Research on the Role-Based Access Control Model and Data Security Method

Junhua Deng[1](✉), Lei Zhao[1], Xuechong Yuan[2], Zhu Tang[2], and Qian Guo[3,4]

[1] State Grid, Jiangsu Marketing Service Center (Metrolgy Center), Nanjing 210000, China
[2] State Grid Beijing Electric Power Co., Ltd, Beijing 100000, China
[3] Global Energy Interconnection Research Institute Co., Ltd, Nanjing 210000, China
guoqian@geiri.sgcc.com.cn
[4] State Grid Key Laboratory of Information and Network Security, Nanjing 210003, China

Abstract. For management of massive data and centralized network, the data security research team has become a central issue. Role-based access control (RBAC) is a common practice to control access to sensitive data. This article uses Flex and C# to develop an RBAC model, which can make data management more flexible. Concentrating on data security, this article presents the layer structure and model design for standardized management of security access. This article also implements detailed approaches for data security by verifying parameters and codes, access control, preventing illegal users and using oracle parameters and stored procedures for database security technology. In addition, an empirical study was conducted to assess effectiveness and usefulness .

Keywords: Access control · Data Security · Hierarchical framework · Parameter · Stored procedure

1 Introductions

Recently, as the big data and the popularization of network platform technology developing, many governments and enterprises have started to devote themselves to research on data access control and data security, and to develop data and information network applications and services to realize centralized and easy management information and resources. Therefore, as the emergence of Internet and Web platform, the development mode of enterprise information system has gradually developed from C/S to B/S [1]. Using the B/S mode, the problems of the traditional C/S mode can be solved, for example, difficult system maintenance and heavy workload. For point-to-multipoint, multipoint-to-multipoint and TCP/IP B/S mode using this open architecture, its security depends on the database server to manage passwords. The security of application layer information system, especially the security of B/S mode, becomes more prominent. Therefore, in the B/S mode, how to take effective approaches to enhance the security system of basic data, solve the security problem of confidential data, and effectively provide services for the big data system, is a hot research issue [2]. This paper adopts both Flex and C#, gives the design method of authorization management, and introduces the technology and method

© Springer Nature Singapore Pte Ltd. 2021
Y. Tian et al. (Eds.): ICBDS 2020, CCIS 1415, pp. 86–96, 2021.
https://doi.org/10.1007/978-981-16-3150-4_8

to ensure data security. Through the use of Adobe Flex Builder 3 and Microsoft Visual Studio 2008 and with Oracle 10 g, the back-end database server adopts the technical route of Flex and Visual C#.Net.

2 Related Overview

2.1 Access Control

Access control (AC) is a mechanism to achieve integrity and confidentiality in software systems. It is often on the basis of access permissions (also known as authorization), and its main function is controlling the system resource permissions scope, for example, deciding which users can access which objects or perform what operations [3]. The authorization management system structure is displayed in Fig. 1. It plays a vital role in restricting users' access to confidential resources and preventing damage caused by unauthorized users' intrusion or accidental operations of legitimate users. Thus, establishing a general AC model will greatly improve the security of enterprise information systems.

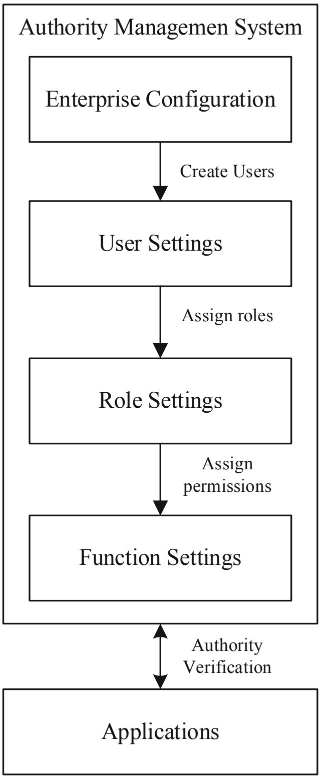

Fig. 1. Authority management system structure

Whenever the resources and users' number are high, such authorizations number becomes large. Furthermore, if the user base is highly dynamic, the number of authorization grant and revoke operations to be conducted will become very hard to manage. Traditional AC mechanisms can neither meet the requirements of fine-grained control, nor can they be implemented rapidly. Therefore, RBAC is put forward as an alternative method. At present, this form of RBAC has obvious advantages and potential, and has been widely applied to replace traditional mandatory AC (mandatory AC MAC) and Discretionary AC (DAC).

