Windocks Installation and Configuration

Windocks is a modern, open data delivery platform, based on a port of the Docker's open source that supports all editions of Windows 8, Windows 8.1, Windows 10, Windows Server 2012 and 2012r2, and Windows Server 2016.   Windocks supports .NET and all editions of SQL Server 2008 onward, Java, Nginx, node.js, and other open source projects. Inquire with support@windocks.com if you're interested to work with Java and other open source environments.

- **WinDocks Community Edition** is a free edition with the full features of WinDocks , but is limited to 2 containers with one SQL Server database per container, and non-commercial use.
- **Windocks subscriptions** start at $169 / month (up to 10 simultaneous containers), $499 / month (25 simultaneous containers) and $799/month for up to 50 simultaneous containers.   Each supports an unlimited number of cloned images.
- **Enterprise License Agreements** are also available.

Planning Windocks use with SQL Server containers

1) For databases of 1 GB or larger use database cloning.   Cloning is supported by Windows 8.1 or Windows 10 Pro or Enterprise editions, Windows Server 2012 R2, or Windows Server 2016.

2) Windocks is a lightweight process, and can be run on 2 CPU core machines with 4 GB of RAM. Servers with 4 or 8 cores, and larger RAM configurations are recommended.

3) Windocks SQL Server containers are based on a SQL Server instance installed on the Windocks host. A default or named SQL Server instance can be used, and user defined databases must be detached prior to creating containers.   Multiple SQL Server releases can be installed and configured to support multiple container images.

4) SQL Server instances are designated for use by the Windocks Service following installation (see the Configuration Section below). The SQL Server instances can be started and used for other purposes, but care should be taken to ensure the Windocks Service is turned off when using the configured SQL Server instance. Return the SQL Server to an off state, and restart the Windocks Service to resume use of Windocks.

5) Clonable images are a fully byte copy of the target environment, and are stored as a Virtual Hard Disk (VHD) in the same directory as the backup (or snapshots) used to build the image. Disk storage should be 3X or larger than the target environment, to support two VHD images and containers.

6) Storage of SQL Server containers is separate from clonable images, and modest in comparison. Each SQL container requires ~350 MB. Configuration options outlined below illustrate how containers can be stored on a separate disk volume, but this is generally not necessary when working with clone based images.

7) Windocks includes a web application that simplifies use of containers, and supports both Chrome and Firefox browsers.   IE is not supported.

Installing Windocks

- If upgrading Windocks prior to release 2.2, stop the daemon prior to the upgrade (highlight the Windows and use Cntrl C). The daemon runs as docker.exe.
- If you have multiple SQL Server named instances on the host, select the instance name used by Windocks in the node.config file located in \\windocks\config directory. Refer to the Configuration section below.
- Important, run the installer using a local **Machine Administrator** login
- The Windocks installation package will install on Windows 8, Windows 8.1, Windows 10, or Windows Server 2012, 2012r2 and Server 2016.
- Windocks includes a web application that is automatically installed in inetpub/wwwroot and will overwrite the index.html file there
- The install will take 10 to 15 minutes to complete, and requires a re-boot on completion.
- For remote client access, ensure the Windocks host is accessible to inbound traffic on the designated ports (default is 10,000 to 10,200, Docker port of 2375, and default SQL Server port 1433).
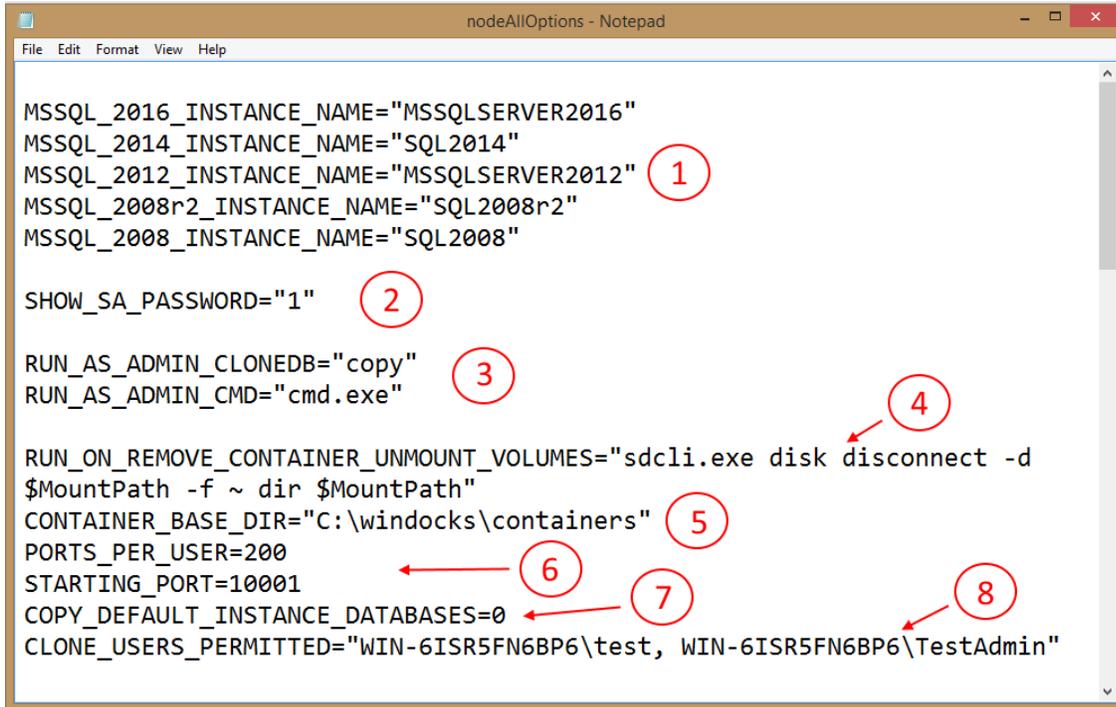
Starting Windocks

