Security Research about Asp.net Web Application

Zhenjun Fan
Computer science department
Tonghua 134002, China
Tonghua Normal University
Email: jlthfjq@163.com

Zhenmei Fan
Steelmaking & Hot rolling Plant
Tonghua Iron & Steel Company
Tonghua 134003, China
Email: lotoffun_1@163.com

***Abstract*-The implementation method and security measure of security about improving Asp.Net application are discussed from two aspects: .Net security platform frame design and application design. The Principles and characteristics are analyzed in detail from the platform frame design level, the specific implementation methods of Form-based authentication are presented, and concrete measure is proposed to prevent Hacker from application design level.**

Keywords-Network security, Authentication, Authorization, Security measurement

I. INTRODUCTION

ASP.NET is WEB application based on Microsoft.NET, it is the trend of future WEB application development, and the security is the most important aspect among the ASP.NET WEB application. It consists of two layers based on the security of asp.net Web application: A)The security frame design of .net platform level, mainly solving the user authentication and authorization information. B)The security design of application transmission level, it mainly consists of reducing the common secure threats, and preventing from hacker's invasion.

II. . THE DESIGN OF SECURITY FRAME ON THE PLATFORM LEVEL

A. The basis of ASP.NET security implementation

ASP. NET and the Microsoft Internet Information Services (IIS) work together [1], which provide an infrastructure for the Web application security, Asp.NET security structure is shown in Figure 1.

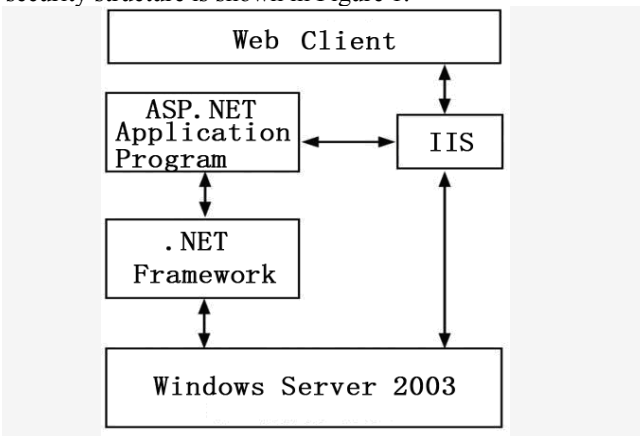

Figure 1. Asp.NET security structure

All web clients communicate with ASP.NET Application by Mircrosoft Internet IIS service. Authentication is done by IIS when needed, and then requested resource is found. If the client is authorized, the resource is available. So, application security can be ensured by built-in security processes of NET security architecture. Several critical security processes of ASP.NET consist of three factors : Authentication, authorization and Impersonation, which are the basis of security implementation [2] .

1.) Authentication

Authentication is identity obtained from the client, for example user name and password, and it is also the process of validating those identity. The purpose of Authentication lies in confirming the identity of user. After the Authentication, Authorization process will decide whether the identity can access the certain resources.

2.) Authorization

The purpose of authorization is to decide whether the access right of the defined resource request should be authorized to the certain identification. Access right is authorized to a certain resource by two ways: file Authorization and URL authorization. The file authorization will check according to Access Control List (ACL) or resource right, and confirm whether the authenticated user can access resources. URLauthorization will authorize the identification of WEB spaces according to URL.

3.) Impersonation

When impersonating, user is authenticated by IIS, and then the authenticated identification will be transmitted to Asp.Net application, Otherwise, if the users can not be authenticated, the un-authenticated identification will be transmitted. However, if the Impersonation is started, Asp.net application will impersonate any achieved identifications[3]. The current Asp.net application permit users to obtain access right or refuse access depending on NTFS catalog and some configuration in the files. In order to configure files, the disk which application files lie in, should be changed to NTFS format.

B. The two basis schemes of the implement of Asp.NET application security

Asp.net has many kinds of secure working schemes, the two general working schemes of security are: A)Authentication using windows, not permitting anonymity access and starting impersonation B) Authentication using windows, not permitting anonymity access and forbidding impersonation.

1)Authentication using windows,not permitting

anonymity access and starting impersonation

The scheme mentioned above can reduce the amount of security programming of asp.net application in maximum[4], it depends on IIS authentication and file security of windows NT.

In general, there are two cases which are suitable for authentication mode using windows. One is Intranet station (i.e. company internal station); the other case is that you know which users will access the stations in advance. You must create a windows account for each user, and set a user name and password as the logon certification when visiting website. You'd better use group if user account is too large. Logon user must possess of access right in the catalog where the Asp.NET application lies in, so you can access the Asp.NET application.

