

SQL Server First Responder Kit

This is a living document. Make it your own!

Instance Name: _____

Triaged By: _____

Date and Time: _____

1) Can you connect ?

Connect to SQL Server and run:

```
SELECT name, user_access_desc, state_desc, log_reuse_wait_desc  
FROM sys.databases;
```

Did you connect successfully? _____

Anything notable in returned columns? _____

Problems connecting?

Try the dedicated admin connection: <http://BrentOzar.com/go/DAC>

2) Who's running?

Gather key pieces of information using Adam Machanic's sp_whoisactive:

```
EXEC dbo.sp_whoisactive;
```

Did you run this successfully? _____

How many rows did it return? _____

Was blocking present? (blocking_session_id column) _____

More info: <http://BrentOzar.com/go/active>

3) What does Brent say?

Check activity over a short sample using sp_AskBrent@:

```
EXEC dbo.sp_AskBrent @ExpertMode=1, @Seconds=10;
```

Did you run this successfully? _____

Did it find running queries? _____

Did it make a diagnosis? _____

More info: <http://BrentOzar.com/AskBrent>

4) Identify root cause of blocking (if present)

If there's blocking, save off the sql_text from the root query of the blocking. Get a screenshot that records the runtime, host name, and other relevant information from sp_whoisactive.

If you didn't catch it in time, consider setting up alerts or the Blocked Process Report. More info at: <http://BrentOzar.com/go/lockthis>

SQL Server First Responder Kit

5) What's SQL Server's Error Log tell you?

SQL Server Management Studio: 'Management' → 'SQL Server Logs'.

Or query the log with this command:

```
EXEC xp_readerrorlog @p1=0, @p2=1
```

Check all logs since at least the last startup and just prior.

(Use @p1=2, @p1=3, etc to access prior logs.)

- Any recent errors/login failures? _____
- When was the last startup? _____
- Was the last restart planned? _____

6) What's the Windows Event Log got to say?

Are there events in the windows logs at the same time or just before the problem periods?

System log: _____

Application log: _____

Security log: _____

Remember, particularly in the system log, informational events may help explain errors you're seeing. *Don't filter and look at ONLY errors all the time!*

7) Capture SQL Server Activity to a Table

Run sp_AskBrent® in a loop to log activity to a table. This gathers data while you keep looking.

Activity is being captured to the table: _____

```
EXEC sp_AskBrent @OutputDatabaseName = 'DBATools',  
@OutputSchemaName = 'dbo', @OutputTableName = 'AskBrentResults'
```

More info: <http://BrentOzar.com/AskBrent>

(If you use this option, make sure to clean up old data from that table!)

8) Capture SQL Server Overall Waits

Find the top three SQL Server waits in sys.dm_os_wait_stats by percent since the last restart using <http://BrentOzar.com/go/getWaits>

Or find high waits captured by sp_AskBrent® within 15 minutes of a time (if you were running it in a loop):

```
EXEC sp_AskBrent @AsOf = '2013-10-15 18:45', @OutputDatabaseName =
```

SQL Server First Responder Kit

'DBAtools', @OutputSchemaName = 'dbo', @OutputTableName =
'AskBrentResults'

9) Review Performance Counters

Look at performance counters (and, ideally, their history) to identify system performance patterns and problems.

Performance counters checked? _____

Notable counter info : _____

More info: <http://BrentOzar.com/go/perfmon>

10) Make an assessment by major areas. Can you identify a bottleneck specific to one of these?

CPU: _____

Memory: _____

Network: _____

Disk: _____

11) Identify Recent changes

Have there been recent changes in these areas?

App Tier: _____

Stored procedures: _____

Schema changes: _____

Index changes: _____

Major data changes: _____

Infrastructure changes: _____

Maintenance changes: _____

SQL config changes: _____

Windows config changes: _____

References:

Adam Machanic created sp_whoisactive:

http://sqlblog.com/blogs/adam_machanic/default.aspx

Latest version of this checklist: <http://BrentOzar.com/responder>