Following installation Windocks will auto-start as a Windocks Service.

Windocks monthly subscriptions includes a license key. Save the full string in a "key.txt" file and place the file in the Windocks file directory.

Configuration Options

Windocks includes configuration options in the node file located in the \windocks\config directory. The options are self-explanatory, and can be edited using notepad.

1. SQL Server instance names used by the Windocks Service are identified here.
2. SQL sa passwords can be displayed, or not, based on this setting.
3. Windocks Dockerfiles can use Windows commands as shown here.
4. Windocks Dockerfiles can work with third party executables, such as NetApp SDCLI.exe
5. Containers can be located on a separate disk drive. Be sure that the directory is setup as shown.
6. The port range for containers is defined here.
7. The ability to copy user defined databases from the default instance into containers is an option.
8. This provides file share permissions to cloned database files to the listed users.   This simplifies use of cloned database files for mounting to non-container SQL Server instances.

Windocks Client Software

Windocks utilizes standard Docker client software 1.7.0.   The installation includes the Windows client, and the docker.exe can be copied to other machines and used with the Windocks host. Clients for linux and Mac machines are included below:

Docker Mac client

Docker Windows client

Docker Linux client

Get Started with Windocks using the CLI

With the Windocks Service running, open a command prompt on the local host and start using docker commands. The **>docker images** command provides a list of available images.   Create your first container by using the **>docker create** command.   Once the container is delivered, start the instance with **>docker start <containerid>.** The **>docker ps** command provides a list of the containers, and confirms the instance is running.

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Pau>docker images
REPOSITORY          TAG             IMAGE ID          CREATED          VIRTUAL SIZE
dotnet-4.5          none            dotnet-4.5        2 years ago      0 B
windows             none            windows           2 years ago      0 B
mssql-2014          none            mssql-2014        2 years ago      0 B

C:\Users\Pau>docker create mssql-2014
ContainerId = 0dadd55fd27bc774e1f86bcfdaee873a67c86b3fae8b72f826e87916541acd1a & ContainerPort = 10002 & Containe
rUserName = prison_oo_1toX9Qb & ContainerPassword = Pr!5U405Z1ncOF & MSSQLPort = 10002 & MSSQLServerUserName = sa
 & MSSQLServerSaPassword = Pr!51O952QX5O5

C:\Users\Pau>docker start 0da
0da

C:\Users\Pau>docker ps
CONTAINER ID        IMAGE           COMMAND           CREATED          STATUS           PORTS
       NAMES
0dadd55fd27b        mssql-2014      ""                About a minute ago   Started          10002/
       backstabbing_wilson/Windocks-id:58b6c6c7-55e0-4fd7-82ec-72256d0ba4a9

