

Re: Web and SQL Security

Source: <http://www.derkeiler.com/Newsgroups/microsoft.public.sqlserver.security/2005-03/0292.html>

From: Chris Weber [Security MVP] (chris_at_dev.nul)

Date: 03/25/05

Date: Fri, 25 Mar 2005 09:39:45 -0800

Your connection string needs to be a low privileged account. Hopefully your developers were able to design it that way. How are they encrypted? Using DPAPI? It will still be much harder for an attacker to compromise the database if it's on a separate server. You've created a barrier by encrypting the connection string, so they won't be able to connect to SQL without it right? Or are you also using trusted connections?

I can't comment on 1 high power vs 2 low power, because I've no idea of the performance requirements and load the servers expect. You know, if there's a lot of database activity, it's best to separate the database files and transaction logs to separate disks and controllers ideally.

SQL auth is never recommended, Windows auth always is. I would say separate into two boxes if you can, harden both servers. Especially lock down SQL and the application – use low privileged accounts, run all requests through Stored procedures or parameterized queries. Apply strict permissions to the database and tables.

Your correct, separating into two servers won't buy you much unless you properly design the application, host configurations, and account access permissions.

Have a pen-test run on your web-app, that's usually where most holes into the network are found.

Check out this guide for more details.

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/ThreatCounter.asp>

/Chris Weber

"David" <Dante@community.nospam> wrote in message
news:C53BF10D-C682-4E7E-A1B3-78598427EECC@microsoft.com...

> *Hi Chris*

>

> *The issue here is that we do not hold the boxes. They will be at a (high
> quality) ISP and managed by them. I understand that typically hosted web
> and*

> *SQL servers reside behind the same firewall configuration. A two tier one*

> in
> this case. Therefore (.NET) web app communicates with the SQL server using
> sql authentication.
>
> I guess my point is that if the two servers are behind the same firewall
> system, then if the web server is compromised, it won't take much to get
> to
> the SQL servre. The connection strings are encrypted of course, but ...
>
> Basically there is a cost issue. We can two low power boxes, 1 for the web
> and the other for SQL or we can one high power box to do both jobs.
> ISecurity
> is the issue that will determine wich setup we go for.
>
> Any comment on this would be much appreciated.
>
> Thanks
>
> David
>
> "Chris Weber [Security MVP]" wrote:
>
>> This has always been a recommendation from the security community. the
>> issue is that separating roles is a security practice – you DON'T want to
>> host your database on the same server that hosts your Web server. Surely
>> however, you would apply proper Firewall rules that only allow inbound
>> TCP
>> 80 and 443, and not 1433. The reasons for separation are numerous.
>> For
>> example, a vulnerability in IIS would lead to a direct compromise of the
>> data.
>>
>> This issue is largely dependent on the application's design. Are you
>> allowing Anonymous access? Then your chances of getting compromised are
>> that much greater.
>>
>> Honestly, this recommendation was originally conceived from the notion of
>> separating application components – one that serves web pages and one
>> that
>> holds data. But it was also conceived during the early days of IIS 4/5
>> when
>> vulnerabilities were very severe and seemed to come out every week.
>> IIS6
>> is much stronger.
>>
>> You could get away with it on one server, but you need to lock down IIS,
>> SQL
>> permissions, and the application's functionality as much as possible.
>> If you can afford two boxes and separate them by a firewall – DO IT.
>> But remember, if your making your connection from IIS to SQL as a full
>> sysadmin or dbo level, then once your IIS box gets compromised, the

>> *hacker*
>> *will likely have access to the database with that level of permission.*
>> *SO,*
>> *USE a LOW PRIVILEGED account for data access.*
>>
>> *The majority of attacks today are exploiting poorly written*
>> *web-applications, not the underlying infrastructure so much.*
>>
>> */Chris*
>>
>>
>>
>>
>>
>> *"David" <Dante@community.nospam> wrote in message*
>> *news:E3F758FD-A178-4DC9-8CB1-2567F9DA9468@microsoft.com...*
>> > *Hi*
>> >
>> > *I know that a couple of years ago I read a Microsoft recommendation*
>> > *that*
>> > *SQL*
>> > *server should not run on the same machine as IIS.*
>> >
>> > *We are looking at taking a managed hosted server for an app. and I*
>> > *wondered*
>> > *if the same recommendation applies. Does it depend on the way the*
>> > *hosting*
>> > *company sets up the server or is it always less secure when the two are*
>> > *on*
>> > *one machine?*
>> > *We can have two less powerful machines or one more powerful machine to*
>> > *do*
>> > *the job and security is the thing that will determine which way to go.*
>> > *We*
>> > *wil*
>> > *use Windows Server 2003, SQL Server 200 and .Net Framework.*
>> >
>> > *Any thoughts appreciated.*
>> >
>> > *David*
>>
>>
>>