

Windocks Enterprise 20 AWS AMI

Prerequisites and recommendations

- 1) The Windocks Enterprise AWS AMI is a standard AWS AMI with Windows Server 2016 and SQL Server 2019 Express, supporting up to 20 simultaneous SQL Server, Postgres, and MySQL containers, with up to 1 TB of virtualized databases.
- 2) A 2 vCPU core server with 4 GB of RAM is sufficient for a handful of containers. A VM with 4 or 8 vCPU cores is recommended for up to 20 simultaneous SQL Server containers.
- 3) Provision with sufficient attached disk for full byte copy virtualized database images. A 1 TB image will require approximately 2 TB of storage, to support one image and backups used for image builds. A single image supports up to 20 writable database clones, each requiring only 40 MB on delivery. SSD disks are recommended.

Installation and configuration:

- Additional SQL Server images can be added by installing additional SQL Server instances using SysAdmin credentials with mixed authentication support. All releases and editions are supported from SQL Server 2008 to 2019. Once the additional instances are installed, they are configured to support SQL Server containers and database virtualization: <u>https://windocks.com/lps/installsqlafter</u>
- 2) A Chrome browser is installed to support the Windocks web UI. IE is not supported.
- Configure storage volumes for VHD images and containers, and set services for running of containers. See: <u>https://windocks.com/lps/doc#step5</u>
- 4) Open **services.msc** and restart the Windocks service following updates to \Windocks\config\node.conf
- 5) Create network and Windows firewall rules to allow inbound IP traffic for container ports (10000 to 10100), and for the web application (3000, or 3001 for TLS encrypted traffic).
- 6) Email <u>support@windocks.com</u> to obtain the default web UI administrator login credentials.
- 7) See <u>https://windocks.com/lps/docpostgremysql</u> for Postgres and MySQL image support.

Getting started:

- 1) Open a chrome or other browser locally on the server, and enter "localhost" or loopback address 127.0.0.1. The page will refresh to prompt for an administrative logon. Additional users and groups are implemented using the users/group tab on the web UI.
- Samples are available in \Windocks\samples. Navigate to the \clonefromfullbackup, and open the dockerfile using notepad. Edit/update the dockerfile to use a configured SQL Server image (ie., mssqlsvc-2016). Confirm available images using the Images tab on the

Private and Confidential, Governed under Mutual NDA



web UI. Save the dockerfile and avoid adding a .txt file extension by enclosing in double quotes, ie., "dockerfile" and save as "all files."

- 3) Click the build tab of the web UI, navigate to the **\clonefromfullbackup** folder, select the dockerfile and script, and assign an image name, and "build." The web UI will present the results on completion, with the image presented on the images tab view. "Refresh" the images page to see the latest results.
- 4) Using the newly built image, enter a container name (required) and click "deliver" for a fresh container. Optionally assign a port and sa login password. The web UI will refresh to present results.
- 5) Click on the "containers/clones" tab to see the container details. The container can be stopped/restarted and deleted.
- 6) Open SQL Server Management Studio or other database client, and connect to the running container, using the local server name and Instance1000X (ie., servername\Instance10001), using the Windows Administrator credentials, or SQL login or sa password. Remote clients should connect using the IP address of the server with a comma separator to the port (ie., 12.144.16.18,10001).

Release notes and tips:

- Locally installed SQL Server instances are set to a disabled/off state following the Windocks installation, to allow Windocks to clone the instance to deliver containers and support database virtualization. Do not restart the SQL Server instances while the Windocks service is running, to avoid corrupting the system databases. To update the SQL Server instances, first stop the Windocks service (run **services.msc**). Windocks requires the instances be put back to a manual/off or disabled/off state to resume use of Windocks.
- 2) Windocks Enterprise edition supports mssqlsvc, mssqlreport, and mssqlall images that deliver the database engine, database engine plus report server, and the full service stack container, respectively, as Windows services. The mssql image runs the database engine as a Windows process.
- 3) Linked servers and distributed transactions requires mssqlsvc or mssqlall images.
- 4) The Windocks server can be configured to present sa passwords for each container, with presentation in encrypted form, not presented, or in plaint text (1, 0, and 2 respectively). SHOW_SA_PASSWORD="1"
- 5) SQL user logins can be added to the configured SQL Server instances, to be inherited by each SQL Server container. This is a popular method for managing SQL login credentials for containers.
- 6) To remove an image, first delete all containers derived from the image, open a command prompt on the server and enter: **>docker rmi <imagename>**
- 7) The Docker commands are available via a local command prompt, but are not supported via remote clients. Remote client support is supported via RestFul API calls, or the web UI. See: <u>https://windocks.com/lps/restapi</u>
- Support for a distributed cross platform data repository, including Linux servers is now available. For information and support for Windocks on Linux email <u>support@windocks.com</u>.

Private and Confidential, Governed under Mutual NDA



Technical support:

Complete user docs are available at <u>https://windocks.com/lps/doc</u> Windocks support is available for general guidance and questions, at <u>support@windocks.com</u>. When facing an error or process that doesn't appear to be working, please email us with a complete description of the problem, and include the dockerfile, commands used, error return strings, and a copy of the platform log located in **\windocks\log\platform.log**.