Description

Zabbix templates for Microsoft SQL Server (MS SQL).

Features

- MS SQL performance counters.
- MS SQL instance Low Level Discovery.
- MS SQL database Low Level Discovery.
- MS SQL agent job Low Level Discovery.
- MS SQL database backup monitoring.
- MS SQL database mirroring monitoring.
- MS SQL Always On monitoring.
- MS SQL Log Shipping monitoring.

Supported versions

Tested on Microsoft SQL Server 2012, 2014, 2016 and 2019. It may work with earlier versions, but some items (with missing performance counters) may be unsupported. For the extensive overview on the performance counters difference between MS SQL 2008 and MS SQL 2012 you can read here (https://blog.dbi-services.com/sql-server-2012-new-perfmon-counters/).

Tested on Zabbix 3.4.0. and 4.0.0. It may work with earlier versions, but some items (for example service.info[service,<param>]) may be unsupported. The template was started on Zabbix 2.4.0 but after each new Zabbix version, objects were modified or new things were added.

Assumptions

These templates are not a one-click solution. It may require some tinkering, so I assume that:

- You are familiar with a Microsoft Window OS.
- You are familiar with a Microsoft SQL server.
- You are familiar with a Zabbix.
- You have a correctly working Zabbix server and Zabbix agents.
- You have tested other Zabbix template with an item type “Active” and it works.
- Zabbix agent is installed to a location C:\Program Files\Zabbix\. If not, you will have to change the paths everywhere

Template includes

Templates:

- “Template Microsoft SQL Server DE Tier 3.xml” – Template for Microsoft SQL Server Database Engine. Database status and backup status.
- “Template Microsoft SQL Server DE Tier 2.xml” – Template for Microsoft SQL Server Database Engine. Instance and database statistical information.
- “Template Microsoft SQL Server DE Tier 1.xml” – Template for Microsoft SQL Server Database Engine. Instance and database performance information.
- “Template Microsoft SQL Server SA Tier 3.xml” – Template for Microsoft SQL Server Agent.

Value mapping:

- “SQL Agent Job status.xml” – Zabbix value mapping for Microsoft SQL Server Agent Job status.
- “SQL Database status.xml” – Zabbix value mapping for Microsoft SQL Server Database status.

Scripts:

- \MSSQL\DiscoveryDatabaseAgent\ – PowerShell scripts MS SQL Agent Low Level Discovery.
- \MSSQL\DiscoveryDatabaseBackup\ – PowerShell scripts for MS SQL Backups Low Level Discovery.
- \MSSQL\DiscoveryDatabaseBasic\ – PowerShell scripts for MS SQL Database Low Level Discovery.

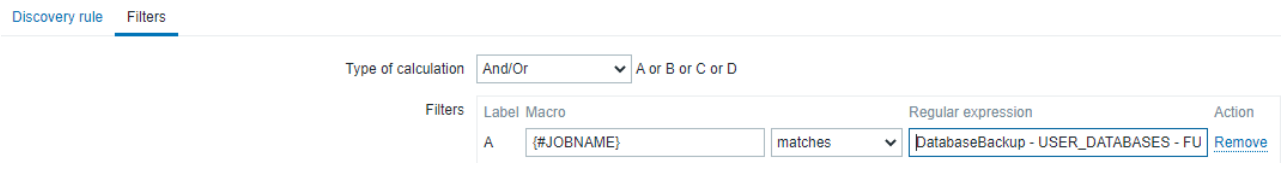
Deployment. Step by step

1. Import templates via Configuration >> Templates:
   - “Template Microsoft SQL Server DE Tier 3.xml”
   - “Template Microsoft SQL Server DE Tier 2.xml”
   - “Template Microsoft SQL Server DE Tier 1.xml”
   - “Template Microsoft SQL Server SA Tier 3.xml”
2. Import value mappings via Administration >> General >> Value mapping:
   - “SQL Agent Job status.xml”
   - “SQL Database status.xml”
3. Copy catalog MSSQL with PowerShell scripts (*.ps1) to a location a Zabbix Agent can access (by default “C:\...\Zabbix\bin\”).
4. Copy 3 *.conf files from catalog “User parameters” to a location a Zabbix Agent can access (by default “C:\...\Zabbix\”).
5. Update “zabbix_agentd.win.conf”:
   - add line “Include= C:\Program Files\Zabbix\mssql.agent.userparams.conf”.
   - add line “Include= C:\Program Files\Zabbix\mssql.backup.userparams.conf”.
   - add line “Include= C:\Program Files\Zabbix\mssql.basic.userparams.conf”.
6. Grant rights for Zabbix Agent service account. It needs read rights on tables:
   - msdb.dbo.sysjobhistory
   - msdb.dbo.sysjobs
   - master.sys.databases
   - msdb.dbo.backupset
   - msdb.dbo.log_shipping_monitor_secondary.
7. By default, Zabbix Agent service account is NT AUTHORITY\SYSTEM which is already in SQL Server. If you need to monitor mirrored databases or databases in Always On, you will have to give Zabbix Agent’s service account (NT AUTHORITY\SYSTEM by default) sysadmin rights. More about it here.
8. Restart Zabbix Agent.
9. Depending on your SQL server edition and monitoring requirements select and add templates to a host.
10. Modify macros in templates according to your needs. Default values are below:

Macros	Macros meaning	Value	Meaning	Trigger
{$SYSDBFTIME1}	Sys db full backup time value 1	25	25 hours	Information
{$SYSDBFTIME2}	Sys db full backup time value 2	50	50 hours	Low
{$SYSDBFTIME3}	Sys db full backup time value 3	75	75 hours	Medium
{$UDBDTIME1}	User db diff backup time value 1	48	2 days	Information