MAC security policy controls access according to the classification of objects and topics. An object is a passive entry that stores data (such as relationships), while a topic is an active entity accessing an object, usually an active process running representing the user. The access class is made up of two components: security level and a group of categories. Security levels are elements of a hierarchical ordered set. The levels usually used are top secret (TS), secret (s), secret (c) and unclassified (U), where TS > s > C > U.

DAC specifies rules according to which the principal can create and delete objects, and grant and revoke authorization for others to access objects. The user's access to data is managed according to the user's identity and the predefined discretionary rules determined by the security administrator. Rules specify the type of object that users are allowed to access for each object and user in the system. Check the user's request to access the object according to the specified permission; if there is authorization, it indicates that the object can be obtained in a certain mode, then the access right will be granted; otherwise, it will be rejected. These strategies are discretionary because they enable users to grant access to objects to other users. Free decision strategy is used in commercial system because of its flexibility, which makes it suitable for different protection requirements.

The AC purpose is limiting the operations that legitimate users of computer systems can conduct. AC limits the operations that users can perform directly and the programs that are allowed to execute representing users. In this method, AC attempts to block activities that could cause security violations. There are two kinds of resources in the computer system: active subject and passive object. The method a principal access an object is called access rights. Access rights enable the subject to manipulate objects (write, read, execute, etc.) or change AC information (transfer of ownership, grant and revoke rights, etc.).

AC can be on the basis of various strategies that follow different principles. The selection of security strategy is very significant since it affects the system flexibility, availability and performance. This strategy is on the basis of the following principles [3].

Principle of minimum and maximum privileges: Based on this principle, users should adopt the minimum set of privileges required for the activity. In this regard, the principle of maximum privilege is on the basis of the maximum data availability principle.

Principles of open and closed systems: For an open system, all access that is not expressly prohibited is allowed. For a closed system, all access must be permitted with explicit authorization. Closed systems are inherently safer.

Centralized and decentralized management principles: This principle addresses the issue of who is in charge of maintaining and managing permissions in the AC model. In different parts of the centralized control system, the authority of the centralized control system is different.

2.2 RBAC Model of the System

RBAC was first proposed by Ferraiolo of NIST in the 20th century [4]. After 1990s, it become the AC model mainstream gradually. Its core thought is to achieve the logical connection between users and authorization by granting role with authorization. This article offers a combined solution of operation authority, column function permission, system table authority and data table permission, so as to satisfy the centralized management organization control requirements in multi-level authority. Dynamically specify the system role based on the actual situation. The authorization administrator is only required to grant and revoke the proper role membership. The relationships defined between users, roles, and privileges are shown in Fig. 2.

There are two advantages for RBAC: (1) the change rate between user and role is higher than that between role and authorization, and the complexity of authorization management and maintenance cost can be reduced. (2) It improves the flexibility of implementing organizational AC strategies and corporate modifications.

