Group Policy Compliance Manager 1.6 Deployment Guide

Information on installing and deploying GPCM 1.6

SDM Software, Inc.

May, 2017

Contents

Overview

Group Policy Compliance Manager (GPCM) 1.6 introduces a new architecture for more scalable collection of Group Policy health and settings data. The architecture now supports both remote collection (pull-based) and agent collection (push). Agents can be installed on any Window Server or desktop system, and will report up to one or more "Consolidator" servers, whose job it is to take collected data and add it to a SQL Server database instance. This document describes the details and requirements around installing GPCM and its constituent components.

GPCM 1.6 Setup

The GPCM 1.6 setup launcher (GPCM1.6Setup.exe) presents the following options on starting:

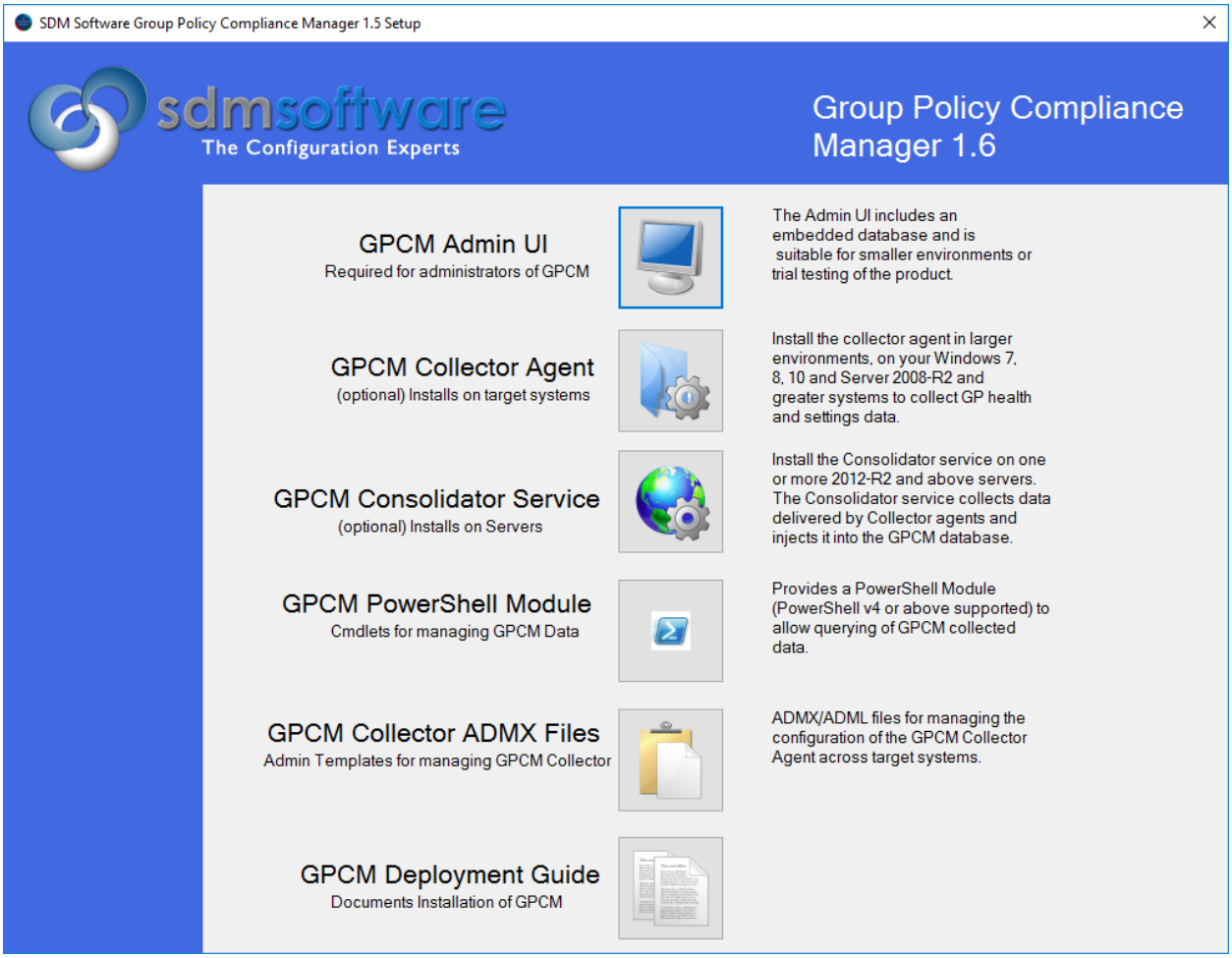

Figure 1 GPCM 1.6 Setup Launcher

Here, each option is described:

GPCM Admin UI: The GPCM Admin UI installer installs the GPCM console. The console can be used to view and collect GPCM data from endpoints. The console supports either using an embedded SQL LocalDB database, stored locally, or a SQL Server instance that has been previously installed. The GPCM

Admin UI is your main interface for interacting graphically with collected data. In a SQL Server-based deployment, the GPCM Admin UI can be installed on any number of systems, all pointing at a single SQL Server repository. If you are not deploying the SQL Server option (recommended for any environments greater than just a handful of systems), then each SQL LocalDB instance installed with the Admin UI knows only about the data that it has collected. The GPCM Admin UI requires .Net 4.5.2 Framework and requires that Microsoft Group Policy Management Console (GPCM) be installed as a prerequisite. The GPCM Admin UI can run on any version of Windows client or server from Windows 7 on up to Server 2016 and Windows 10.

GPCM Collector Agent: The GPCM Collector Agent is used solely in conjunction with a SQL Server deployment of GPCM. The collector agent can be installed on a Windows Server or desktop system. It installs as a Windows Service (see Figure 2).

