Automating the setup of Deployment Manager Agents

When installing large numbers of agents, or using virtual machines with a temporary lifespan, it can be helpful to be able to automate the creation of Deployment Manager agents. Here is one way to do so:

Install an agent manually to serve as a template

First of all, install an agent using the normal manual install process. This will ensure that you know which firewall settings you need to tweak on the target machines, and generate an X509 encryption key which can then be shared among the agents you want to automate.

Note: using this method will allow the agents to impersonate each other if one is compromised. Think carefully before using this technique to share security settings between agents with different levels of security

Run the installer silently on the target machine

Using Microsoft's Group Policy, or your organization's preferred equivalent, execute the agent installer silently on the target machine. If using a script to execute the installer, remember to set the /quiet flag for a silent install.

Update the registry keys on the new agent to the same values as the template machine

Inside "HKEY_LOCAL_MACHINE\SOFTWARE\Red Gate\Deployment Manager\", set the registry keys "Agent.Security.TrustedDmThumbprints" and "Cert-cn=Red Gate Deployment Agent" to be the values of the equivalent keys on your template machine. This will instruct the agent to use the same x509 encryption certificate to encrypt its messages and identify itself to the server. It will also instruct it to accept incoming instructions from the same server.

If you choose to roll out these registry changes using Microsoft's Group Policy, you may find the following template for a custom policy useful:

```
CLASS MACHINE
CATEGORY "Deployment Manager Agent security"
  POLICY "Deployment Manager Security"
    EXPLAIN DmAgentHelp
    KEYNAME "SOFTWARE\Red Gate\Deployment Manager"

    PART server_thumbprint EDITTEXT
      VALUENAME "Agent.Security.TrustedDmThumbprints"
    END PART

    PART agent_certificate EDITTEXT MAXLEN 8192
      VALUENAME "Cert-cn=Red Gate Deployment Agent"
    END PART
  END POLICY
END CATEGORY
[strings]
DmAgentHelp="This configures the Deployment Manager Agent security settings; the server it will accept
instructions from and the key it will use to identify itself"
```

This article about setting up a custom policy in Group Policy may be of help: see part 2: "the hard way" for advice on using the above template.

Register the new Agent with the Deployment Manager

All that remains now is to tell the server where it can find the new machine. You can do this by making a POST request to:

http://<Deployment Manager server address and port>/api/environments/<EnvironmentId>/machines/add

- EnvironmentId is the ID of the environment to add the machine to. You can view Environment IDs on *http://<Deployment Manager server address and port>/api/environments*
- Set the header, *X-RedGateDeploymentManager-ApiKey*, in the POST request to the API key of the Deployment Manager user. To find your API key, see Finding your API key.

Set the following URL parameters:

Parameter Name	Example	Description
Name	*test-agent-01*	The user-facing name for the agent in the Deployment Manager system.
AgentHostName	*agent-01*	The host name the server should use to communicate with the agent.

AgentPort	*10301*	Unless you've specifically set this differently, it should be *10301.*
Thumbprint	*CF8676EE5BEB8AD8864A7AD5D55CBA9B97E86B3E*	This should be the value of the thumbprint you used to set up your initial template machine.
TargetType	*Machine*	The type of target to create. Allowed values: ○ *Machine* - a general target machine ○ *SqlServerInstance* - a SQL Server

- If you're creating a SQL Server target, the following URL parameters are also required:

Parameter Name	Example	Description
ServerName	*my-machine\sql2008r2*	The fully qualified address of the SQL Server instance to add.
AuthMode	*Sql*	The type of authentication to use when connecting to the SQL Server instance. Allowed values: ■ *Sql* - use SQL Server authentication (recommended) ■ *Integrated* - use Windows authentication
Login	*bob*	SQL Server user with sysadmin permissions. Only required when using SQL authentication.
Password	*password1234*	SQL Server password. Only required when using SQL authentication.

Example POST request

http://localhost:8050/api/environments/DeploymentEnvironments-1/machines/add?Name=MyMachine&AgentHostName=localhost&AgentPort=10301&Thumbprint=3578820162E0121A6B41F67D9B7FE7BD58E9243F&TargetType=Machine

Using PowerShell to register a new target machine with the Deployment Manager Server

This PowerShell script adds a new target machine to Deployment Manager:

```
# Input Variables
$DmServer = "localhost"
$DmPort = "8050"
$DmApiKey = "5K1UMAUGDZWKOHWD1TFMRRHK"

$EnvironmentId = "DeploymentEnvironments-1"
$Name = "dm-agent-01"
$AgentHostName = "localhost"
$AgentPort = "10301"
$Thumbprint = "3578820162E0121A6B41F67D9B7FE7BD58E9243F"
$TargetType = "Machine"
# ----------------------------

$RequestURL = "http://${DmServer}:${DmPort}/api/environments/${EnvironmentId}/machines/add"
$PostData = "Name=${Name}&AgentHostName=${AgentHostName}&AgentPort=${AgentPort}&Thumbprint=${Thumbprint}
&TargetType=${TargetType}"
$UrlWithFormData = "${RequestURL}?${PostData}"

Write-Host "Sending request to url: $UrlWithFormData"
$response = Invoke-WebRequest -Uri $UrlWithFormData -Method Post -Headers @{"X-RedGateDeploymentManager-ApiKey"
="${DmApiKey}"}
Write-Host $response
```