Manage Your Shop with Policy Based Management & Central Management Server

Ryan Adams
Blog - http://ryanjadams.com
Twitter - @ryanjadams
Email – ryan@ryanjadams.com
Objectives

- CMS Configuration
- CMS Import and Export
- Policy Creation
- Policy Evaluation
- Alerts
- Reporting
Central Management Server

CMS stores connection information for all your SQL servers in one central location.

It’s just like registered servers but is housed in MSDB
Requirements

- Must be on a SQL 2008 Instance
  - All editions are supported including Express
  - You can register SQL 2000 and 2005 instances
- Security managed by two MSDB database roles
  - ServerGroupAdministratorRole
    - Allows management and configuration of the CMS
  - ServerGroupReaderRole
    - Allows connect and read to the CMS
Advantages

 ✓ Connection repository that all your admins can use

 ✓ Allows you to run a query against multiple servers simultaneously

 ✓ Allows you to group servers logically. Examples are by function, geographic location, or version

 ✓ Allows you to evaluate PBM policies against groups of servers
Disadvantages

• The CMS server cannot be a part of its own group
  • Workaround is to use 127.0.0.1 or an FQDN

• Registered servers can only use Windows authentication. Why is this a disadvantage?
Import and Export

Import your locally registered servers

Export servers from your CMS groups

Exports are stored in XML format

- Exporting Local Connections
  - Windows Authentication
    - Username and password are not exported
  - SQL Authentication
    - Username exported, and password exported as hashed
Jump In!!

DEMO
Policy Based Management

Gives us centralized management of our SQL Servers

Allows us to evaluate, configure, and enforce standards across the enterprise.
Requirements

- SQL Server 2008 Standard, Enterprise, or Developer
  - You can evaluate policies against SQL 2000 and 2005 instances, but SQL 2008 is required for the policy server
- MSDB
Terminology

- Facet
- Condition
- Policy
- Target
- Server Restriction
- Category
Creating Policies

Manual Policy Creation

- GUI
  - Create a check condition
  - Create a policy
    - Define check condition created in previous step
    - Define target condition
    - Define evaluation mode
    - Define server restriction condition

- T-SQL
  - dbo.sp_syspolicy_add_condition
  - dbo.sp_syspolicy_add_object_set
  - dbo.sp_syspolicy_add_target_set
  - dbo.sp_syspolicy_add_target_set_level
  - dbo.sp_syspolicy_add_policy
Creating Policies

Import

- Import policies exported from other servers
- Import Microsoft best practice policies
  - `%installdir%\100\tools\policies\databaseengine\1033`

- Option – Replace duplicates with items imported
- Option – Policy state
  - Preserve state
  - Enabled
  - *** Disabled ***
Policy Evaluation

• Evaluation Modes
  • On Demand
  • On Schedule
  • On Change: Log Only
  • On Change: Prevent
Policy Evaluation

• Methods
  • Evaluate a single policy against a single instance
  • Evaluate multiple policies against a single instance
  • Evaluate a single policy against multiple instances
  • Evaluate multiple policies against multiple instances
Alerts

- Requirements
  - Database Mail
  - Operators
  - Enable alert system notifications in SQL Agent
  - Policy must be enabled to raise an alert

- PBM Error Numbers
  - On Change: Prevent (automatic) – 34050
  - On Change: Prevent (on demand) – 34051
  - On Change: Log Only – 34053
  - On Schedule – 34052
Enterprise Policy Management Framework is an open source reporting project for PBM available on CodePlex.com

**Requirements**
- SQL 2008 SP1 CU3
- PBM
- CMS
- PowerShell
- Management Database
- SQL Reporting Services

**Setup**
- SQL script to create the database
- PowerShell script to evaluate policies against CMS
- BIDS reporting project
Jump In !!

DEMO
<table>
<thead>
<tr>
<th>Use Cases</th>
<th>Authentication Mode</th>
<th>SQL Password Expiration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable Auto Shrink</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SQL Password Policy</td>
<td>Guest Permissions</td>
<td>Last Backup Time</td>
</tr>
<tr>
<td>Auto Grow</td>
<td>Auto Create Statistics</td>
<td>Auto Update Statistics</td>
</tr>
<tr>
<td>Enforce Naming Convention</td>
<td>DB Compatibility Level</td>
<td>Recovery Model</td>
</tr>
<tr>
<td>Encryption</td>
<td>Is Trustworthy</td>
<td>Login Auditing</td>
</tr>
<tr>
<td>Enable Database Mail</td>
<td>Disable SQL Mail</td>
<td>Backup Compression</td>
</tr>
<tr>
<td>SQL Agent Running</td>
<td>Agent Jobs Notify on Failure</td>
<td>Monitor SQL Agent Jobs</td>
</tr>
<tr>
<td>Transaction Log Backups</td>
<td>Disable xp_cmdshell</td>
<td></td>
</tr>
</tbody>
</table>
Summary

- CMS Configuration
- CMS Import and Export
- Policy Creation
- Policy Evaluation
- Alerts
- Reporting
Resources

• Pro SQL Server 2008 Policy-Based Management
  • Ken Simmons
    • Twitter – @KenSimmons
    • Web – CyberSQL.blogspot.com
  • Colin Stasiuk
    • Twitter - @BenchmarkIT
    • Web – BenchmarkITConsulting.com
  • Jorge Segarra
    • Twitter - @SQLChicken
    • Web – SQLChicken.com

Slides
http://www.ryanjadams.com/presentations

Speaker Rate
http://speakerrate.com/ryanjadams

Ryan Adams
Blog - http://ryanjadams.com
Twitter - @ryanjadams
Email – ryan@ryanjadams.com

QUESTIONS?