

Re: Windows service

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2007-10/msg03583.html>

- *From:* "Bela Istok" <bela_i@xxxxxxxxxxxx>
 - *Date:* Fri, 26 Oct 2007 15:43:34 -0400
-

Hello Arnold, if you know all of this why you recommend to Rotsey not to use Domain Security?

Because the persons you have worked for doesn't do the work in the right way, doesn't mean you need to omit the solution at all?

Regards,

Bela Istok

"Mr. Arnold" <MR.Arnold@xxxxxxxxxxxx> wrote in message
news:uN7clMAGIHA.1208@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

"Chris Mullins [MVP - C#]" <cmullins@xxxxxxxxxxxx> wrote in message
news:uGgtDBAGIHA.3600@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

"Mr. Arnold" <MR.Arnold@xxxxxxxxxxxx> wrote

"Chris Mullins [MVP - C#]" <cmullins@xxxxxxxxxxxx> wrote
in message
news:eiaWX%23%23FIHA.1208@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

"Mr. Arnold" <MR.Arnold@xxxxxxxxxxxx>
wrote

The recommend practice is to use Windows Auth from the service to SQL Server. It's easier to configure, less maintenance, and much more secure. In general, you should not use SQL Auth for much of anything these days.

You might convince me on the little more secure, but not the less maintenance part of just having a generic user-id and

Re: Windows service

psw that an app uses.

Well, SQL Auth is considered to be insecure. It's easily cracked (I've used the tools, and verified this, much to my client's amazement), doesn't have any metering on it to prevent brute force attacks, transmits the credentials to the database in plain-text, and doesn't integrate at all into the standard security infrastructure already being used by the organization.

Most organizations don't know the first thing about security -- not really.

There no way to enforce password strength rules, and the passwords are store as case insensitive hashes. There's no default monitoring of the invalid password attempts, no automatic account lock-out, etc. There's a ton of documentation on this found on the web.

I know about all of this. I have been working on the NT based O/S platform for many years.

All of the DBA's that I work with prefer Windows Auth to SQL Auth for ease of use reasons as well. It's one less set of passwords to remember, less configuration in the long run, fewer plain-text passwords floating around in email & config files. It's much easier to switch between databases without having to continually enter passwords.

Most solutions are not swithing between databases. The only way that switching of databases by an application would be is when the appliication moves from dev/test/QA/prod, but they are the same database. The only thing that changes is is where the database is located.

It's also easier to manage in the HR case – when you fire an Admin or a Dev, you don't have to change all the SQL Passwords, just lock-out his account.

You and I both know that most are not doing this. :)

When you hire someone new, you don't have to email them the credentials. You just add the role to their Active Directory account, and it works.

I sounds good, but again, you and I both know that they are not doing this.

Re: Windows service

If you ever want to scare yourself, go download Cain & Able off the web, and watch the insecure traffic fly across your network. You'll have all the account credentials in no time. Makeing sure the network layer is secure is important!

I don't disagree with you. But they are not doing it, and I have been in those shops.