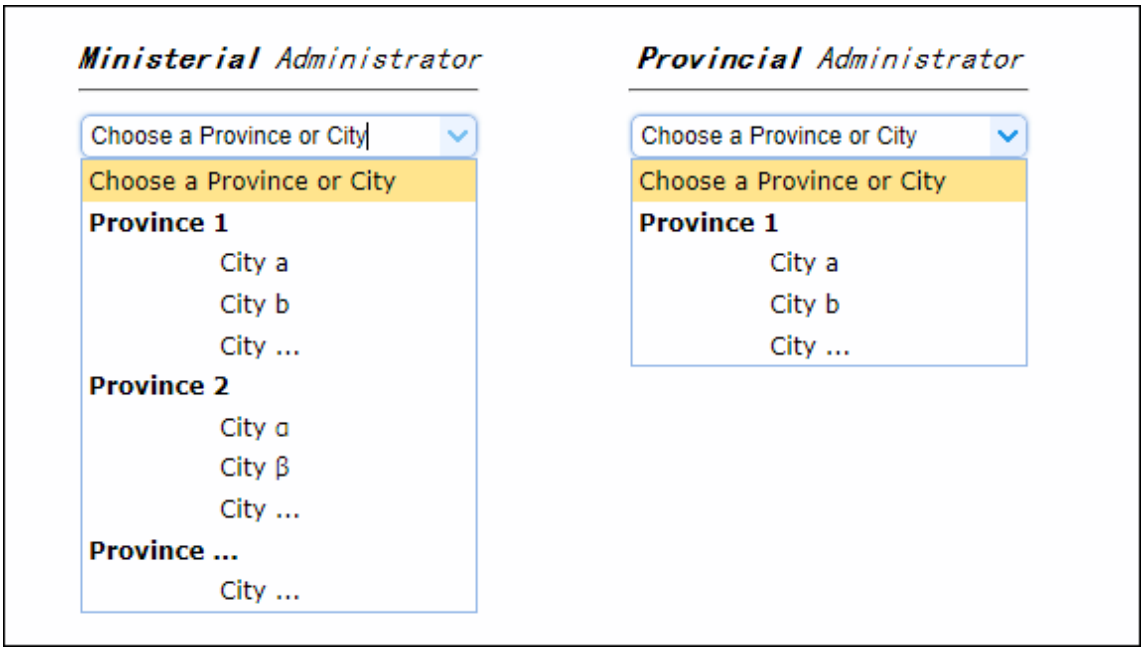

Fig. 2 Comparison of metadata scope between different Roles

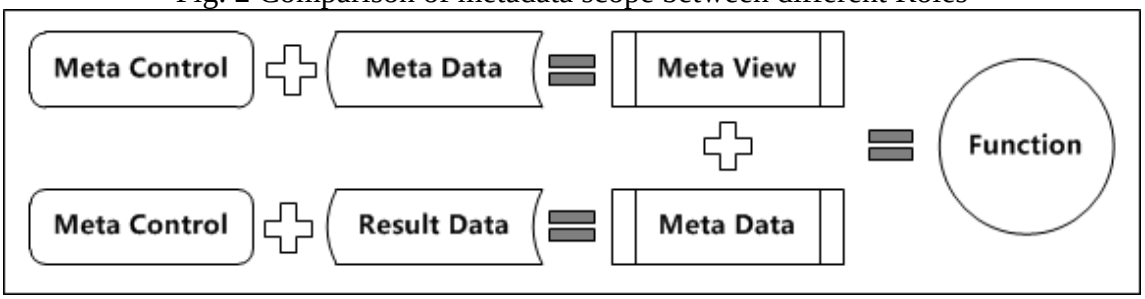

Fig. 3 Composition of Function view

In any management information system, each *page* or *form window* is made up of one or more controls, such as buttons, text boxes, drop-down boxes, group boxes, tables, and so on, meta control is also one of those elements. Meta control is responsible for loading and rendering metadata, and the methods of showing are usually changeless because of metadata's fixed types and ranges, such as combo box control of region, textbox control for time ranges, other discrete data and so on. All of the meta-controls are part of the page view.

Function view. The new model refines granularity of permission to a single function presented to the user as a view that named as Function View. A full function view is consist of meta view and result view. Meta view is made up of various types of meta control, which determines the range of data in result view. For example, a simple query interface, the collection of meta controls consisting of query conditions is named meta view, and data grid or detail sections is called result view. Both of meta view and result view form a complete and unified function view. The relationship is shown in Fig. 3.