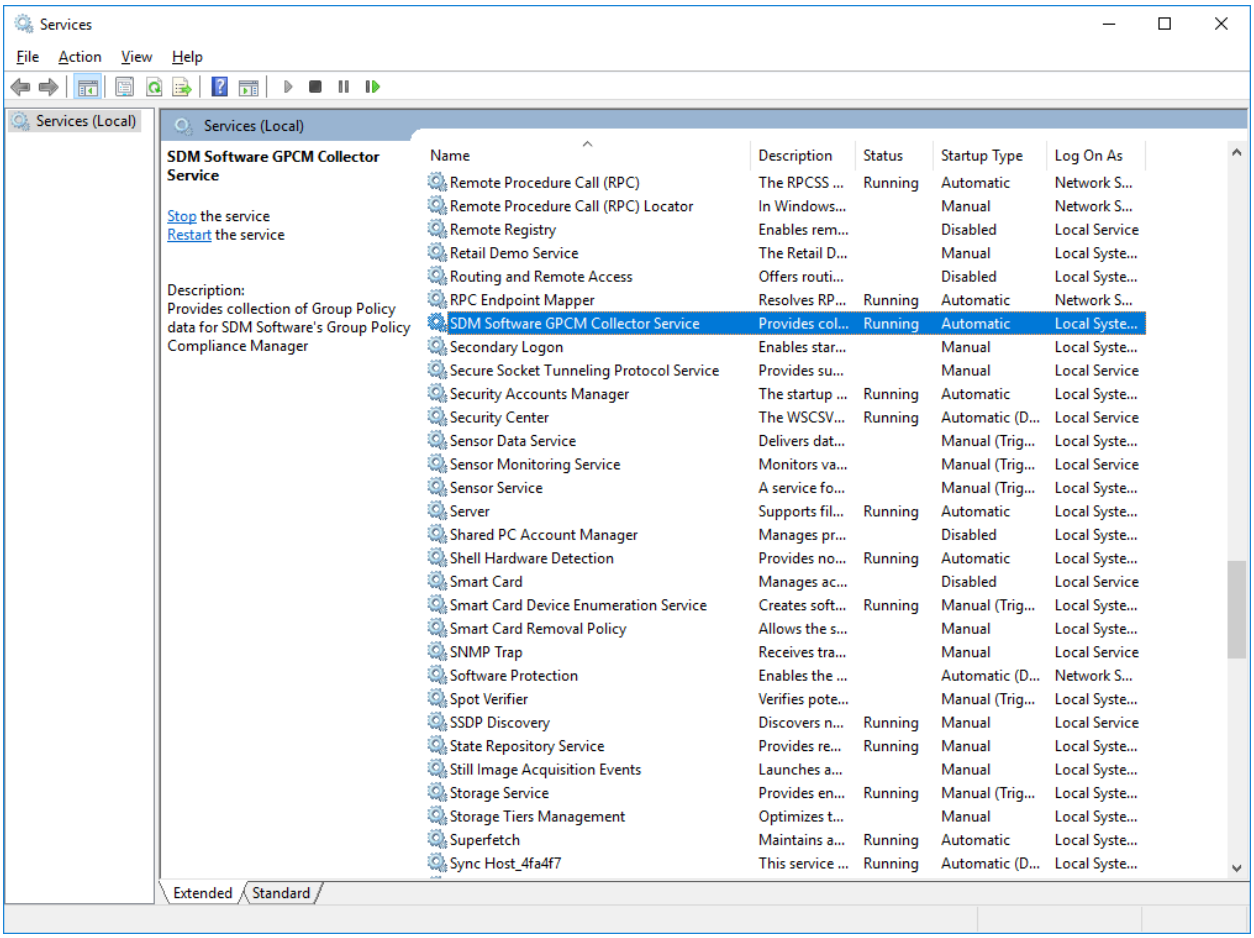

Figure 2Viewing the GPCM Collector Agent

The Collector Agent periodically collects GPCM health and settings data and copies it to an SMB share for eventual collection and injection into the GPCM database by the GPCM Consolidator Service (described next). The Collector Agent gets its configuration information via Group Policy, using the included GPCM ADMX files that allow you to deploy GPCM Collector information such as frequency of collection and which SMB share to send data to. The GPCM Collector Agent runs in the context of LocalSystem (i.e. the computer account) and thus no service account is required to run it. In addition,

the GPCM Collector Agent requires .Net Framework 4.5.2 be installed on each target system where it is installed.

GPCM Consolidator Service: The GPCM Consolidator Service is a Windows Service that is responsible collecting data that has been delivered via the GPCM Collector Agent, and injecting it into the GPCM database. The GPCM Consolidator Service requires .Net Framework 4.5.2 and Microsoft Group Policy Management Console (GPMC) be installed where it is deployed. The GPCM Consolidator Service can be installed on **Windows Server 2012-R2 and later**. The GPCM Consolidator Service runs in the context of **a domain user account** that has local administrator access and has read/write access to the GPCM SQL Server Instance. For more information on the deployment and configuration of the GPCM Consolidator Service, see the section below on "GPCM Deployment Scenarios".

GPCM PowerShell Module: The GPCM PowerShell Module is a module for interacting with and querying GPCM-collected data from the PowerShell command line. The module includes 6 PowerShell cmdlets that let you do everything from query computer and user information to comparing settings values to searching for specific settings that have been delivered to collected systems. The Module requires PowerShell v4 and the Microsoft Group Policy Management Console (GPCM) as pre-requisites. If you plan to use the module against the local SQL LocalDB instance, you should install it on the same system that you installed the GPCM Admin UI.

GPCM Collector ADMX Files: The GPCM Collector Agent configuration is controlled via the GPCM Collector ADMX files. This option includes a zip file that contains two ADMX/L files—**SDMGPCMCollector.ADMX** and the corresponding US-English (en-us) SDMGPCMCollector.ADML file. These files should be copied to either your ADMX Central Store within SYSVOL or into the local c:\windows\policydefinitions folder on the machine where you edit Group Policy. The setting options will appear under **Computer Configuration\Policies\Administrative Templates\SDM Software\GPCM Collector Service**, as shown in Figure 3:

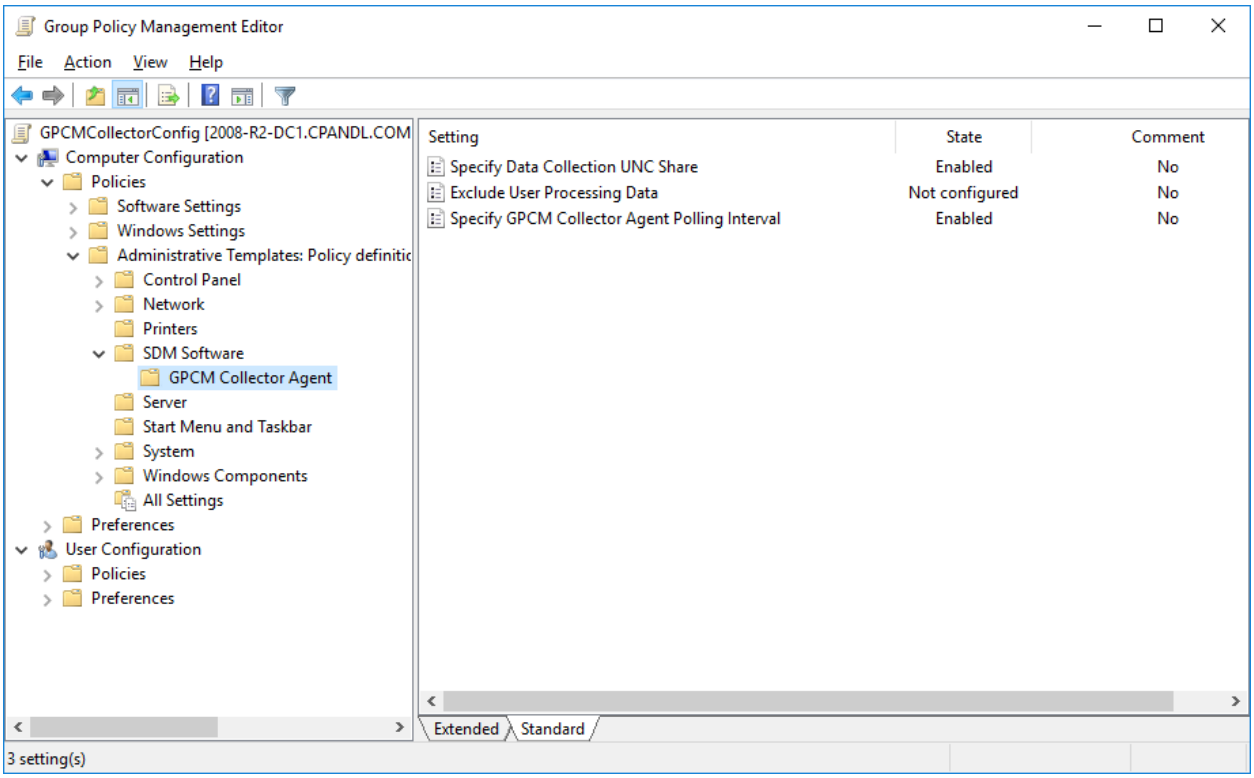

Figure 3 GPCM Collector Service ADMX Options

For more information on configuring the GPCM Collector Agent and associated ADMX files, see the section below entitled, “Configuring the GPCM Collector Agent”.

