УДК 004.056.5

Малишев Андрій Ігорович
бакалавр комп'ютерних наук
Національний технічний університет України
«Київський політехнічний інститут»,
Навчально-науковий комплекс
«Інститут прикладного системного аналізу»

Малышев Андрей Игоревич
бакалавр компьютерных наук
Национальный технический университет Украины
«Киевский политехнический институт»,
Учебно-научный комплекс
«Институт прикладного системного анализа»

Malyshev A. I.
Bachelor of Computer Science
National Technical University of Ukraine
"Kyiv Polytechnic Institute"
Educational-scientific complex
"Institute for Applied System Analysis"

РОЗРОБКА СПОСОБІВ УПРАВЛІННЯ В КОРПОРАТИВНОМУ JAVA-ДОДАТКУ

РАЗРАБОТКА СПОСОБОВ УПРАВЛЕНИЯ В КОРПОРАТИВНОМ JAVA-ПРИЛОЖЕНИЯ

DEVELOPMENT OF METHODS OF MANAGEMENT IN AN ENTERPRISE JAVA APPLICATION

Анотація. *Метою написання даної статті було порівняння декількох способів розробки управління доступом в корпоративному Java-додатку. Була піднята проблема використання того чи іншого способу в залежності від потреб та предметної області. Також запропоновано можливі шляхи вирішення.*

Ключові слова*: ECM-додаток, URL, business-to-business, клас, анотація.*

Аннотация. *Целью написания данной статьи было сравнение нескольких способов разработки и управления доступом в корпоративном Java-приложении. Была поднята проблема использования того или иного способа в зависимости от потребностей предметной области. Также предложены возможные пути решения.*

Ключевые слова: *ECM-приложение, URL, business-to-business, класс, аннотация.*

Summary: *The purpose of writing this article was to compare several methods of development and access management in an enterprise Java application. There was a problem with the use of a particular method depending on the needs of the subject area. Also proposed possible solutions.*

Key words: *ECM application, URL, business-to-business, class, annotation.*

Access control is one of the main parts to the security of web applications. Access control ensures that only authenticated and authorized individuals may have access to confidential information, and only a user with a valid role can perform the given action. The formation of the role is to define clear and understandable for users of the information system rules of access. Role separation allows the use of flexible, dynamically changing the operation of the application rules access control [1].

Let us consider a few application scenarios that may require security for the individual classes in addition to implement security based on URL.

Automation of business processes

Workflows (workflows) in applications for business automation are composed of processes. For example, the sequence analysis of blood in a clinical laboratory may consist of several steps, and each step considered a business process:

- An employee takes a sample of blood of the patient and assigns the sample identification number.
- The laboratory technician performs sample analysis and preparing the results of the analysis.
- A qualified doctor writes a report based on the results of the analysis.

It is obvious that each process executes a single user who authorized to execute this process. One who not authorized to perform process should not be able to start this process. For example, the technician must be able to prepare the results of the analysis, but should not be able to write a report about the analysis.

Authorized business processes usually have applications for business automation. Typically, each business process implemented as a Java class, and these classes must be protected by appropriate access control policies.

Integration intercorporate

Intercorporate (business-to-business (B2B) integration is usually called the scenario when two companies need to provide each other with access to its functionality. For example, the hotel can access the operator access to its functionality to reserve rooms, which the tour operator can use to reserve rooms for the tourists. As a partner, tour operator can receive a special discount when booking rooms. In this case, the system of booking rooms in a hotel should be able to authenticate the tour operator, before giving him access to special classes for room reservations and discount.

Consider a few ways to implement access control in an enterprise Java application.

The first method of implementation provides for the restriction of access to parts of the project. In addition there is imposed a limitation on the number of failed login attempts to the system, then the account is locked for a certain time, it will make the application of the method of "brute force" [2]. Implemented class manager users, which should provide add, delete, modify and read the user account, check for the existence of the login user and login the user to the system. In the class of the user information, you need to make unchangeable the user ID and login. This class should also contain registration information about the user (login / password, and user role) and his last login. You must also store information about the number of failed authentication attempts and the time / date unlock the user account if it locked. You need to implement a filter that restricts access to URL of the site according to the given constraints. Limitations should be set in the XML file [3] the path specified in the configuration settings of the program. When applying security restrictions you must choose which resources to protect, setting limits on the URL pattern and roles, who forbidden to visit the specified part of the site. The filter should intercept all requests from the client that come. If the URL of the HTTP request and the role of the user match the pattern and specified role in limiting access to the resource is limited.

An alternative method consists in application level security controller methods using annotations support [4]. Will check user rights to perform the methods, which labeled abstract. Function that annotation executed only if the user (of this thread) have the rights specified in the parameter annotation. Possible that annotation the whole class. Then all the class methods require the relevant rights, and we can override those methods, the fulfillment of which need special rights. An object that is configured with the annotation can be created using the factory container, because by creating the object using new operator would have to perform additional steps to verify user rights.

The proposed means of access control chosen depending on the requirements of the subject area and are not always interchangeable. Security, which is based on an URL that is not enough, when multiple roles have access to the same resource (the URL), but the role needs to perform different actions. For this scenario, you need to implement a more subtle security policy, applying different access rights to different parts of the resource.

In this work, we looked at several ways of designing access control policies, which based on an URL and call the methods of the controller, as well as cases in which better to apply certain ways.

Литература

1. Управление доступом на основе ролей, [Электронный ресурс]: URL: http://www.java.net/ (дата обращения: 5.07.2015)

2. Атака грубой силой, [Электронный ресурс]: URL: http://www.techopedia.com/definition/18091/brute-force-attack (дата обращения: 1.07.2015)

3. XML, [Электронный ресурс]: URL: http://www.w3.org/XML (дата обращения: 1.07.2015)

4. Java-аннотации, [Электронный ресурс]: URL: http://tutorials.jenkov.com/java/annotations.html (дата обращения: 1.07.2015)

References

1. "Role based access control", consulted on July 5, 2015, http://www.java.net/

2. "Brute-force attack", consulted on July 1, 2015, http://www.techopedia.com/definition/18091/brute-force-attack

3. "XML", consulted on July 1, 2015, http://www.w3.org/XML

4. "Java annotations", consulted on July 1, 2015, http://tutorials.jenkov.com/java/annotations.html