2)Authentication using Froms, not permitting anonymity access and forbidding impersonation

The authentication based on Forms can customize logon page and logon principle. Unauthorized user will be redirected to logon page . This kind of authentication is a popular method on the many web sites of internet.

For the Forms authentication, the logon user information is already in the Database. The main idea is that re-directing the request of unauthenticated sent from client to a certain page, and then asking the user to submit the required certification information on this page. If the information is authenticated and authorized, the application will send out a Form (called authentication Form) that contain the secrete key (send to client as Cookie, called authentication ticket), then redirected to the page of the user original request. When user request again the page in the same session, the request header will contain authentication information in order to re-authenticate and re-authorize.

C. Implementation example of Forms authentication scheme

Forms authentication can be achieved by the 4 methods 1. authenticating users directly in code; 2. Using web.config to achieve validation; 3. using database to achieve validation; 4. using certificates (xml) files to implement authentication. Due to space limitations, only No.2,No 3 methods are discussed in this paper.

1) Authentication using web.Config

In the configuration file, user name and password are defined by <credentials> item of <forms>'s sub-element. When user click "Logon "button, the "Authenticate" method of "FormsAuthentication" object is called in the event of click, the system will automatically compare the certification information entered by user with user name and password of <Credentials> item. If they are matched each other, authentication is passed.

If validated, the Authenticate method returns "true" result, then Cookie is created by "GetAuthCookie"method of "FormsAuthentication" object, and then the Cookie is saved to the client, and finally redirected to the protected page. If not validated, it will be redirected to the logon page.

The implementation scheme is suitable for the system which has less user account. However, plaintext codes are transmitted which is the shortcoming of this scheme, It would be better if the information inputted by user is verified by "RegularExpricessValidator" .

2) Authentication by database.

Adding a user in the database table , with userName, password field, noting password field will be encrypted using MD5 or SHAI algorithm. The username and password of usertable in database is compared with the account information provided by logon page, if they are matched each other, authentication is passed. Otherwise, redirected to the logon page [5].

The Main steps are as follows:First,The database data is read by "sqlConnection" and "SqldataReader" objects, Sceond,The user's password is encrypted by "HashPasswordForStoringInConfigFile " Method of "FormsAuthentication" object, Third,The password is defined whether it is stored in the database, if it exists, the cookie used for saving user informatin is created, and then protected page is popped up, otherwise it will be redirected to login page.

III. SEVERAL MEASURES IS MADE IN THE TRANSPORT LAYER OF APPLICATION TO IMPROVE SECURITY

.Net framework and IIS functionality are fully used in the following two methods, so safety can be well assured. but these two methods are transmitted by plaintext code, security risks brought by Hacker attacks are considered insufficiently. Thus, in order to achieve real security, further measures should be taken to prevent all kinds of hacker attacks.

A. The prevention of Script Injection

Many security risks are caused by not properly handling the code inputed by user. Such as "<" and ">",those are Javascript which can execute directly in the web page。So，all places where user can input parameters, such as input box and URL,should validate the input parameters. The "<"and ">" characters should be shielded .Otherwise, System is likely to attacked or malicious accessed. To enhance the security of ASP.NET applications, we should limit the length of the user input information to an appropriate range. The input field allows the user to input an infinite amount of information, give them the opportunity caused by buffer overflow attacks, which would cause the application to crash or allow them to gain control of the computer.So, during development, we should restrict the information, including the length,content and format.

B. Storing secret information by Hash

Confidential content such as database connection strings, user names or passwords are stored somewhere in the system, they are the most vulnerable objects. We can store the hash value instead of storing encrypted value. When we want to compare the hash string, we can only compare the hash information, rather than the original plain text value . Hash calculation is achieved by a standard well-known formula (e.g. MD5) , it is very difficult to reverse engineering or counterfeit engineering. ASP.NET can easily achieve encryption of password, the class called FormAuthentication is included in the System.Web.Security

namespace . In this class, a method called HashPasswordForStoringlnConfigFile can achieve information encryption. It supports "SHA1" and "MD5" hash algorithm used to encrypt the string. the user's password can be changed into messy code, and then stored. And these two encryption algorithms are not reversible, even if the intruder enter the server database system, it can not restore the password stored in the database, so that encrypted password, instead of actual password is stored in the database to ensure the security of the user, data and database.

C. avoid sql injection attack

Any errors in the application will result in unauthorized access to the database, a common method of attack is SQL injection attacks, it is very dangerous. The so-called SQL injection attacks, is that the attacker insert SQL commands into input field of web form or query string of page request to deceive the server and execute malicious SQL commands. To avoid sql injection attack , information inputed by user should be verified by RegularExpressValidator controls.Standard stored procedure is made on the basis of verification, at the same time, parameters and the corresponding columns should have the appropriate size, so hackers can not put large amounts of data or large blocks of text to the stored procedure [6]. Database table is accessed by executing a stored prvocedure, which can avoid direct access to table, achieve isolation of user and data, reduce the risk of application disclosure, and finally ensure the security of data. Another advantage of using stored procedures is that you can control the access right .Non-privileged user can not access it, So that it can ensure the safety of the program.