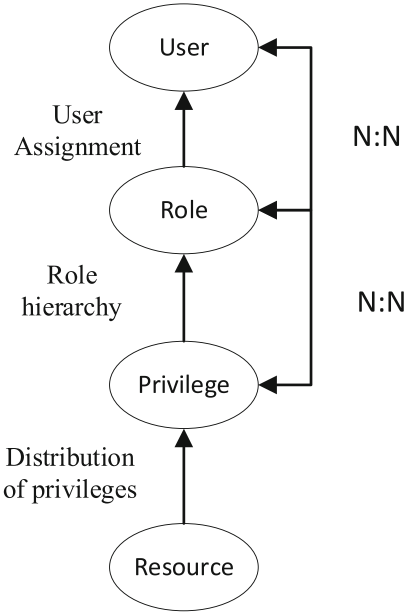

Fig. 2. RBAC model

3 RBAC Model Design

3.1 Layered System Framework

In the basic RBAC of this standard, the entire management system is on the basis of B/S and adopts a mainstream layered structure. This design's advantage is the hierarchical encapsulation of business subsystems, which reduces the degree of coupling, thereby improving the system scalability and maintainability [5].

3.1.1 The Communication Mechanism Between the Server and the Client

The Flex used in front-end user interaction is applied to solve the framework [6] by the third-party component FluourineFx, which calls C# classes directly from the server to realize the business logic layer interaction between the server and Flex.

If C# is required to interact with AS, the problem of data type conversion between the two languages shall be solved first. FlorineFx can facilitate that. The RemoteObject mechanism of AS is adopted to implement the calling approach. In the setting file, the tag [RemotingService] will be added before the called C# class, and then the following nodes will be added in file-remoting-config.xml to describe the remote call service setting file, where "source" specifies the remote object in the form of a standard name. Target decides whether the Flex Client can correctly access the remote object. This paper also suggests using struts framework model to support the development of B/S structure.

The design diagram of the role authority management system based on the Struts framework is shown in Fig. 3. The system is developed using Struts architecture, which is a classic MVC framework. The system model is composed of JavaBeans, which are

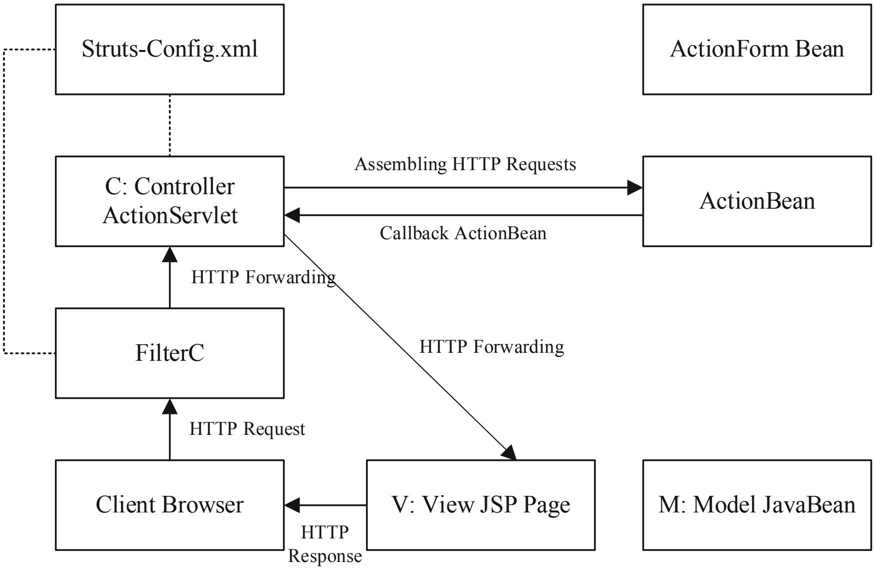

Fig. 3. Design diagram of role authority management system based on Struts framework

used to implement business logic, the controller is implemented by Action Servlet and Action, and the view is composed of a set of JSP files. In the application, Action is treated as an adapter for user request and business logic processing, and the business logic is completed by JavaBean.

When the ActionServlet controller receives a user request, it will forward the request to the corresponding Action instance.

3.1.2 Layered Design

By integrating C#.Net and Flex together, Flex programs can operate in a browser, and the browser's plug-in FlashPlay is responsible for interpretative execution [7]. Flex is mainly in charge of customer display. The middle layer is separated into interface layer, business logic layer, web layer, and data layer. Figure 4 indicates the layered system architecture.