GPCM Deployment Scenarios

GPCM 1.6 can be deployed in a variety of configurations, depending upon the size of your environment, layout of the network and Active Directory domain boundaries. The basic relationship between the key components of GPCM are displayed in Figure 4 below:

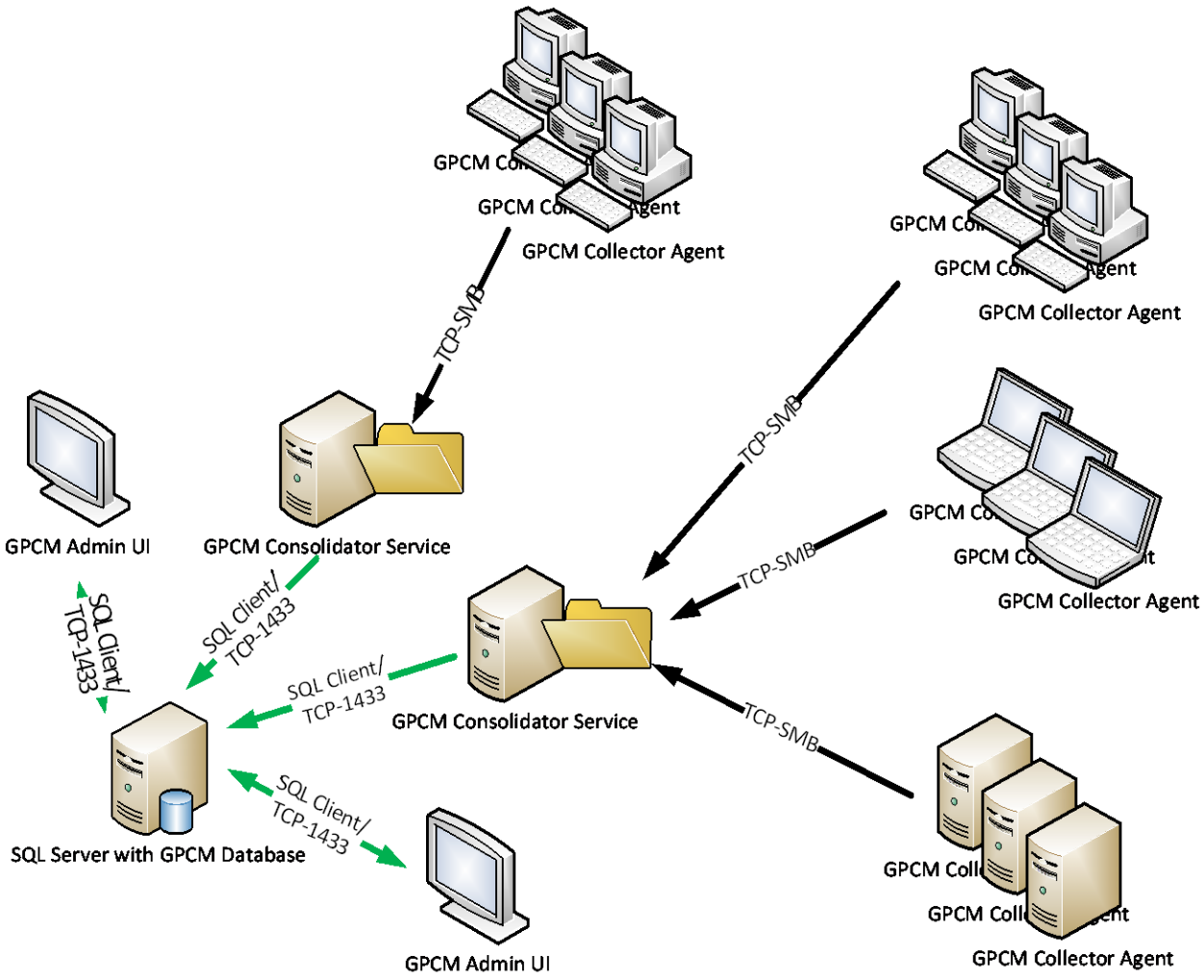

Figure 4 A typical large-scale GPCM deployment.

You have three main options when deploying GPCM. These are based largely on the size of your environment. The options are as follows:

Deploying GPCM In Small Environments

For environments collecting against approximately **100 endpoints or fewer**, where only a **single administrator** need collect and access GPCM data, the GPCM Admin UI installation with embedded SQL LocalDB database is sufficient and will meet most needs. Again, the database resides locally with the GPCM Admin UI and thus is only able to be viewed from that GPCM Admin UI installation. In this scenario, GPCM Collector Agents or Consolidator servers are not available for deployment—**all collection is done manually through the Admin UI.**

Deploying GPCM in Larger Environments

For environments collecting against **more than 100 endpoints**, we recommend the use of a **SQL Server** database along with the GPCM Collector Agents and GPCM Consolidator service. The deployment of these latter two components ultimately depends upon the size and makeup of the environment being collected against. Note that in this deployment scenario, multiple GPCM Admin UI and/or GPCM PowerShell Module installations can view/query and collect into the GPCM database. This last point is

important. Even though the GPCM Collector Agent has been deployed to an endpoint for scheduled collection, you can still perform ad-hoc collections of one or more endpoints from the GPCM Admin UI into the SQL Server repository. See the sections below on “Configuration the GPCM Collector Agent” and “Configuring the GPCM Consolidator Service” for more information on deploying these components.

Hybrid Deployment—SQL Server with No Agents

A hybrid deployment is one where you deploy the GPCM Admin UI but also deploy a SQL Server-based repository, but don’t rely on the GPCM Collector Agents and Consolidator service to collect data. This deployment model can be useful for larger environments that require the scalability of SQL Server as a repository, but don’t want to deploy agents to collected endpoints. In this scenario, all collection is done to the SQL Server repository, using the pull-based mechanism available through the GPCM Admin UI.

Deploying the GPCM Admin UI, PowerShell cmdlets and SQL Server Repository

The GPCM Admin UI, when used in conjunction with a SQL Server deployment, can be installed on multiple Windows server or desktop machines and can be used to view, report on and collect GPCM settings data. The GPCM Admin UI is also the interface you will use to create the initial SQL Script that can be imported into **SQL Server Management Studio** to create the GPCM database.