D. Enhance the security of value transmission

Data transfer is simple, for example, value is transferred using the Querystring , which is a more traditional mechanisms. But in the process of transferring data, data content in the URL bar can be seen by user. A malicious user can enter an address in the address bar, achieve unauthorized access to resources , and may damage these resources ,which are unsafe to the whole application. Therefore, this method is only used for pass-through not important infromation or a simple value. There are many methods to solve the security of passing values. For example, using of Session variables and server.Transfer methods achieve the value transformation from page to page. Under normal circumstances, a malicious user can not gain permission to access the resources.

E. Enhance the security of Viewstate

Viewstate is stored in one or more hidden fields of a web page of Asp.NET, it can be tampered with. Programmers normally like to use the hidden field to store a variety of information in the application based on Asp.NET. There are significant security risks, such as session information, shopping cart, and other information, etc., As long as users have little knowledge of html, they can save the html page on a local disk, remove all the JavaScript code and html code associated with the validation, and load the file into the browser. In this case, the contents loaded in the browser don't have of any validation code, so users can send any content they want to the server . Thus, great harm to the security of the application can be caused. The following principles can be used to solve this problem.In the actual programming

1) Always use Asp.NET Session object to check whether the user has a valid session

2) re-authenticate is needed On the server using Validation controls of Asp.net. Regardless of whether the client is validated, the data must be verified on the server

3) any sensitive information are not stored in the Hidden fields

4) Setting ViewStateUserKey property helps to prevent malicious users attacks by click . This property.must be handled in the Page_Init event page.

F. Custom encryption class

ASP. NET provides a complete set of encryption class hierarchy for the developer to complete a variety of encryption tasks. It consists of three levels.The first layer is a set of abstract base class. It consists the SymmetricAlgorithm class, AsymmetricAlgorithm class and HashAlgorithm classes. SymmetricAlgorithm class implements asymmetric encryption algorithm, AsymmetricAlgorithm class implements symmetric encryption algorithms, HashAlgorithm class implements a hash algorithm. The second layer of encryption is derived from the first layer , used to achieve a specified encryption algorithm. The third layer is a series of classes to achieve encryption.An example of non-symmetric encryption algorithm is given to briefly show how to make the custom encryption class . first create a static class AsymEncryUtility, then create several major mehtods of the class. Class diagram is shown as the following Figure 2.

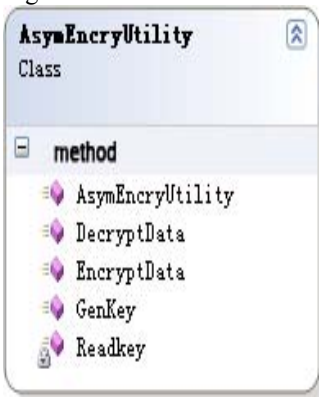

Figure 2. AsymEncryUtility class

GenKey()method creates an instance of an RSA encryption algorithm to generate secrekt key. Public key is returned as a string in this method. ReadKey () method is called by the decryption function to read secrekt key from the file, and initialize the encryption algorithm instance. EncryptData () method and DecryptData () method achieve encryption using public key and decryption using the secrekt key. Limited to space limitations, the above method implementation code are omitted.

IV. SUMMARIES

Security is a very important aspect of ASP. NET Web application. More aware Web application security principles, including on how to reduce common security threats, how to protect the Web application resources, and how to verify and authorize user, the better researchers can understand their content and Principle, which has important theoretical and practical significance.

REFERENCES

[1] Qu Weihua. ASP. NET Security Analysis. Taiyuan University [j]. 2009 Volume 10 No. 3 Total 39, pp.137-139

[2] Li Wei. Application design and implementation of security based on ASP.NET. Technology information Development and Economy [J]. 2008 Vol 18 No. 5, pp. 172-173.

[3] Suyuzhao, Zhao Yan. Design and implementation of user simulation based on asp.net security model [J]. Computer Information and Technology. PP. 26-29

[4] Gui Xueqin, Yuying Hong. The implementation of asp.net security scheme and form authentication [J]. Micro Computer Applications. 2005.1, pp.50-52.

[5] Zhang Limin. Learning asp.net with examples [M]. Wuhan: Huazhong University Press. 2006.6,pp663-665.

[6] Russ Basiura,Richard Conway. Professional Asp.Net Secury[M]. Beijing: Tsinghua University Press, PP.79-80

Published by Atlantis Press, Paris, France.