{$UDBDTIME2}	User db diff backup time value 2	72	3 days	Low
{$UDBDTIME3}	User db diff backup time value 3	96	4 days	Medium
{$UDBFTIME1}	User db full backup time value 1	168	7 days	Information
{$UDBFTIME2}	User db full backup time value 2	192	8 days	Low
{$UDBFTIME3}	User db full backup time value 3	216	9 days	Medium
{$UDBLTIME1}	User db log backup time value 1	30	30 minutes	Information
{$UDBLTIME2}	User db log backup time value 2	60	60 minutes	Low
{$UDBLTIME3}	User db log backup time value 3	90	90 minutes	Medium
{$EVENTLOGTIME}	Event log recovery time value	28h	28 hours	Medium
{$DAYS}	Maintenance job time value	7	7 days	None

11. "Template Microsoft SQL Server SA Tier 3.xml" lets you discover SQL agent jobs. Discovery rules consist of:
    - "SQL Server Agent Discovery" – discover SQL Agent service.
    - "SQL Server Agent Jobs P1 Discovery" – discover SQL Agent jobs.
    - "SQL Server Agent Jobs P2 Discovery" – discover SQL Agent jobs.
    - "SQL Server Agent Jobs P3 Discovery" – discover SQL Agent jobs.
12. Difference between "SQL Server Agent Jobs P1 / P2 / P3 Discovery" are triggers. They can be configured differently. For example:
    - "SQL Server Agent Jobs P1 Discovery" – alerts after trigger failed. Good for monitoring jobs, which need immediate attention. Like failed job "CHECKDB".
    - "SQL Server Agent Jobs P2 Discovery" – alerts after trigger failed two times. Good for monitoring jobs, which need attention, but not immediate. For example, job "DB LOG BACKUP" failed 1st time, but it will run again in 30 minutes. If 2nd time it fails again, then alert is raised.
    - "SQL Server Agent Jobs P3 Discovery" – alerts after trigger failed but with additional conditions. Good for monitoring jobs, which do not need immediate attention. Like failed job "IndexOptimize". Alert will be raised only during Monday – Friday, during 08:00 – 16:00. If you want to change day and hour parameters, you can do it directly in triggers.
    - In mssql.agent.userparams.conf I placed 2 additional user parameters. In case you need to create your own custom items for monitoring P(riority)4 and P(priority)5 jobs.
13. Every discovery rule "SQL Server Agent Jobs P1 / P2 / P3 Discovery" has its filters there you can enter the job name, you want to associate with a selected rule:



If you leave a filter empty, all agent jobs will be discovered. To avoid that, I entered a simple place holder for every rule – ENTER_JOB_NAME.

Notes

1. MS SQL system databases (master, msdb etc.) are discovered and monitored by default. I saw templates, where system databases there excluded from discovery and monitoring, but I am against it. You need to know the status (Online, Offline, Backup) and performance parameters of system databases as well.
2. MS SQL Agent job status is queried from table msdb.dbo.sysjobhistory. If the job is never run and no record exists in this table, Zabbix item will be unsupported.
3. MS SQL backup history is queried from table msdb.dbo.backupset. If there is no info in this table or the info is incorrect, template may not work correctly. I know there are 3rd party backup solutions, which may insert incorrect values or do not insert at all.
4. If you want only a specific host to have other macros values than default, you can change them in that host's macros section.
5. Zabbix value mapping section does not support macros, therefore, if you change a default macros values for a template or a specific host, remember to update "SQL Database Backup status" via Administration >> General >> Value mapping.
6. If you have a non-default Zabbix installation directory or you want to put PowerShell scripts to other directory, you will have to update files and directories.
7. This template does not provide values every second, but only every 30 sec. For some items it is even a larger item update period. If you want to have values every second, you should probably use other tools like WinOS performance monitor, MSSQL profiler or trace. But these tools, gathering values every second, puts a stress on the system (OS or MSSQL). I would use Zabbix with this template, to have a general understanding about how my systems are doing. In case of alert, I would switch to more precise tools
8. The triggers settings are configured for an ideal MS SQL performance, which is hardly achieved in a real environment. Therefore, I strongly advise you to review all the triggers, before going live. You can either lower severity or change trigger value.
9. Items Update interval (30 seconds), History storage period (90 days) and Trend storage period (365 days) are set accordingly to the rule "as much as possible, for as long as possible". If you have ~10 instances with an average ~8 databases per instance you will be fine. But if you have more than that, you should check "Zabbix calculations for MS SQL template.xlsx". Enter your values and see how much disk space for a Zabbix database you will need. Also, depending on the number of processed values per second (NVS), you may consider upgrading your Zabbix server or increasing Items Update interval from 30 seconds to any value you feel comfortable. All calculations are based on Zabbix manual and they provide only average values, not the exact numbers.
10. Because performance counters items are with English name key, template will work with servers, which have English locale. Otherwise you will have to translate these items to the language, you are going to monitor.

Thanks

MS SQL performance counters selection is based on a 2010 Quest Software poster written by Kevin Kline (MVP) with Brent Ozar (MCM, MVP) and contributions by Christian Bolton (MCM, MVP), Bob Ward (Microsoft), Rod Colledge (MVP), and Raoul Illyaos. You can find original poster by googling “quest sql server performance counters of interest”.

PowerShell scripts are based on the scripts of Nate Jones, Jon Pangburna, Александр Александров (https://share.zabbix.com/databases/microsoft-sql-server/mssql-2008-2016-multiinstance).

Thanks for the additional ideas and help with the scripts to Domantas P., Jonas S., Pavel L., Oleg D., Nerijus P., Klaidas I., Mark F.

Contacts

Let me know if you find any errors. Or maybe you just have a great idea which really, really must be added to the templates.

mantas.tumenas@gmail.com