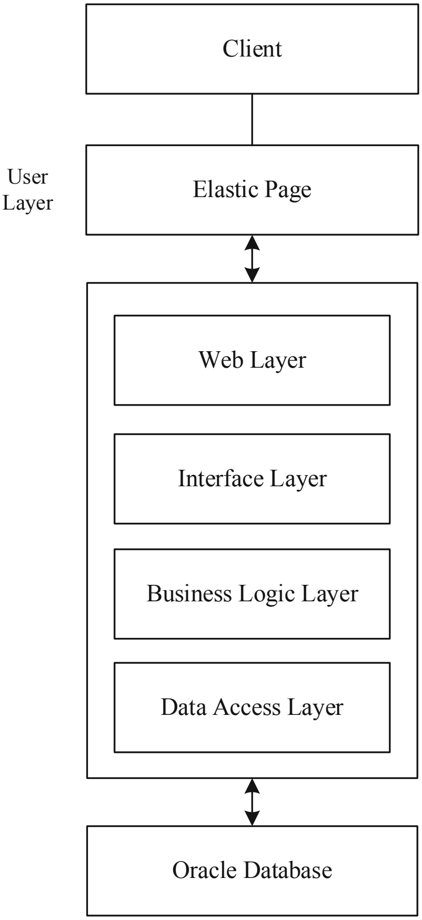

Fig. 4. Hierarchical system framework

User layer: Flex elements such as MXML and ActionScript are adopted as a service to provide users with a graphical interface [8];

Interface layer: Provide interface functions to facilitate the interaction between the user layer and the business logic layer;

Web layer: display several setting files applied in the communication mechanism;

Business logic layer: the intermediate connection between the data access layer and the interface layer, used to realize the data exchange, separate user operations from the database, including user class (BUserInfo.cs), role class (BUserRole.cs) and Permission class (BUserPrivilege.cs), etc., can realize business processes, such as adding, updating, deleting, etc.

e) Data access layer: Directly interact with the database for the underlying operations of the database, and realize the connection with the data and the interaction with the business logic layer.

3.2 Database Design

The massive data is a multi-disciplinary project, involving basic geographic data, marine environment data, RS image data, metadata and so on, and it requires all-around consideration of data sources. Thus, RBAC model is designed, which is made up of the following ten tables: table for user, role, user role, permission, role permission, permission menu, menu, list, database and data. In the data management system, it is a convenient method to use the above model as the basic infrastructure, which can dynamically assign and revoke authorization, and precisely control data access, data operation and operation modules. Thus, users can play and activate different roles simultaneously, and various permission types are available to access infrastructure managed resources.

4 Method Design

4.1 Identity Verification

There are four kinds of authentication for RBAC: user name and password, X.509 certificate, network address, and existing authentication token.

Password authentication, through a dedicated network service, checks the user name and password according to the central nice account database. User account information is kept in RBAC's own database.

Certificate authentication, if the user has X.509 certificate, it can be used on the standard client authentication mechanism of TLS/SSL protocol. The certificate data is then adopted to find the user name in the RBAC database.

Network address authentication, some clients can use the lookup table in RBAC database to verify their IP address. Generally, address authentication is only allowed on a very limited number of devices (e.g. control room console).

Using existing token to request a new token as long as the original token has not expired, the token will have a valid tag, and be published to the same location address. The validity of the new token will not exceed that of the original token.

User identification and authentication is adopted to identify the visitor's identity. It can identify whether each user is in the user set through the unique identifier (i.e. user

ID) in the user table; besides, it adopts the password mechanism, only the user entering the user name, the correct password, and the consistent verification code is allowed [9].

4.2 Verification Code

The system program will automatically generate pictures composed of a series of random numbers and letters mixed, these pictures deform the font and place them on a complex background. Actually, the user should enter the password and submit the user information through visual recognition, and then enter after successful verification.