C:\Users\Pau>_
```

The example illustrates the use of the docker commands on a local machine.   For use on a remote client, the same commands are used, with some additional host details:

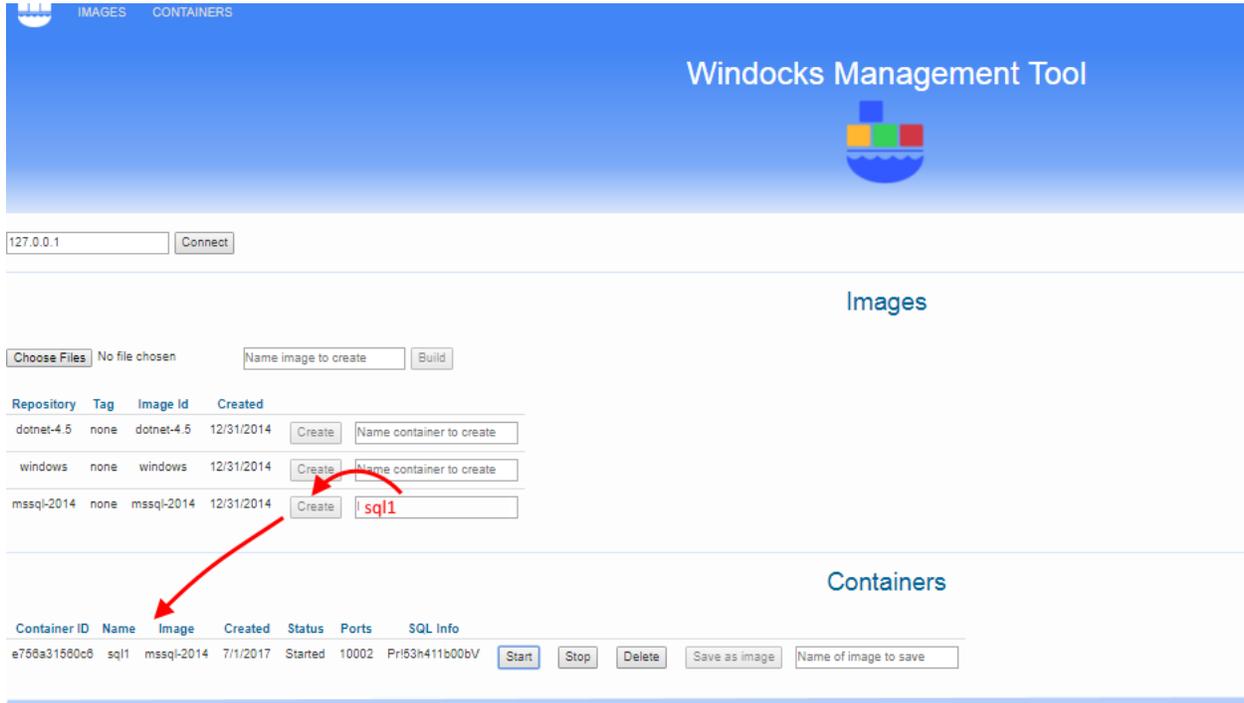**>docker -H=tcp://windocks.host.ip.address:2375 <command>**

Be sure the host is configured to allow inbound traffic on ports 10000 – 10200, the Docker daemon port 2375, and the SQL Server port 1433.

Get Started with the Windocks web UI

Windocks includes a web application that supports Chrome or Firefox browsers (IE is not supported). Open a Chrome or Firefox browser on the Windocks host, and direct the URL to local host.   Once the web page resolves, enter the local loopback address (127.0.0.1), and "connect." The available images (and containers) are presented.

Remote clients can use the web UI by opening a Chrome or Firefox browser, directing the URL to the IP address of the Windocks host, and once connected entering the IP address again in the IP address box, and "connecting."

Choose one of the SQLServer or dotnet images, assign a name, and "create."   The container will be presented below.   Start the container, and it is ready for use.

To access SQL Server containers using SQL Server Management Studio, refer to the container by using the loopback address, with a comma separator and port.   Server: 127.0.0.1,1000X

Additional Resources:

1) For work with larger and more complex database environments, see the companion article on [Getting Started with SQL Server containers with in container data](#)
2) [Windocks Command Line Reference](#)
3) Windocks containers operate with DNS: https://windocks.com/blog-2/docker-windows-containers-and-DNS
4) To understand Windocks licensing options for organizations: https://windocks.com/files/WinDocks_Licensing_and_Support.pdf
5) Forthcoming Jenkins CI pipeline support?   Email: info@windocks.com
6) For information on working with multi-tier environments, including .NET see: https://windocks.com/lps/gitbuildtest
7) For technical support email: support@windocks.com

For questions or technical support, email support@windocks.com