Ops Server Configuration Preparation

Install Software Preparation

In order to use the ops-server-config scripts to install the ArcGIS Platform software you will need to unpack and copy the individual software installers to specifically named and structured folders. Unless otherwise noted, all of the installers/executables required can be downloaded from My Esri. See the instructions below for more information:

1) Run the ops-server-config\Utilities\CreateOpsServerReleaseFolderStructure.bat file to create the folder structure required by the ops-server-config repo installation scripts.

2) Copy the ops-server-config repo scripts and the software installers to the appropriate OPSServerInstall\Software folder created by the CreateOpsServerReleaseFolderStructure.bat file:
   a. Ops-server-config scripts
      i. Copy the files/folder in the ops-server-config repo to the OPSServerInstall\Software\ops-server-config folder.

   b. ArcGIS Data Store
      i. Double-click the ArcGIS_DataStore_Windows_103_144654.exe file to unpack the installation files.
      ii. Copy the unpacked files/folders to the OPSServerInstall\Software\ArcGISDataStore folder.

   c. ArcGIS GeoEvent
      i. Double-click the ArcGIS_GeoEvent_Extension_for_Server_103_142135.exe file to unpack the installation files.
      ii. Copy the unpacked files/folders to the OPSServerInstall\Software\ArcGISGeoEvent folder.

   d. ArcGIS Server
      i. Double-click the ArcGIS _for_Server_Windows_103_142101.exe file to unpack the installation files.
      ii. Copy the unpacked files/folders to the OPSServerInstall\Software\ArcGISServer folder.

   e. Chat Server
      i. Download the Windows Openfire 3.9.3 installer from https://www.igniterealtime.org/projects/openfire/ and copy the .exe file to the OPSServerInstall\Software\ChatServer\Openfire folder.

f. PostgreSQL database
    i. Create a folder named "Postgre_installation" under the folder OPSServerInstall\Software\Database\ PostgreSQL_9.2.2.
    ii. Copy the PostgreSQL installer executable (i.e. postgresql-9.2.2-1-windows-x64.exe to the OPSServerInstall\Software\Database\PostgreSQL_9.2.2\Postgres_installation folder.

        NOTEs:
        - If you are using a PostgreSQL build other then 9.2.2, modify the `ops-server-config\Install\PostgreSQL\InstallPostgreSQL.bat` file to reference different folder paths and file names.
        - The `InstallPostgreSQL.bat` replaces the installed PostgreSQL configuration file (postgresql.conf) with a customized file to meet specific team needs and is specific for PostgreSQL 9.2.2. If you are using a different PostgreSQL build or don't need a custom configuration file, edit or comment out commands as appropriate for your installation.
        - If you are installing a build of PostgreSQL 9.2.x, you will need to modify the major/minor build number set by the `ops_postgresqlInstallDIR` variable in the `InstallOpsServer.bat` file.

g. Message Simulator
    i. Download the ArcGIS Qt Message Simulator (10.2.2.1) from the [ArcGIS for the Military download page](#).
    ii. Extract the .exe and dlls from the AFM-ArcGISQtMessageSimulator_10.2.2.1.zip file and place in the OPSServerInstall\Software\MessageSimulator\MessageSimulator folder.

h. Operations Dashboard Deployment Utility
    i. Download the deployment utility.
        1. Please see the following link for more information on the download: [http://doc.arcgis.com/en/operations-dashboard/windows-desktop/author/portal-deploy.htm#ESRI_SECTION1_7B50BB5EFFA14FAC964A4E9D5CC35FA8](#)

            NOTE: you don't need to perform the "Create the Operations Dashboard files" and "Copy the files to your portal" steps listed in this link, as the ops-server-config scripts will perform these tasks, however, you will need to perform the step "Create an item on the portal" after Portal for ArcGIS has been installed.

2. Copy the OperationsDashboardUtility.exe and the other downloaded files to the OPSServerInstall\Software\OpsDashboardUtility folder.

   ii. Code-signing certificate
      1. Create a code-signing certificate and copy the certificate to the "Certificate" folder.
      2. Edit the `OPSServerInstall\Software\ops-server-config\Install\OpsDashboardUtility\CreateOneClick Installer.bat` file; replace "TODO_CHANGE_CERT_FILENAME.pfx" with the name of your code signing certificate; replace "TODO_CHANGE_CERT_PASSWORD" with your code signing certificate password.

   i. Portal for ArcGIS
      i. Double-click the Portal_for_ArcGIS_Windows_103_144656.exe file to unpack the installation files.
      ii. Copy the unpacked files/folders to the OPSServerInstall\Software\PortalForArcGIS folder.

   j. Web Adaptor for IIS
      i. Double-click the Web_Adaptor_for_Microsoft_IIS_103_142112.exe file to unpack the installation files.
      ii. Copy the unpacked files/folders to the OPSServerInstall\Software\WebAdaptorIIS folder.

