

## Re: SQL2K WIN2K3 CONNECTION SECURITY

*Source:* <http://www.derkeiler.com/Newsgroups/microsoft.public.security/2005-06/0293.html>

---

*From:* Mercury ([me\\_at\\_spam.com](mailto:me_at_spam.com))

*Date:* 06/12/05

Date: Mon, 13 Jun 2005 00:18:00 +1200

My own benchmarking indicates that TCP/IP is faster than named pipes. Named pipes is a layer on top of whatever protocols are available. So, being a layer it adds to the response time ==> decreases performance. The effect is small, but measurable.

I suggest that unless you have reasons to be concerned about performance, that you secure the system first, get it working, verify it works, verify the security, then benchmark real performance – if it is a concern then take the benchmark stats as the starting point...

If you have performance issues, the place to start is those improvements that will result in large improvements, not tiny ones such as TCP/IP vs. Named Pipes. The performance of SQL Queries will have an exceedingly greater impact on response times than anything else (probably...). Missing indexes will show up when you use the correct performance monitoring tools EG Perfmon and SQL Server Profiler along with SQL Server Query Analyser (QA). Use QA to tune queries – assess the poor performing ones, use Execution Plans to show you which parts of the queries are consuming the most resources. Profiler is important as it will show you the elapsed execution time of queries, CPU, Read and Write IO's and the statements executed. You can record the traces in Profiler for rerunning on a test system to facilitate performance tuning.

*>From the security perspective I suggest you research DMZ – get (if budget permits) 2 h/w firewalls and configure them as you describe – Ports 80 and/or 443 in to the Web machine, and 1433 only from the web to the DBMS server. The trouble with s/w firewalls is that if your system gets infected, your firewall can be torn down at the same time. You will need SQL Server authentication for this. The DSN or Username / password should be stored encrypted on the web server.*

There is substantial information available on hardening systems for such use. Research at MS, on google, or take a look here:

<http://www.nsa.gov/snac/>

Since you are asking these questions, then no answer would be complete without asking you if you have defensively coded your system against code injection attacks and cross site scripting attacks? If you cannot

authoritatively say Yes 100%, then it would be more than prudent to attend to this important sources of site / system hacks promptly.

I suggest you consider architecting the web server setup such that it is a throw away system. If you ever suspect it is infected then fdisk and restore or rebuild. Design it so that you never need to backup the web server once it has touched the internet.

HTH.

<jens.aggergren@lycos-europe.com> wrote in message  
news:1118378086.080331.325050@g49g2000cwa.googlegroups.com...  
This question got rejected from the SQL Server group, but i'll try here  
as it relates to security.

I moving an old SQL Server-backend-IIS5/ASP-fronte--nd application to servers with windows 2003 standard edition. One server will run the database the other will run IIS 6.0. Note that i haven't set-up a domain, which i think requires one machine to be domain controller which would decrease performance and stuff. I've simply put them on the same group.

I wan't to restrict access to the sql server so only the incomming connection from the webserver is allowed. I can use either named pipes(which should be the fastest protocol) or tcp(which should be slight slower than named pipes) but I seem to have a problem. If I use named pipes to connect, the IUSR(the user under which IIS is running) must have access-rights to IPC\$ share on the sql server.

I can't seem to set any access-right directly for IPC\$ share, but I can reactivate my guest user and then it works, but then everyone can now access the ipc\$ share so it's not really what i'm looking for.

I can also connect through TCP( and set up some kind of filter only allowing incomming connections on port 1433 from the ip of the web server. But i don't know how to do this. I've taken a look at the IPSec stuff but it's all about kerberos authentication and other bull which i don't think i need.

What i need is a simply ip port filter, which does nothing else but reject incomming connections to sql server on port 1433 originating from any other ip's than my webserver.

My question is how do I do this? Do i need to have a additional "firewall" service running and, if so, how much extra overhead will this create for the sql server.

Alternately, is it possible to change the access rights for the IPC\$ share manually?

Thanks in advance for any input you might have on this?