

## Re: Connecting to a remote sbs from a website?

---

*Source:*

<http://www.tech-archive.net/Archive/Windows/microsoft.public.windows.server.sbs/2008-07/msg00894.html>

---

- *From:* Joe <joe@xxxxxxxxxxxxxxxx>
  - *Date:* Thu, 10 Jul 2008 22:06:34 +0100
- 

T.Mthinkulu wrote:

Hi Cliff,

Thanks for that..!Yes,I am running a SBS premuim 2003.By creating a new instance,do you mean,creating a new table?By the sounds of it,there wont be any implications of connecting to a remote SBS then?What are the normal required parameters for making an SQL server connection?

If you don't know that...

I've never done it, but I'd assume you need to log on to SQL Server with a valid SBS user name and password, one that has the required read and write permissions on the database table(s). Your programming/scripting language will have library calls to make and break the connection and to carry out SQL transactions.

Please bear in mind that what you are doing is the equivalent of running, not with scissors, but with several open cut-throat razors. An active web page linked to an SQL database running on an organisation's main/only business server is just the sort of thing a cracker would be overjoyed to find. I take it you know all there is to know about cross-site scripting and SQL injection?

I've only done this once, and here's the coward's way in outline:

- A master SQL database running on the in-house server with \*no\* remote access allowed
- A slave SQL database running on a commercial web host's server
- Timed scripts run on the master to update the slave with any changed records on the master, via email

At no time is anything open on the master or slave servers, all data movement is initiated from the servers themselves. And if the slave ever gets cracked, the web host re-images it and the full database gets backed up on the master and restored to the slave. No data is ever passed back to the master in this system, though this would have been easy to do, again using email but in the other direction. A timed script running on the receiving server pulls in the email from an IMAPS server and updates the database from it, after certain integrity checks.

It's not the only way, nor even the best, but it's resilient and pretty much bomb-proof and requires no listening servers, no open ports at either end except the slave web server itself. Best of all, absolutely nothing that happens to the web server can in any way compromise the in-house server, no matter how full of holes my

## Re: Connecting to a remote sbs from a website?

web scripts may be.

Occasionally, the master system loses Internet access, but just sends out the queued emails when it's back online. It's been running with only minor database housekeeping for nearly two years, without any faults or data loss. It doesn't happen to run on Windows, for a reason not related to security, but I'm sure a Windows equivalent would be no more difficult to organise.

.