Four-dimensional access control model. The four-dimensional access control (FDAC) model adds metadata on the basis of RBAC model which contains user, role and permission, and "permission" is replaced by "function" in the new model. The relationship between the four dimensions is shown in Fig. 4.

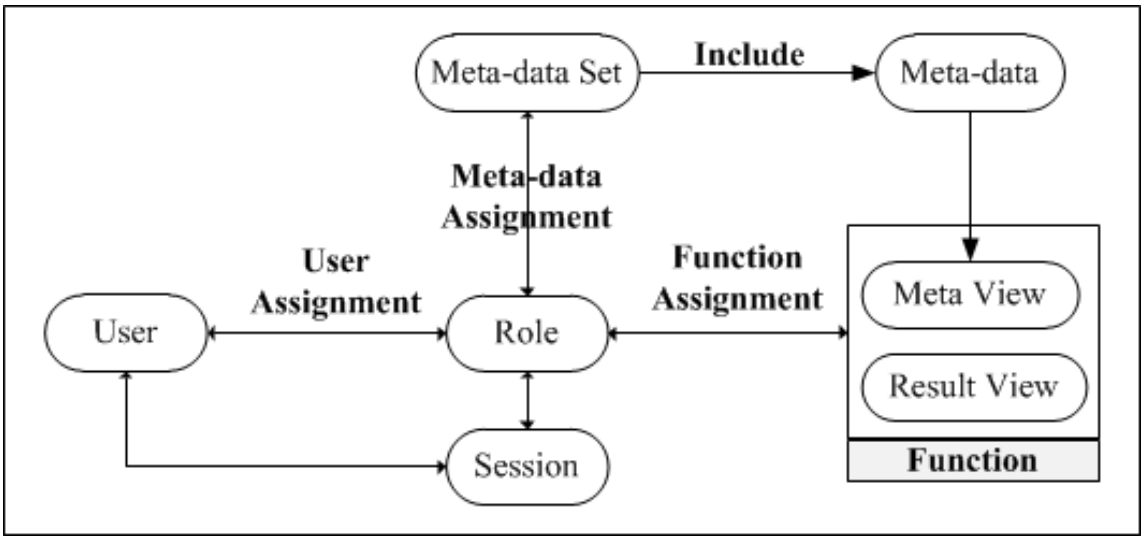

Fig. 4 Four-Dimensional Access Control model

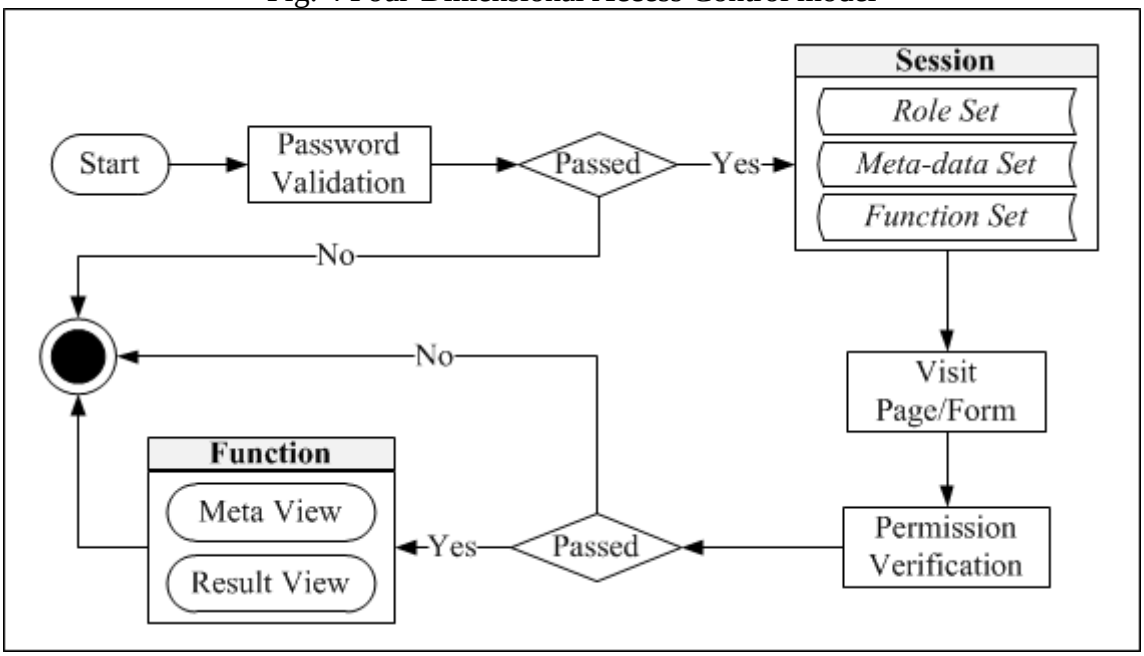

Fig. 5 Process of FDAC model

After user logged in, the session stored authorized roles and permitted functions and metadata set was produced. So the whole session includes four parts: logged user, role collection belonged to the user, function collection authorized to the user and operational data set, its function control flow shows in Fig. 5.

At first, user must input correct username and password and then pass the password authentication, at the next step, session will load roles of this user, metadata and function set of roles belonged to this user. After the above steps, if this user visit a function, the model should validate his right, he will be rejected if without authorization of the function. If verification is successful, the function view will initialize meta controls by metadata assigned to the user, and he may be able to operate a certain provincial data or all data in the country that depends on the roles he played. Finally, the data control authority is realized.

Implementation of FDAC model and its advantages

Implementation of FDAC model. In comparison with the RBAC model, metadata and meta control are very important elements in the FDAC model, so their design and implement are key factors in the model. From the foregoing discussion, most metadata exists in the form of data set, therefore, metadata could be stored in the session cache by a data array or data list after user logs in. It not only facilitates access but also improves the efficiency. There are some difference in the

implementation of meta control because of various programming languages and platforms.

In the C/S software development based on the .NET platform, packaging and implementing meta control can be realized through the Custom-Control technology [5]. In the B/S software, it can be accomplished through User-Control technology [6]. If the system developed by ASP.NET MVC, we can achieve it by Partial View or Custom HTML Helpers Extension technology [7]. Though, of course, it has some similar ways in Java platform.

