

Custom Membership provider without password management

Source:

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2008-01/msg00072.html>

- *From:* paolo_paganotto@xxxxxxxx
 - *Date:* Thu, 24 Jan 2008 08:06:50 -0800 (PST)
-

The next ASP.NET 2.0 web application I've to develop will be hosted in a Web Access Control environment so that authentication is performed by a front-end security system that checks the user credentials and, by means of a reverse proxy, redirects the request to my protected application.

In this context, I don't have to concern about authentication since every user that accesses the application is just authenticated and his data is contained into the HTTP header (user name, and metadata also, necessary for following authorization).

For this reason, the ASP.NET authentication method in Web.config is 'NONE'

The application manages protected resources so I need to manage users, roles and profiles.

I need to take care just of authorization but NOT authentication.

I want to use ASP.NET 2.0 membership provider model, but as you can imagine I need to completely get rid of password management. I don't need to authenticate users, so I don't even need to store passwords (nor is this allowed by the company's security policy).

I still need profiles and roles, so the role/profile providers are useful for me.

But to use them I also need to have a table with users, and methods to create/edit them, so I also need a membership provider.

In this context, I've met some troubles playing with SqlMembershipProvider.

In my derived class I can override the ValidateUser(...) method ignoring the 'password' attribute and creating a GenericPrincipal object to be assigned to System.Web.HttpContext.Current.User property. So far so good.

I can also ignore the methods dealing with password management (change, questions, etc.), but problems come when facing the CreateUser(...) method: it is not allowed to create an user without password (sounds logic, in a normal security environment).

The problem is the minRequiredPasswordLength property: it must be

Custom Membership provider without password management

greater than 0.

Ok, I can create users with a fake password and ignore it, but... sounds quite a dirty approach.

For modifying this rule, I figure I should either:

overload the Initialize(...) method rewriting it so that the property `minRequiredPasswordLength` is not initialized

or

overload the CreateUser(...) method rewriting it so that no check is done on the password provided

in that cases I can invoke the method with null, or `String.Empty` as value for password parameter.

I must also check whether the standard aspnetdb database allows empty or null passwords (I feed some doubt about it, but in any case I can use a different schema that simply ignores passwords).

The question is: does a derived class based on `SqlMembershipProvider` represents a valid approach for me, deserving a further study, or it's better to design a new provider?

If this is the case, I should create a class inheriting `MembershipProvider`, avoid implementing the methods dealing with passwords, and overriding every other methods, as `CreateUser(...)` method.

And... according to the db: is it better to create a new database with a schema that doesn't contain passwords, or just keeping the standard one?

any feedback will be appreciated sincerely

Paolo

.