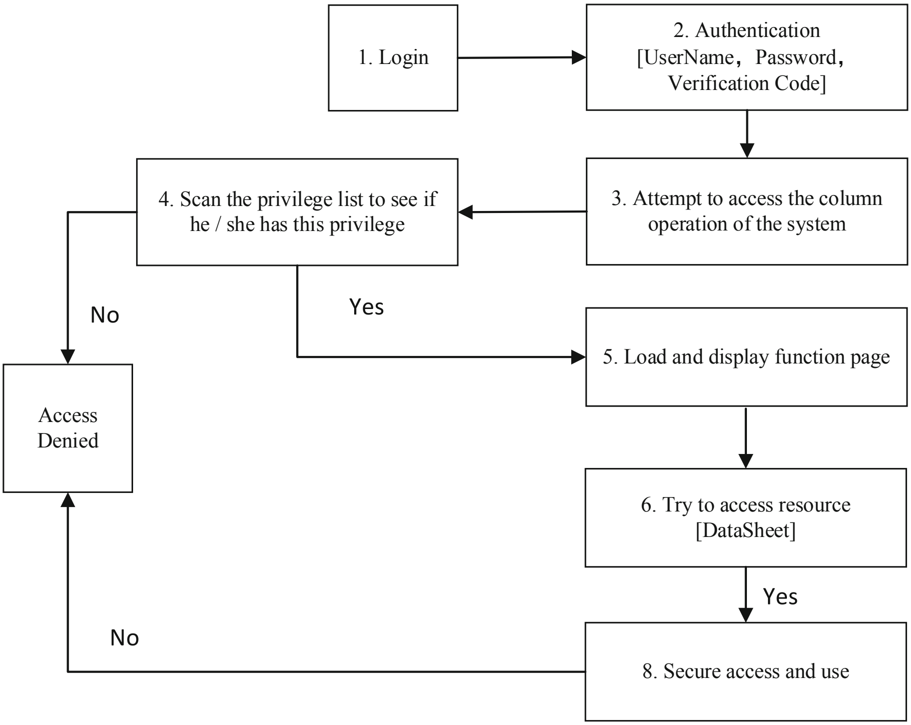
1. Login
2. Authentication [UserName，Password，Verification Code]
3. Attempt to access the column operation of the system
4. Scan the privilege list to see if he / she has this privilege
No
Yes
Access Denied
5. Load and display function page
6. Try to access resource [DataSheet]
Yes
No
8. Secure access and use

Fig. 5. Authorization process

4.3 AC Interface

For users, it is very significant to supervise authorization and control access scope. Users will build a session when logging into the system during which he/she can request activation of a subset of the roles he/she is entitled to play. The user obtains all permissions linked with the role activated. Figure 5 indicates the authorization processing flow.

4.4 Prevent Illegal Users from Logging in to the System

For the information system in the Web environment, it can be accessed through IP address or computer name. To prevent any illegal users, it can be solved by the AC page. When a link is opened, each function interface (adding a custom function) will judge the user’s permissions. If the user is illegal, the system will reject this user and display a prompt and switch to the login page. Besides, the login page is also added to some pre-tested content, calling the code named “HasDangerousContents” of the BinJection class to confirm whether there are dangerous characters to protect the system from being similar to “SQL injection” Attack of the attack [10], which will add some operators to your legitimate input parameters, such as AND or OR.

4.5 Database Security: Oracle Parameters

The purpose of user data and resources separation kept in various databases is to reduce the data security risk. After judging the access legitimacy, the system will obtain the connection information to the data source so that resource operations can be performed. On the other hand, by using triggers, unique IDs can be specified, for example, role IDs must be automatically enabled for roles. Besides, the database operation SQL statement in this process uses OracleParameter. Writing is complicated to some extent, but it makes sense to avoid security risks brought by the code and can improve the performance of Oracle. For instance, here are some common programming statements practically:

```
string sql = "select * from dtName where field = ’" +value + "’";
OracleParameter[] param = {
New OracleParameter("@tbName",OracleType.VarChar),
new OracleParameter("@field",OracleType.VarChar),
new OracleParameter("@Value",OracleType.VarChar)};
param[0].Value = dtName.ToUpper();
param[1].Value = field;
param[2].Value = Value;
string sqlStr = "select * from " + @dtName + "where " + @field + "=’"
+ @Value + "’ and IsValidate=1";
```