Regardless of which method to use, the development of meta control is trying to get metadata scope of the user from its current session, and then data access control is realized while initialization of meta control is completed.

Advantages of FDAC model. First, compared with RBAC model, the FDAC model enables dynamic configuration of user's data permissions, user's operational metadata range can be easily adjusted after management information system published, so it brings about great convenience to users of the system.

Second, FDAC model facilitates the assignment and control of system functions, and at the same time, it reduces the complexity of business logic and difficulty of coding. In addition, the new model gets pages/forms reusable because of the combination between data permission and function permission.

At last, design and implementation of the model is in line with the principle of modular method of development and design. It gives consistent interface, reduces duplication of coding, and increases efficiency of development. So, it’s very beneficial to maintain and expand the system later, that the promotion and use of the system is very helpful and will save a lot of manpower and money.

Summary

Based on the RBAC model, this article proposed a four-dimensional access control model which added the limitation of metadata. It resolved the problem of data permission and function permission could not be controlled at the same time. It meets a wide range of management information systems for rights management requirements. Now, the new model has been used in many systems and got a greater appreciation of software developer and user of system, it has a better promotion and practical significance.

References

[1] Guo Hui, Wang Jianzhen. 7th International Conference on Computer Science & Education(2012) Melbourne, Australia, July 14-17,2012:1294-1297.

[2] Elisa Bertino.RBAC models-concepts and trends[J].Computers and Security,2003,22(6): 511-514.

[3] David F. Ferraiolo, Richard Kuhn D. Role–Based Access Controls.15th National Computer Security Conference (1992) Baltimore, Oct 13–16, 1992:554–563.

[4] Ravi S. Sandhu, Edward J. Coynek, Hal L. Feinsteink, Charles E. Youmank. Role–Based Access Control Models. IEEE Computer,1996,29(2):38–47.

[5] Andrew Troelsen. Pro C# 2010 and the .NET 4 Platform: New York, Apress, 2010.

[6] Matthew MacDonald et al. Pro ASP.NET 4 in C# 2010:New York, Apress, 2010.

[7] Jeffrey Palermo et al. ASP.NET MVC 4 in Action:New York, Manning Publications, 2012.