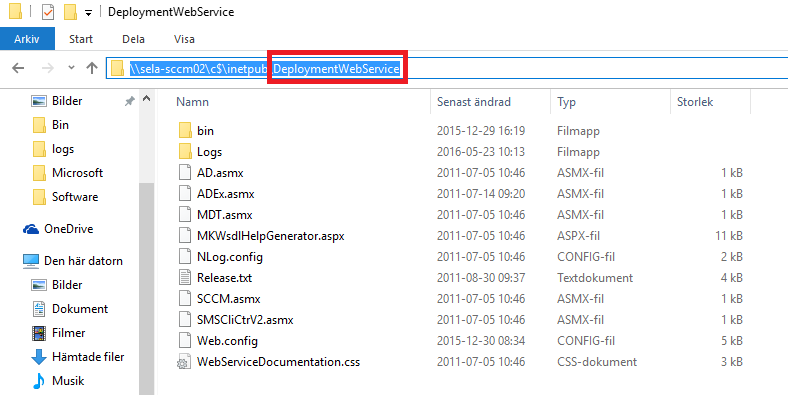
Web Service

A web service can be used as a proxy to configure or fetch information from AD / SCCM. <http://mdtcustomizations.codeplex.com/documentation?referringTitle=Home>

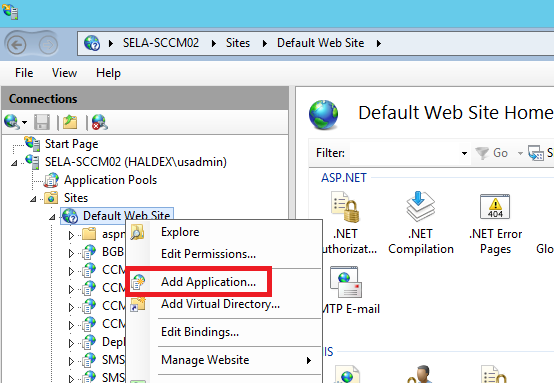


Installation:

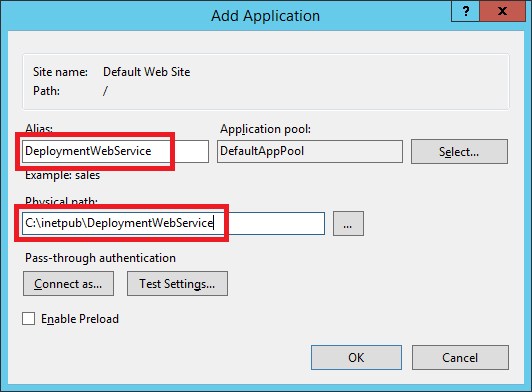
1. Create a folder under inetpub and copy the content of Deployment webservice files into this.



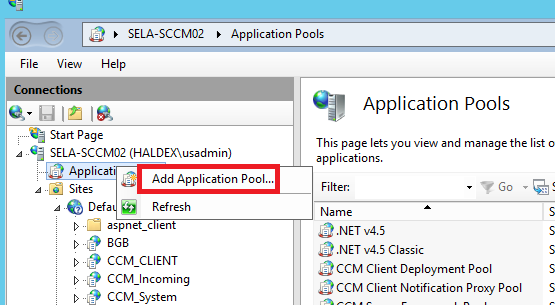
1. In IIS management console create a new “Application” under “Default Web Site”.



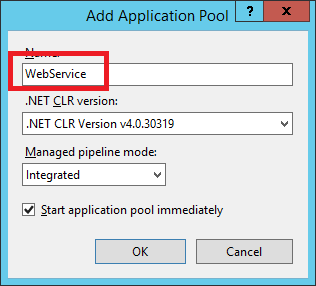
1. Specify an Alias name and point physical path to the folder where the files from 1. Was placed.



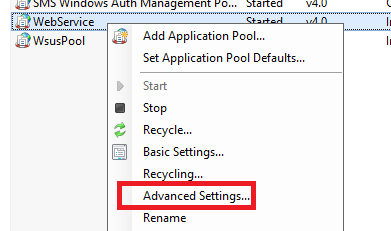
1. In IIS manager add an Application Pool.



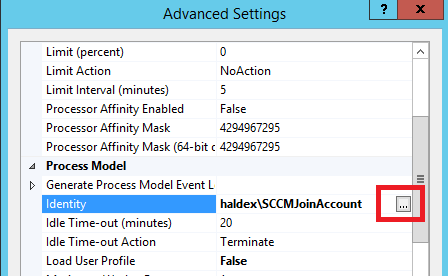
1. Give the Application pool a name.