4.6 Database Security: Stored Procedures

Security database is a special part of the system, which has two main functions: audit rules and audit analysis. To meet higher security requirements, the system need to achieve access through views by executing stored procedures. Rather than using SQL statements directly in the application, calling stored procedures [11] has some advantages. First, several SQL statements can be put into a stored procedure and conducted in batch. Therefore, the network traffic can be reduced, and secondly, it is faster. Stored procedure needs to be analyzed and optimized for the first call. However, when the stored procedure is called again, it will be called directly from internal memory. Besides, oracle can be used to pass parameters rather than introducing specific values directly. An example is given below, showing the integration of stored procedures:

```
OracleParameter[] param = {
new OracleParameter("p_url",OracleType.VarChar),
new OracleParameter("p_name",OracleType.VarChar),
new OracleParameter("p_pass",OracleType.VarChar),};
param[0].Value = "210.37.45.87"; // parameter1
param[0].Direction = ParameterDirection.Input;
param[1].Value = "aud";// parameter2
param[1].Direction = ParameterDirection.Input;
param[2].Value = "a";// parameter3
param[2].Direction = ParameterDirection.Input;
OracleCommand cmd = new OracleCommand();
cmd.CommandText = storedProcName; //state the name of storage procedure
cmd.CommandType=CommandType.StoredProcedure;
```

5 Conclusion

This article focuses on the massive data's centralized management. Flex technology for user interface interaction has the characteristics of unified planning, overall layout and direct display. Its security model for reducing authorization management workload is absolutely effective. All the approaches used in this article mainly lie in the encapsulation of security policies and the applications flexibility. For complex data management systems, this method can also be used to obtain role access and data security in engineering security.

Acknowledgement. This paper is supported by the science and technology project of State Grid Corporation of China: "Research and Application of Key Technology of Data Sharing and Distribution Security for Data Center" (Grand No. 5700-202090192A-0-0-00).

References

1. An-Qi, Y., et al.: Hospital information system based on C/S structure automatic update practice. Smart Healthcare (2019)
2. Linan, Z.: Design and realization of surveying and mapping instrument information management system based on B/S. Geospatial Inform. (2019)
3. Power, D., Slaymaker, M., Simpson, A.: On formalizing and normalizing role-based access control systems. Computer J. **52**(3), 305–325 (2018)
4. Li, Q., Xu, M., Zhang, X.: Towards a group-based RBAC model and decentralized user-role administration. In: 2008 The 28th International Conference on Distributed Computing Systems Workshops. IEEE (2008)
5. Technology, Technology, Hefei. Model design of a network security system for project management systems based on the B/S architecture in ASP.NET platform. Comput. Sci., 35(2), 101–103 (2008)
6. Qian, H.: Server communication technology based on Flex. J. Guangdong Transport. Vocat. Tech. Coll., 2012(04):20–22+101

7. Zou, S., et al.: Peer-assisted video streaming with RTMFP flash player: a measurement study on PPTV. IEEE Trans. Circuits Syst. Video Technol. **28**(1), 158–170 (2018)
8. Jones, M.E.: Developing Flex 4 Components: Using ActionScript & MXML to Extend Flex and AIR Applications. Addison-Wesley Professional (2010)
9. Sun, B.: Research on identity tracing based on ABS fine-grained privacy isolation. Electron. Design Eng., 26(377(03)), 10–14 (2018)
10. Lian, K., et al.: Research on multi-level detection methods for SQL injection vulnerabilities. Comput. Sci. Explor. **05**(005), 474–480 (2011)
11. Zhang, W., Chang, H.: Model design of a network security system for project management systems based on the B/S architecture in ASP. NET Platform% design. Comput. Sci., 035(002), 101–103,108 (2008)