3) Modify paths in the ops-server-config repo scripts:
   a. Edit the SupportFiles\OpsServerConfig.py script:
      i. Update the PostgreSQL database names set by the `databasesToCreate` dictionary variable to match the databases you want the ops-server-config scripts to create.
      ii. Update the `installOnlyPublishingFolders` dictionary variable to match the path(s) of where your file based data is located; used to create temporary data stores registry entries during ArcGIS Server service publishing to map the data path of source service and where this data is located on the target ArcGIS Server.
      iii. Update the `installOnlyPublishingDBServers` list variable to match the names of the PostgreSQL databases servers; used to create temporary data stores registry entries during ArcGIS Serer service publishing to map the database server of the source server and the databases on the target ArcGIS Server. Used in conjunction with the `databasesToCreate` variable to create a data store registry entry for each server/database name combination.
   b. Edit the Publish\Portal\PortalContentPost.py script. Update the `source_hostname` variable to match the fully qualified server name

4) Follow the instructions in the *Ops Server Installation Guide* to use the ops-server-config repo scripts to install the software.

Content Publishing Preparation

Source ArcGIS Server configuration

The `PublishToOpsServer.py` script (see the "Publish the ArcGIS Server services" section of the *Ops Server Installation Guide* for more information) publishes ArcGIS Server services from service definition files (.sd) that were created on your source ArcGIS Server. Service definition files are created when you publish services to ArcGIS Server. By default, ArcGIS Server deletes the service definition after the service has been published. To configure your source ArcGIS Server from deleting the service definition files, following the instructions below:

1. Add a Windows registry key to prevent the service definition file from automatically being deleted after the service has been published.
    a. Log into your ArcGIS Server machine.
    b. Open the regedit (**NOTE**: you need to run regedit as the system account that is running the ArcGIS Server windows account; see the runas command line help for more information)
    c. Create the "KeepPublishedSD" REG_DWORD in Computer\HKEY_CURRENT_USER\Software\ESRI\Server10.3\ArcToolbox\Settings and set its' value to 1.
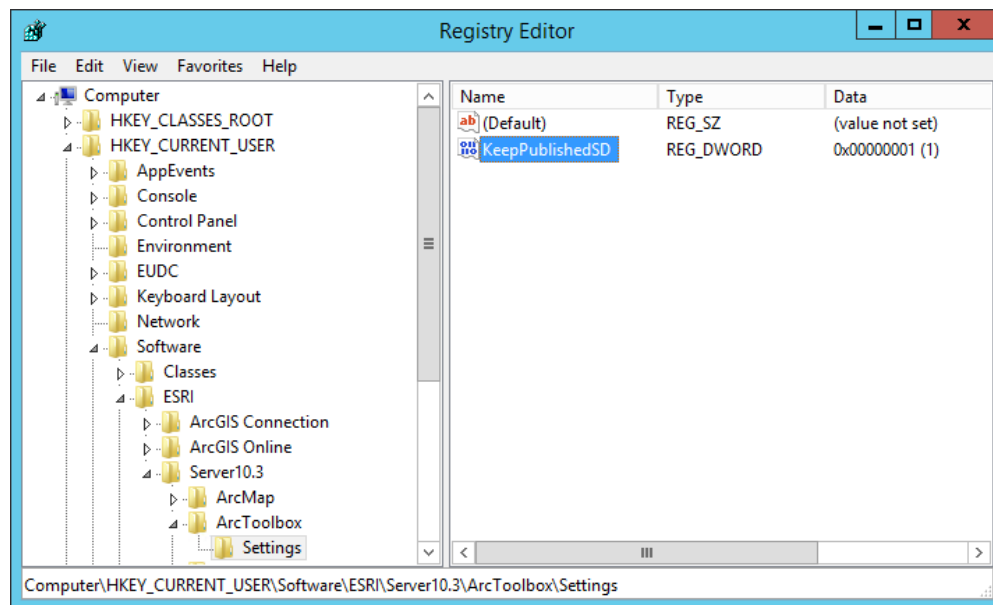
NOTE: if you have configured a multi-machine ArcGIS Server site you will need to add this registry key to each machine in the site.

2. Disable the cleanup mode of the ArcGIS Server "System" directory (i.e. arcgissystem folder) to prevent ArcGIS Server from deleting the service definition files at register time intervals.
   a. Log into ArcGIS Server Manager with an administrator account.
   b. Navigate to Site > Directories.
   c. Click the "Edit" button for the "System" directory.
   d. Change the "Cleanup mode" property to "None".
   e. Click "Save".

NOTE: to extract the service definition files from your source ArcGIS Server for use by the PublishToOpsServer.py, you must use the GetSDFiles.py script; this script extracts additional information about the associated portal items required by the PublishToOpsServer.py script to successfully publish the service.

Source Portal Content

The `PublishPortalContent.py` script (see the "Publish portal content" section of the *Ops Server Installation Guide* for more information) publishes portal content that has been extracted from your source Portal for ArcGIS site using the *PortalContentExtract.py* script.