Research and design of system security based on .NET frame

Jing Zhou¹, Yurong Guan² and Fei Zhou³

¹Department of Computer, Huanggang Normal University, Huangzhou, Hubei, 438000, China
²Department of Computer, Huanggang Normal University, Huangzhou, Hubei, 438000, China
³Department of Computer, Huanggang Normal University, Huangzhou, Hubei, 438000, China
*jszhoujing@hgnu.edu.cn

Abstract. For using the .NET framework model of Web application system for Web development to some security problems in the analysis and discussion, this paper put forward some security prevention programs. Though .NET framework system itself has some security protection mechanism, but users in developing Web mode when the application is to avoid safe hidden trouble, should be more reliable in every link of security Settings.

1. Introduction
The reliability and high security of Web mode application system based on.net framework system structure has been the goal pursued by program developers and users. Although the computer hardware configuration is higher and higher and the price is lower and lower, upgrading hardware becomes the most direct and convenient way to improve system performance. But in the current.net framework Web mode application system is getting larger and larger, developers and users continue to increase the needs of the situation, relying on frequent hardware upgrades can’t achieve satisfactory results. How to make better use of the resources occupied and improve the performance and reliability of the Web mode application system itself has become the key issue of the system [1].This article will be based on .NET framework application system security analysis and discussion.

2. .NET Analysis of security strategy
There are three main types of authentication in .NET: Windows authentication, forms authentication and Passport authentication. Windows authentication works only for sites in a controlled environment, which is more suitable for an Intranet. Form authentication is particularly suitable for the application of the Internet; Passport authentication is ideal for cross-site applications and allows users to access any member site with a single username and password.

• Form-based authentication allows users to customize the login page and login guidelines. .NET user authentication redirects an unauthorized user to the login page. This authentication is a popular method used by many Web sites on the Internet.
• Windows-based authentication is mandatory and only applicable to websites in a controlled environment, and administrators can control client machines in a variety of ways. A Web. Config file needs to be configured in a .NET application to use Windows authentication.
Passport authentication is ideal for cross-site applications, where users can access any Passport member station with a single username and password and all Passport information is cleared when they log out.

With the continuous development of computer technology and network technology, the security of network environment attracts more and more attention. In .NET environment security analysis, you can use Net technology to design your own system security and application of new powerful.

3. .NET framework system common security problems

Data transfer between multiple Web pages is handled when making dynamic Web applications. The choice of data transfer mode between Web pages plays an important role in the rationality and security of system design and operation.

3.1. User login security issues

The user information (such as username and password) is stored in the database when the user registers with the .NET framework Web schema application system. When authentication is required, the browser sends the authentication information to the application server, and the application reads the corresponding fields in the server database and compares them with the data received by the form. If the user is considered to be a legitimate user, it is allowed to enter the system; otherwise, it is forced to boot back to the authentication interface. During user login, the system introduces a validator for user name and password in such a way that it cannot prevent all attacks by modifying parameters. When performing a numeric range check, the appropriate Type attribute should be specified based on the data Type required by the input field.

Example: suppose that the input value must be a symbol between a-z, the page code of the general verification method is:

```
<asp:RangeValidator... MinValue = "a" MaxValue = "z" />
```

But since the Type attribute value is not specified here, the RangeValidator validator only ensures that the starting string is validated by the system with characters between a-z.

3.2. Back-end SQL security issues

The primary source of user input data is parameters submitted in HTML forms. The user's input parameter content contains only "normal" username and password data, but no special characters. An attacker can change the meaning of a query with the help of some characters with special meanings, and then call any function or procedure to fool the server database into executing a malicious query to obtain any information stored in the back-end database.

The following C# code queries the back-end SQL Server database, assuming that the values of the user and password variables are directly taken from user input:

```
SqlDataAdapter my_query = new SqlDataAdapter("SELECT * FROM counts WHERE user = "+user +" AND password = "+ password, the connection);
```

On the face of it, these lines of code are perfectly fine, but in fact they can be a recipe for SQL injection attacks. As long as the attacker enters "OR a=a" in the user input field, he can log into the system smoothly.