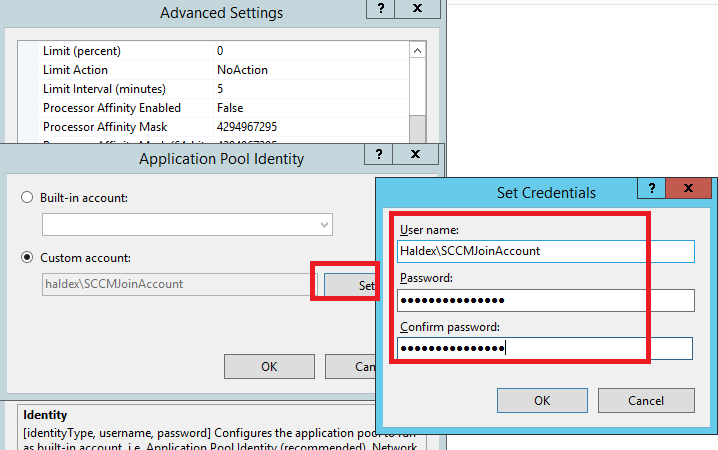
1. Configure the new application pool in using advanced settings.



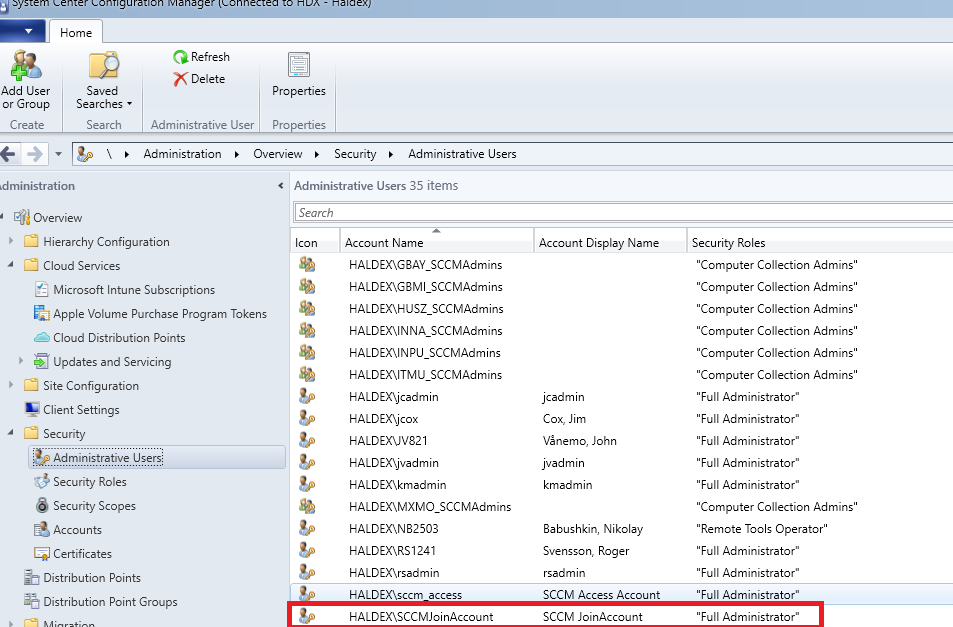
1. Add a service account that must have sufficient permissions in SCCM and AD.

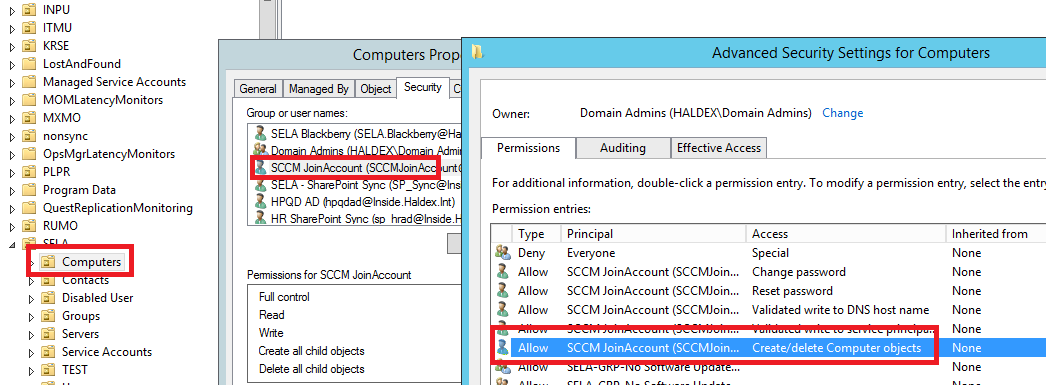


1. Enter account information.



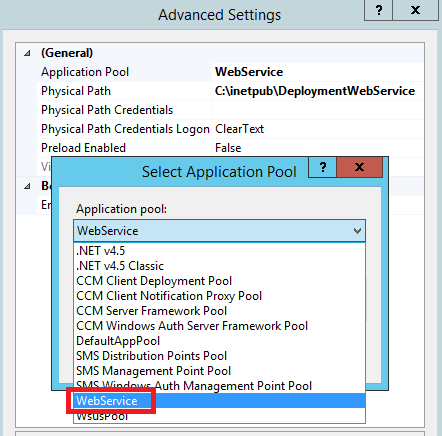
For Haldex SCCMJoinAccount has admin permissions in SCCM database and on OU:s containing computer objects:





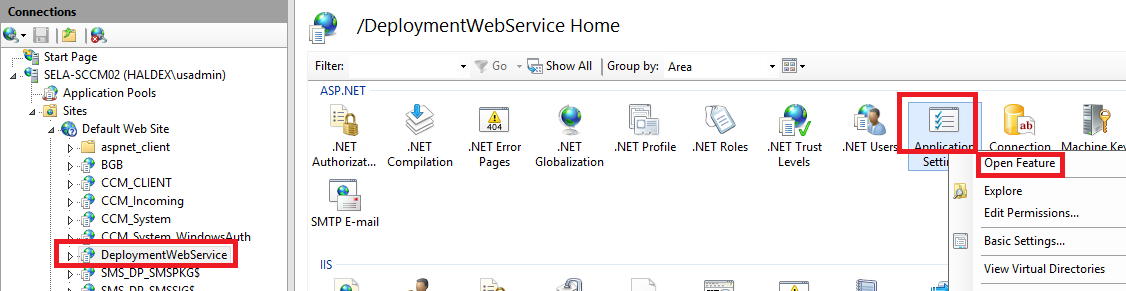
The password is available in the handover Excel sheet.

1. Configure Web Service Application to use the new Application Pool. Right click on Application pool and enter advanced configuration.

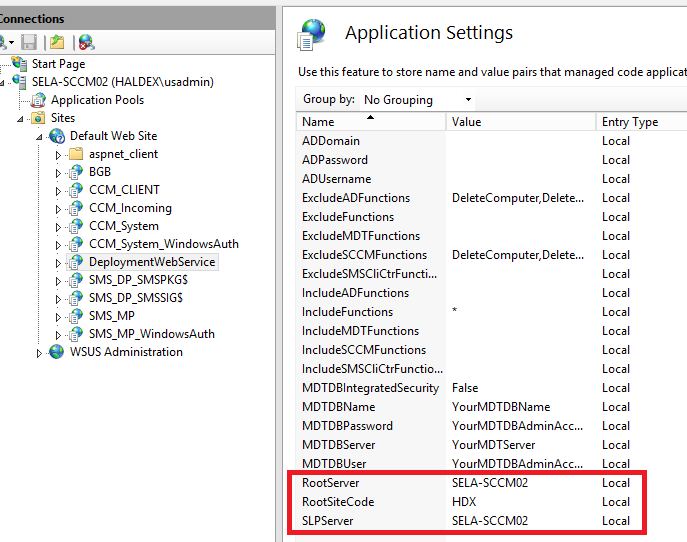


1. Configure default connection such as RootServer (The SCCM server), SLPServer (One SCCM server with SLP role) and RootSiteCode (The SCCM sitecode).

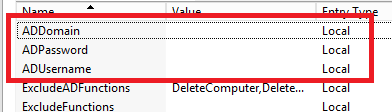
In IIS manager click the Webservice and right click on “Application Settings”:



Configure following according to your domain:



If you want to use a different account connecting to Active directory you might configure:



If left blank, the account configured in Application Pool is used.

Now the Webservice should be functional and you can access this:

http://SELA-SCCM02/DeploymentWebService/ad.asmx - For Active Directory related functions

http://SELA-SCCM02/DeploymentWebService/mdt.asmx - For MDT related functions

http://SELA-SCCM02/DeploymentWebService/sccm.asmx - for SCCM related functions

Examples on methods calling webservice:

# Get Computer Name from SCCM database

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/sccm.asmx?WDSL

$ComputerNameFromSCCM = $ADWeb.GetComputerName("$MacAddress","$UUID","$SMSTSAssignedSiteCode")

# Get AD groups the computer is member of

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/ad.asmx?WDSL

$groups = $ADWeb.GetComputerGroupsByName("$ComputerName")

# Get information about if computer object exist in SCCM database

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/sccm.asmx?WDSL

$ComputerExistInSCCM = $ADWeb.IsComputerKnown("$MacAddress","$UUID","$SMSTSAssignedSiteCode")

# Get information about if computer object exist in AD

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/ad.asmx?WDSL

$ComputerExistInAD = $ADWeb.DoesComputerExist("$ComputerName")

# Get LDAP Path in AD

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/ad.asmx?WDSL

$OUPath = $ADWeb.GetComputerParentPath("$ComputerName")

# Get OS version attribute from AD object

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/ad.asmx?WDSL

$AD\_Attribute = "OperatingSystem"

$OSVersionFromAD = $ADWeb.GetComputerAttribute("$ComputerName", "$AD\_Attribute")

# find and delete the computer from AD

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/ad.asmx?WDSL

$result = $ADWeb.DeleteComputer("$ComputerName")

# Remove computer object from SCCM database

$ADWeb = **New-WebServiceProxy** -Uri http://sela-sccm02/deploymentwebservice/sccm.asmx?WDSL

$result = $ADWeb.IsComputerKnown("$MacAddress", "", "$SMSTSAssignedSiteCode")