Both the initial creation of the SQL script (a one-time task) as well as configuring a connection between the GPCM Admin UI and the installed SQL Server database, can be done from the File, Settings menu in the GPCM Admin UI, as shown in Figure 5 below

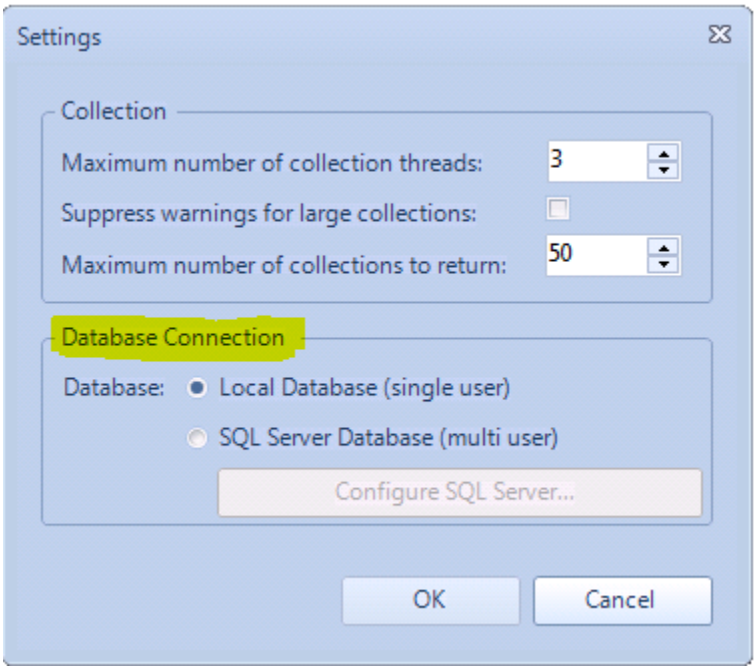

Figure 5 The Database configuration dialog

You can set the current mode of the GPCM Admin UI by choosing either “Local Database” to use the embedded SQL LocalDB instance, or by choosing “SQL Server Database” to either connect to an existing SQL Server instance, or generate a SQL Script to create a new instance. To configure the SQL Server connection, press the “Configure SQL Server” button and then the SQL Server connection dialog will appear:

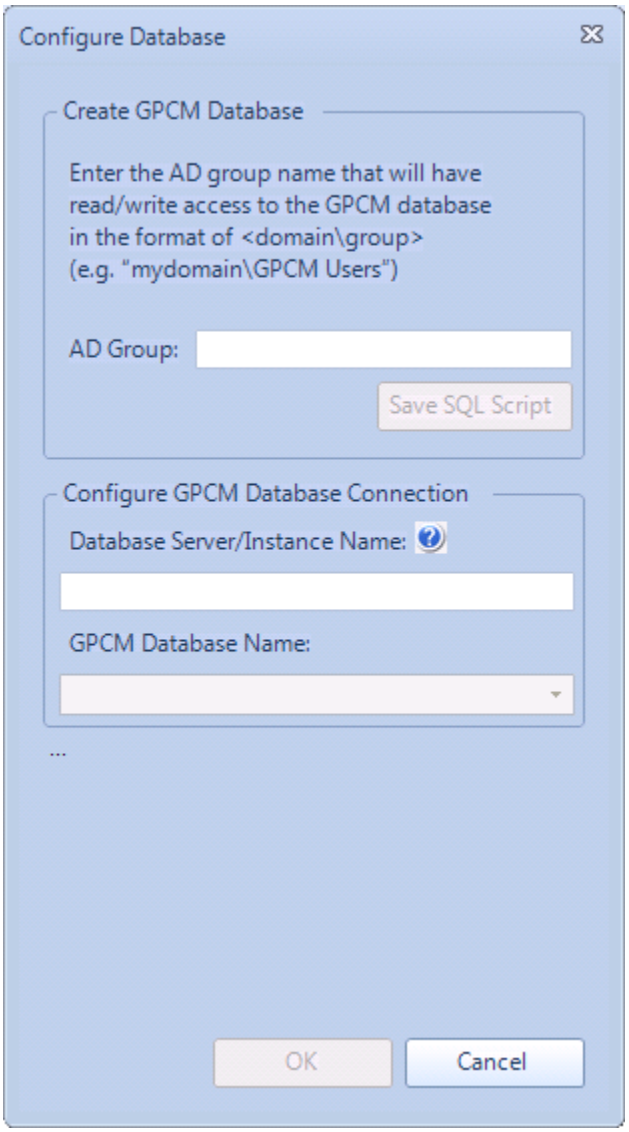

Figure 6 Configuring the SQL Server Connection

Creating and Selecting the SQL Server Database

If you don’t yet have a GPCM Database built in SQL Server, the upper section of the dialog, entitled “Create GPCM Database”, is where you’ll start. The first pre-requisite here is to create an **Active Directory Global Security Group**. This group should contain administrative users who will be using the GPCM Admin UI or GPCM PowerShell cmdlets, or service accounts that might be used by the **GPCM Consolidator Service**. Since this group will grant both read and write access to the GPCM SQL Server instance, members of this group can both read/query GPCM collection data, as well as write new

collections to the database (either via the Consolidator or by user-driven interactive collections from the GPCM Admin UI). Enter the name of the AD Group in the format of <domain\group name> (e.g. for the cpandl.com domain, the group name entered would be ***cpandl\GPCM Admins***). Once you enter a group name, the "Save SQL Script" button will become active. Pressing that button will open a file dialog where you save the resulting SQL script for use on your SQL Server Database server.

The easiest way to create the GPCM database from the script, is to connect to your SQL Server using SQL Server Management Studio, highlight the "Databases" node in the left-hand treeview, press the "New Query" button on the menu bar, and paste in the contents of the *.sql file you created above, as shown in Figure 7:

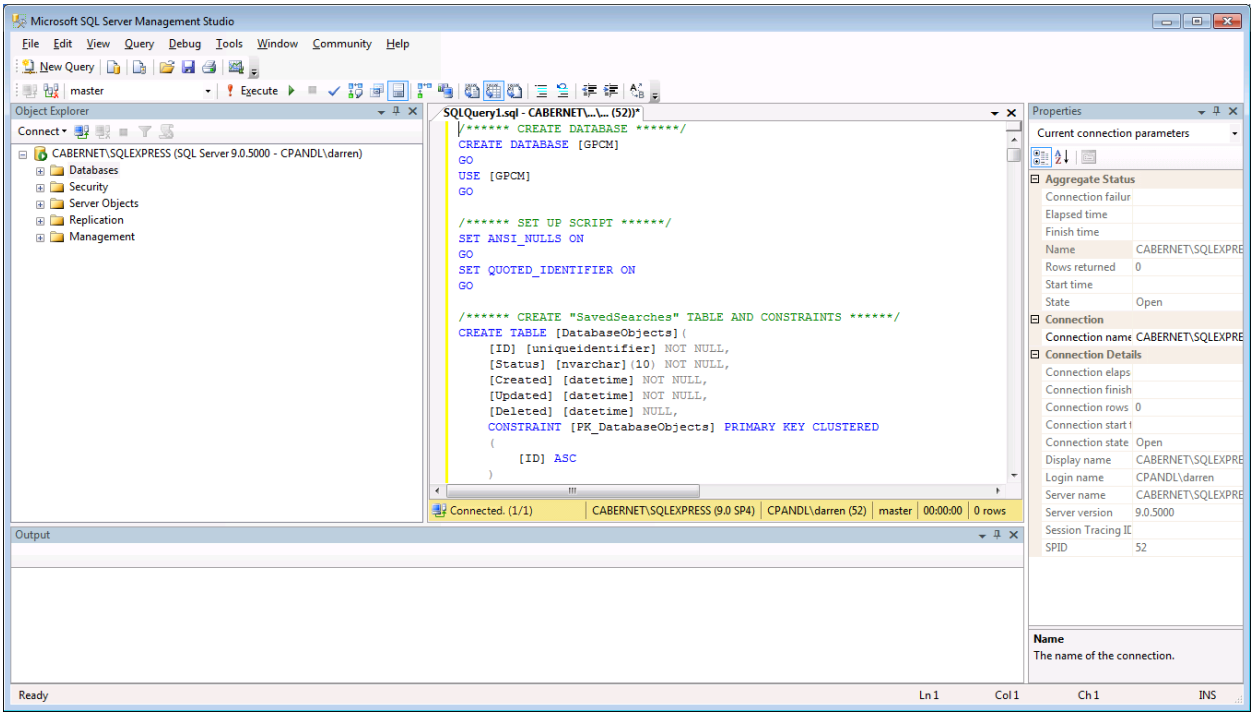

Figure 7 Creating the GPCM Database in SQL Server Management Studio

Once script is pasted in, run the script by pressing the Execute button on the toolbar. You should see the script complete with no errors and the GPCM database now appear under the Databases node in the left-hand tree-view, as shown here:

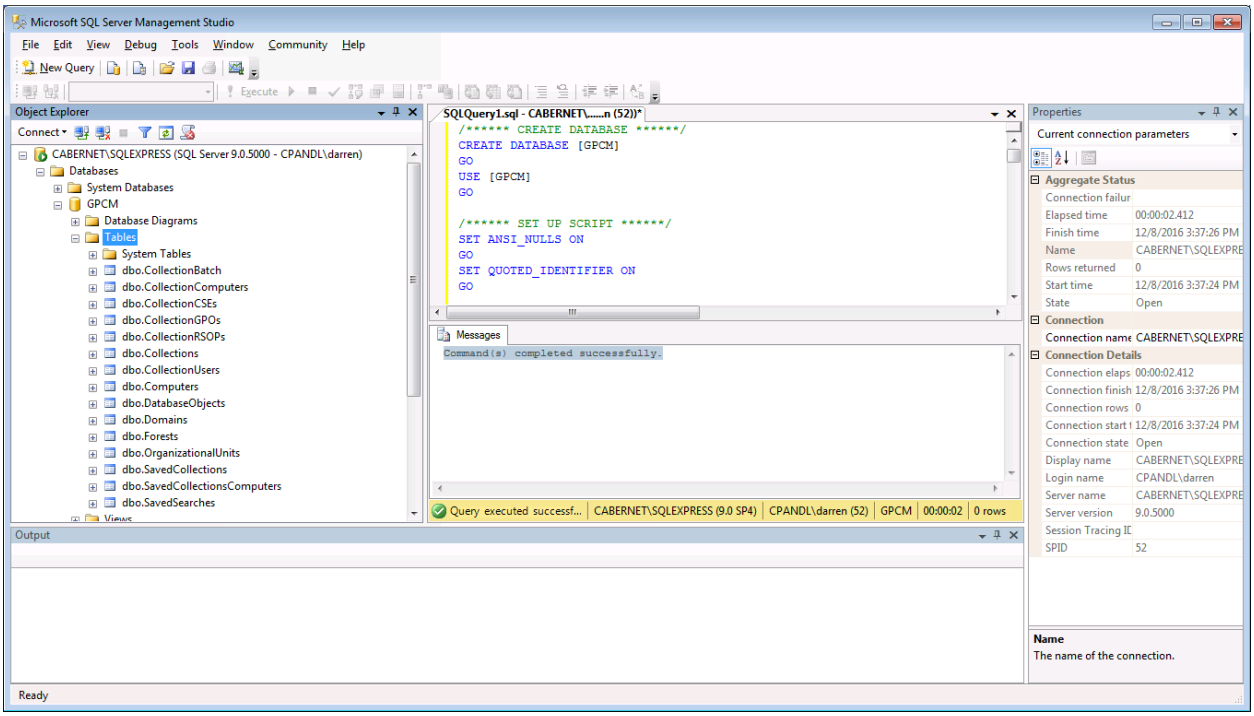

Figure 8 A successful GPCM database creation

Once the GPCM database exists, you can go back into the GPCM Admin UI Database connection dialog shown in Figure 6 above, and type the database server and, optionally the instance name, to connect to from the GPCM Admin UI. When the server name is entered, dropdown the GPCM Database Name dialog to find the GPCM database. It should appear in the list of available databases on that server. If it does not, an error message will appear in black text below the drop down indicating the status of the connection, and any errors found, as shown here:

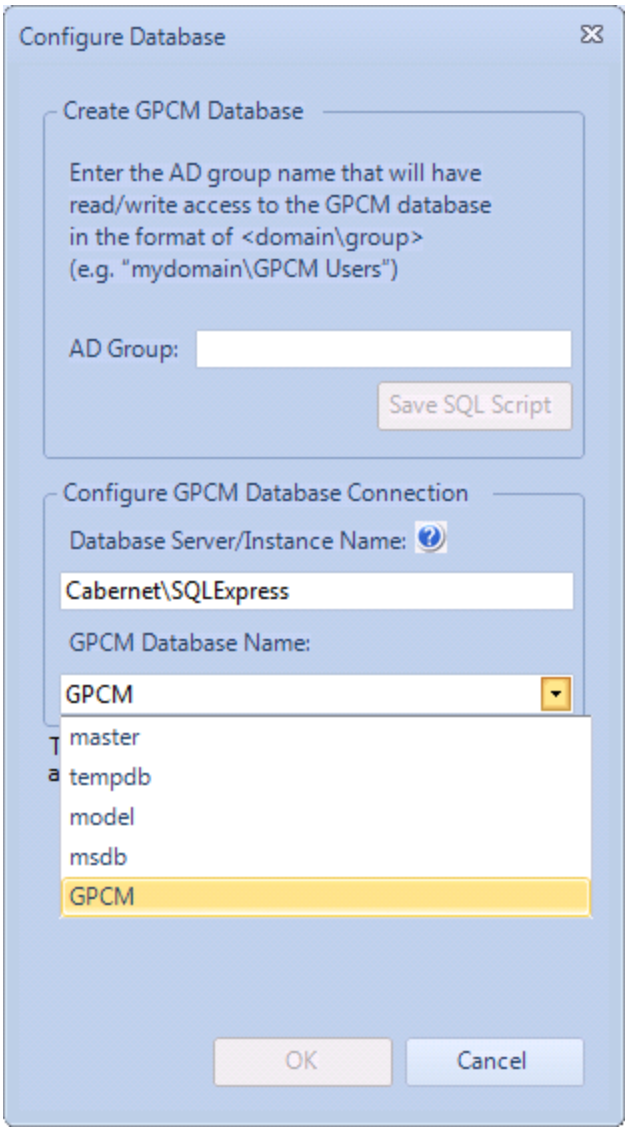

Figure 9 Successfully connecting to the GPCM database

Make sure you are running the GPCM Admin UI as a user who is a member of the GPCM AD group defined above—if you are not in that group, you may not be able to see the GPCM database and therefore select it for use in the UI.

Deploying the GPCM PowerShell Module

The GPCM PowerShell module provides a set of cmdlets for querying the GPCM database. The module supports either the local SQL LocalDB instance or a SQL Server-based GPCM database. In the case of the local database, it is best to install the module on the same machine where the local database is installed. By default, the local database is used by the cmdlets to perform operations. If you are using a SQL Server repository for GPCM, then the PowerShell module can be installed on any machine. When you call a cmdlet, you will use the /UseSQL parameter and supply the server and optionally, the instance name to tell the cmdlet to operate against the GPCM database table on that server. For example:

```
GET-SDMUSER -USESQL SQLSERVER1\INSTANCE1
```

Connects to a named instance on SQLServer1 to run the query.

If you don't provide the -UseSQL parameter, then the cmdlets operate the local SQL LocalDB instance in the default location.

Deploying and Configuring the GPCM Collector Agent

The GPCM Collector Agent is a 64-bit Windows Service that you can install on Windows Server or desktop OS'. It supports installation on Windows 7, 8.1, 10, 2008-R2, 2012, 2012-R2 and 2016 and requires .Net Framework (the full framework) 4.5.2 be installed on target systems.