3.3. Injection of malicious script security issues

Injecting malicious script refers to embedding malicious user information into response page. A .NET framework Web schema application system that does not handle user input well may allow users to inject their client scripts into a server-side database to perform illegal operations. For example, the following code implements a welcome message after a user enters a name and clicks ok, and a user verification message script is accepted by the system as normal input (assuming "zhoujing"). If the normal input method is not used, and a piece of malicious script code is input, such as: `<script language="javascript"> alert("script injection");</script>`. Then the system will accept the script of malicious user authentication information [2].
3.4. Information leakage security problem

Information about the application can be found in the _VIEWSTATE hidden field of almost all HTML pages in the .NET framework Web mode application system. Since _VIEWSTATE is encoded by BASE 64, it is often ignored, and an attacker can easily decode BASE 64 data to get the details provided by _VIEWSTATE.

- dynamic data from page controls
- data explicitly saved by developers in ViewState
- data password signature

4. .NET framework system security prevention scheme

The main question in .NET security is: who has access to the system? The common practice is to maintain a list of users' rights in the system, and the system verifies the legitimate identity of users who enter the system. The process of accepting user credentials and validating them against the specified authority is called authentication.

4.1. Effective use of security solutions provided by the .NET framework

The Microsoft .NET framework and IIS work together to provide Web-mode application security. Authentication, authorization and simulation are the basis of security implementation. Web mode applications in the .NET framework implement authentication by using authentication providers, which are code modules that validate credentials and implement other security functions, such as Cookie generation. .NET framework system supports Windows authentication, form authentication and Password authentication three authentication providers.

4.2. A security scheme for back-end SQL attacks

A domain validator is a mechanism for .NET developers to impose restrictions on the values of a domain, such as, Field values that limit user input must match specific expressions. For the above attack security problem, you can adopt the method of restricting the content of the input field to belong to a collection of legal characters, such as "[a-z]".

4.3. Malicious script injection security scheme

Perform input validation and HTML encoding of outgoing user data that appears in HTML pages in a multi-tier security architecture, ensuring that the browser treats user input data as pure text and not as anything with special meaning. For the previous example, add HtmlEncode call: response.write (server.htmlencode (textbox1.text)); The response HTML stream will then contain the HTML encoded version of what the user entered, that is, the browser will not execute the JavaScript code that the user entered.

4.4. Security scheme for data transfer between pages

There are many ways to pass values between two .NET framework Web pattern application pages [3], such as using QueryString method and Session variable method and server. transfer method to achieve value Transfer between pages.
QueryString methods pass values as a more traditional solution, much like urls in ASP applications. The main advantage of this approach is that it is simple to implement, but the drawback is that the value is displayed in the browser's address bar, especially for sensitive data such as user names.

The Session variable method is used not only to transfer the data value to the next page, but also to cross multiple pages until the Session variable's data value is removed.

The server.transfer method is more complex to transfer values, but it is especially useful to transfer values between pages. This method can be used to access exposed values on another page in the form of object properties, ensuring the security of the code. Use this method, it need to write some extra code to create properties that can be accessed on another page.

5. Conclusion
NET framework Web mode application applications are exploding, more and more applications are pushed out of the firewall, the security of vulnerable Web applications will only increase the risk. There is also growing pressure on developers to quickly complete applications before tight deadlines. Focus on the security aspects of writing code while investing the resources necessary to prepare for future Web services applications while ensuring the high quality of current ones. High quality and safe applications can only be constructed by taking the right steps to ensure their safety from the time they are born.

Acknowledgments
I thank the reviewers for their comments and suggestions, which helped me improve the quality of this paper. This paper is devoted to the education Science and technology research project of Hubei provincial department, the project name: "research on multi-point communication mode of Internet + local heterogeneous network" (no. D20172901). And this paper is devoted to the application of the Science and Technology Research Program of Hubei Education Department for the key project of youth in 2019. The project name: "PIL digital image processing research based on Python". And thank other scholars for their research and Discussion on Python image processing. Thank you for the support of the teaching and research platform of Huang gang Normal University and the virtual simulation experiment centre platform of the department of computer.

References
[1] Wang, wenlong, (2015), ASP.NET technical insider [M]. Beijing: people's posts and telecommunications press, page 550, the first printing.
[2] Zhong, chuyue, (2017), ASP.NET application security defects and prevention strategies [J]. Journal of anhui university of technology. 3rd issue, 58-62
[3] Zhou, jing, (2016), Exploration of data transfer mode between ASP and ASP. Net pages [J].Microcomputer information, 2006, 5-3:11, 4-115.