The service runs using the "**LocalSystem**" account. Thus, no domain service account is needed for this service. This account has full access to the machine and runs as the domain computer account when accessing the network. The installer for the service is an MSI file (**GPCMCollector1.6Setup.msi**). The MSI can be run interactively or it can be installed using any standard enterprise deployment tool, including Group Policy Software Installation (GPSI). To use GPSI, you will want to first make an administrative install of the GPCM Collector Agent setup. You can do this by issuing the following command:

```
MSIEXEC /A C:\PACKAGE\GPCMCOLLECTOR1.6SETUP.MSI
```

This will install the installation files under c:\program files\sdm software\GPCMCollectionService on the system where the command is run. Copy all the files in that folder to a server share that grants read access to computers in your domain. You can then create a GPSI package that points to the MSI file on that share, as a Computer Assigned package, as shown here:

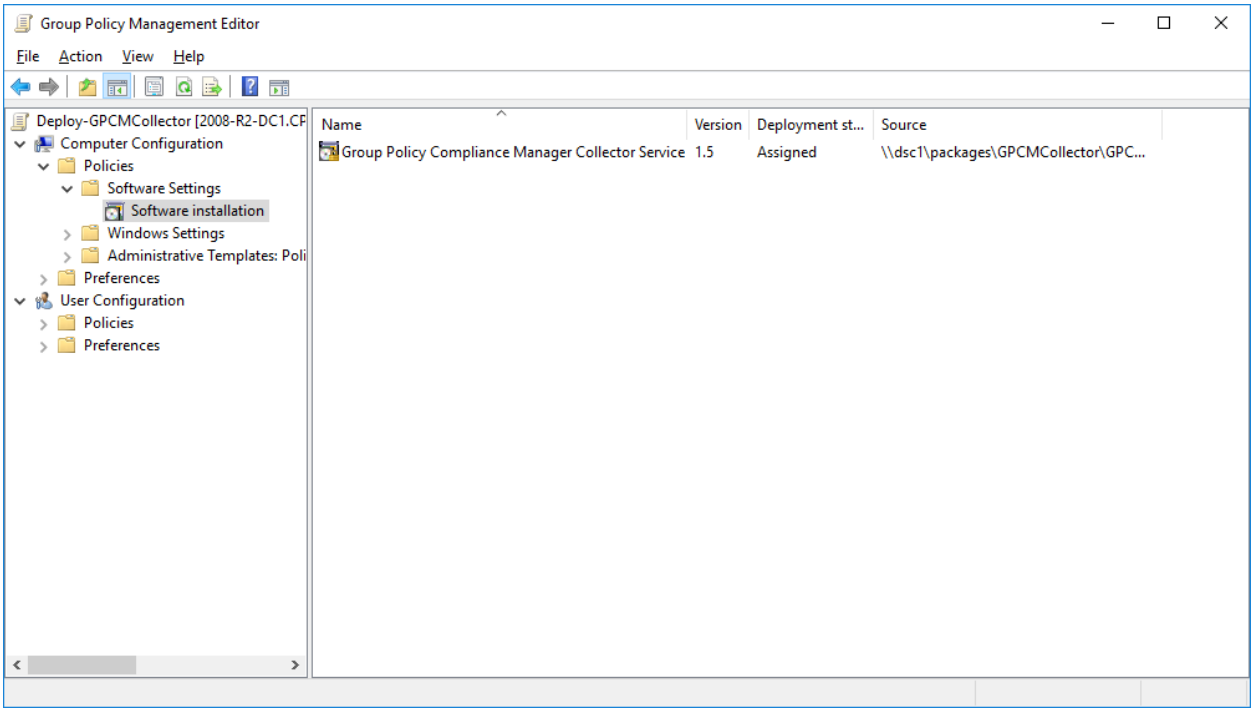

Figure 10 Deploying the GPCM Collector Agent via Group Policy Software Installation

Configuring the GPCM Collector Agent

The agent itself has no configuration utilities built into it. Configuration of the GPCM Collector Agent is managed using the provided ADMX Administrative Template files, whose settings can be deployed via Group Policy. As shown in Figure 3 earlier in this document, the ADMX file provides 3 options to configure with respect to the GPCM Collector Agent, as follows:

- **Specify Data Collection UNC Share**: This policy holds the share path that the GPCM Collector Agent will send it's locally collected files to (e.g. \\server\GPCMData). Note that the computer account will need read and write access to the share and its underlying folder to be able to copy data up to it.
- **Exclude User Processing Data**: When this policy is enabled, the GPCM Collector Agent only collects and report per-Computer GP processing data. The default (Not Configured) state automatically collects any GP processing from the currently logged on user (logged on locally or via Remote Desktop).
- **Specify GPCM Collector Agent Polling Interval**: This setting controls how often the GPCM Collector Agent collects and reports up GP processing data. The default (Not Configured) value is 1440 minutes, or one day.

  NOTE: In large environments, it's important that this value not be set too low to avoid growing the database rapidly . **Since GP processing rarely changes on a given set of systems, a one-daily collection** (or even less frequent) **is usually sufficient, but it can be tuned up or down for some machines using Group Policy-based targeting of this policy to one OU or other another. Note that to get immediate information about the state of Group Policy on a given machine, the "pull method" should be used from the GPCM Admin UI.**

Creating the SMB Share for Collector Agent Reporting

Note that file share that you create for reporting up GPCM Collector Agent data should grant computer accounts in your domain with read and write access, as shown here:

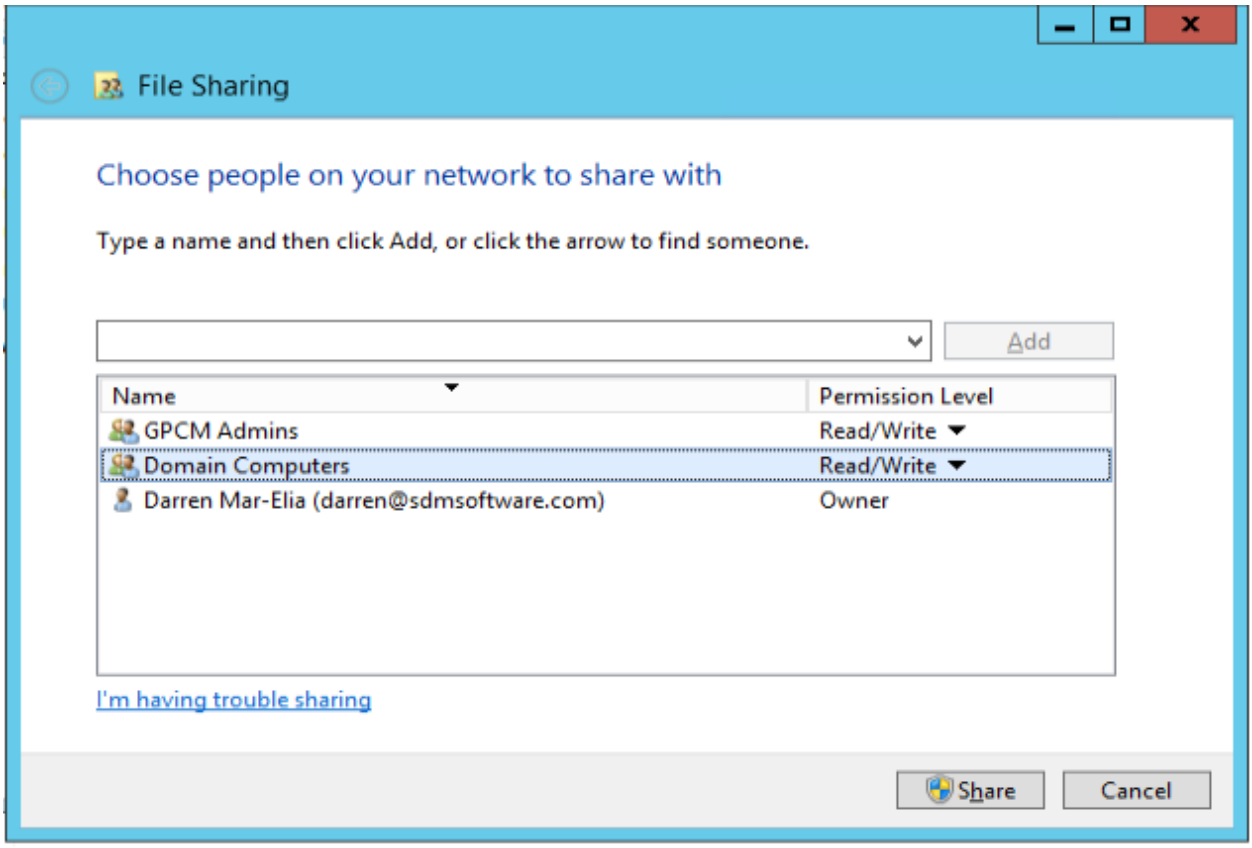

Figure 11 Granting access to the GPCM data share

Note that you can use the Domain Computers group to include access for all computer accounts in a domain, or you can specify a computer security group to ensure that only specific computers can report up their data. Also, note that in the figure above, there is a "GPCM Admins" group that has read-write access to the share as well. A member of that group is the service account being used for the GPCM Consolidator Service, described below.

The SMB Share will collect XML files from individual GPCM Collector Agents. Each file is named with the hostname of the machine that delivered it, and a date/timestamp, as shown here:

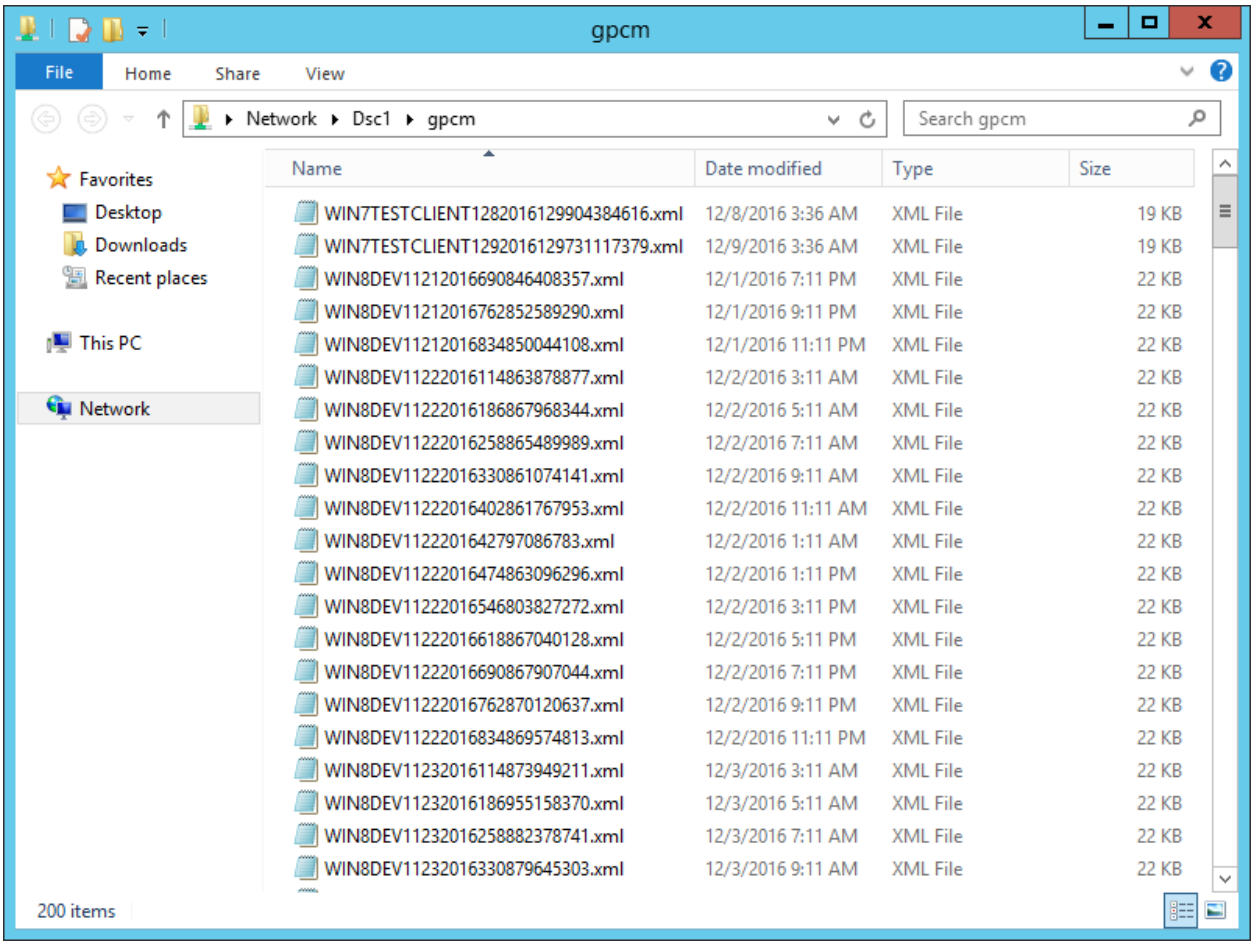

Figure 12 Viewing data collected to the SMB Share from GPCM Collector Agents

Note that these files are automatically removed by the Consolidator service as they are processed.

Deploying and Configuring the GPCM Consolidator Service

The GPCM Consolidator is responsible for taking XML files delivered to the SMB Share on a given File Server, and injecting that data into the GPCM database. In the optimal scenario, you should install the Consolidator service on the same file server where the SMB Share is collecting GPCM data. If that is not possible, then the Consolidator Service should be in close network proximity to that SMB server. You can deploy multiple GPCM Consolidator Service installations within your network, but there should always be a **one-to-one mapping between a SMB Share that is collecting GPCM data from the GPCM Collector Agent, and a given Consolidator Service installation**. That is, you should not deploy multiple Consolidator installations, pointing at the same SMB share. In any case,

GPCM Consolidator Service Account

When you run the MSI file that installs the GPCM Consolidator Service, you will be prompted to enter a service account and credentials during the install (note that the dialog appears in the upper right of your screen and could be missed) as shown here:

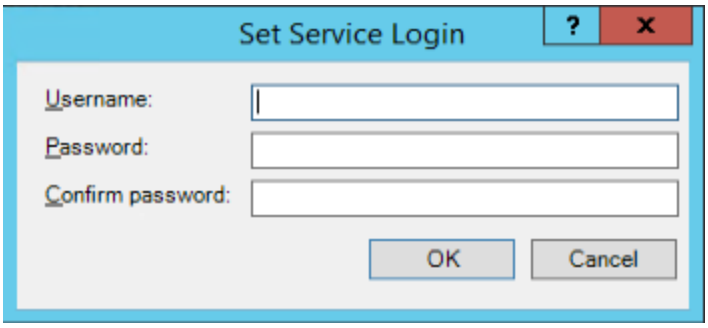

Figure 13 Prompting for the GPCM Consolidator Service Account

Permissions Required by Service Account

Provide a username in the form of <domain\samAccountName> (e.g. cpandl\svc_GPCMConsolidator). If that user account does not have the "**Logon As a Service**" user right on that server, then you will need to add that to the server prior to starting the service. In addition, the service account should have local administrator access on the server. This is required at least during the first startup of the server in order to add the appropriate event log entry related to the Consolidator service (the Consolidator service will send error events to the application event log if it encounters problems during operation). The GPCM Consolidator service account also needs read/write access to the GPCM SQL Server database, either via putting the account in the AD group you created when creating the database installation script, or by directly granting the account db_reader and db_writer within SQL Management Studio. Finally, the service account needs **read/write** access to the SMB share that holds the data that is being collected and injected into the database. Write access is needed because after a given XML file is injected into the database, it needs to be deleted from the share.

Configuring the Consolidator Service

After installing the Consolidator Service, you will see a "Configure GPCM Consolidator Service" shortcut appear on the Start Menu in the SDM Software program group. This configuration utility must be used to configure the Consolidator Service prior to starting it, and presents the following options:

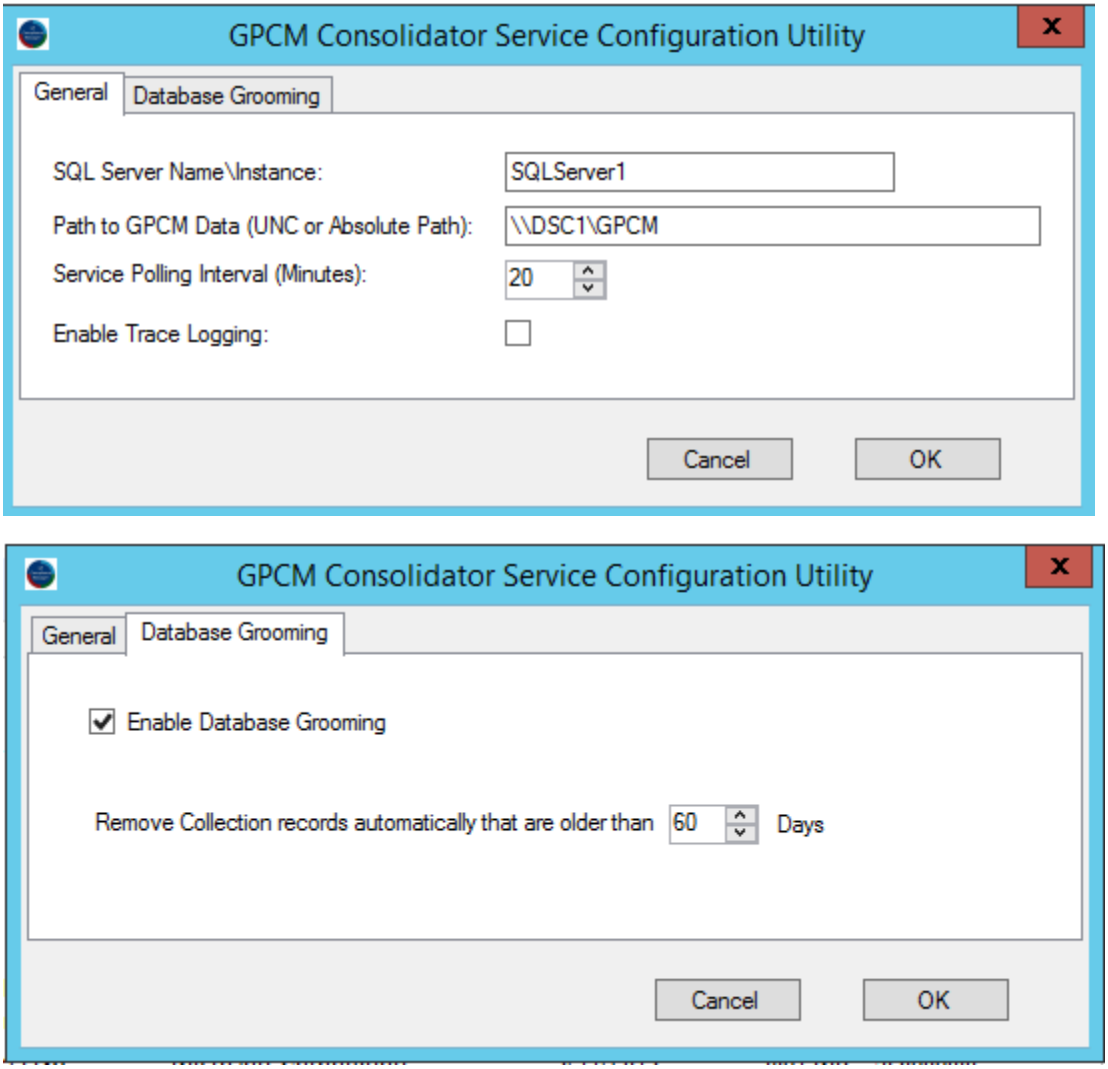

Figure 14 The configuration utility for the GPCM Consolidator

The configuration utility provides five options to configure. The first three must be configured prior to starting the service. The options are described here:

- **SQL Server Name\Instance**: This where you provide the name of the database server and instance running the GPCM Database. The configuration utility assumes that the database name itself is the default, which is “GPCM”. If you are using the default instance on SQL Server, then just enter the server name here.
- **Path to GPCM Data (UNC or Absolute Path)**: This is where you tell the Consolidator service where to find the GPCM XML files that have been collected from Collector Agents. If you are running the Consolidator on the same file server where the files are stored, you can enter an absolute path to the files (e.g. c:\data\gpcm) but if you are collecting the files remotely, then enter the UNC path to the share.
- **Service Polling Interval (Minutes)**: This controls how often the Consolidator wakes up to gather files to inject into the database. The default is every 20 minutes. In large environments, where many files are being put in the share during a given interval, you might need to adjust this

number upwards. The goal is for the Consolidator service to complete injecting all outstanding files from the share before the polling interval is up. 20 minutes is a good starting point and the rate of injection will vary based on network and database performance characteristics.

- **Enable Trace Logging**: by checking this box and restarting the Consolidator service, a trace log will be generated during Consolidator operation. The trace log can be found under c:\windows\temp\gpcmingestor.log and can be provided to SDM Software Support if it's needed.
- **Enable Database Grooming**: If you enable the database grooming feature, then you can configure the age of GPCM collections after which the database will automatically be deleted. The grooming feature is performed by the GPCM Consolidator Service and defaults to grooming collections and their related data that are greater than 60 days old, when enabled. You must check the "Enable Database Grooming" option on the Database grooming tab in order to implement grooming.

Once the service is configured using this utility, you can start the service and after the polling interval elapses, consolidation will occur and you will begin to see GPCM data in the Admin UI. You can identify data that was collected in this way vs. manual collection in the Admin UI by looking at the collection duration column

Summary

The deployment of GPCM and its components depends upon a variety of factors, including how many endpoints are reporting to an SMB Share, how many Consolidator services are running and how often each polls for data. Contact support@sdmsoftware.com for help in determining the optimal deployment strategy for your scenario.