



Deploy Digital Access component on Docker

This article describes how to deploy the [Smart ID Digital Access component](#) on Docker using docker swarm.

This article is valid for Digital Access 6.0.5 and later.

You can also deploy Smart ID complete package using docker compose for which you can refer to [Deploy Smart ID](#).



Known issue with the network card VMXNET3 on VMware virtualization platform

If you use the network card VMXNET3, there may be an issue with the docker swarm overlay network. The issue is that the swarm containers can ping each other (ICMP protocol), but TCP and UDP fail.

It is recommended to change the network card for all involved servers to E1000. This is done in the VMware configuration of the guest virtual machine.

This issue is only present on the VMware platform, and not on other virtualization platforms, to our knowledge.

Related information

- [Deploy Digital Access component](#)
- [Deploy Digital Access component as virtual appliance](#)
- [Do initial setup of Digital Access component](#)
- [Known limitations in Digital Access](#)
- [Docker swarm commands](#)

[Expand/Collapse All](#)

Prerequisites

The following is needed:

- Docker client and engine version 20.10.0 or later
- Linux that supports the above docker versions
- Software Components: Any of the PostgreSQL, MySQL, Sql-server, or Oracle database
- Configuration files: **setup.tgz**
- System Requirements: 8cpu with 32GB RAM and 100 GB physical space is recommended.
- For offline setup docker must be installed

See also "Preparations" below.

Preparations

- Download the **setup.tgz** file on system from the support portal.
- It is always recommended to take backup/snapshot before starting setup.
- If you have multiple network interfaces, note down the NIC name or IP address of the network interface which will be used by docker swarm.
 - When docker swarm is initialized it binds to one NIC and one IP address for communication, and if you have multiple NICs, you have to specify which one to use, by NIC name or IP address.
 - If you want traffic for swarm to be restricted to a private/internal network then provide an internal IP address. But if you want to expose some of your service then you need to provide your public IP address.

Step-by-step instruction

1. Copy **setup.tgz** to your working directory.
2. Extract the file using `tar` command.



Extract file

```
tar -xf setup.tgz
```

3. Navigate to the scripts folder inside the setup folder.

Navigate to scripts folder

```
cd setup/scripts
```

4. Run the script **setup.sh**.

Run the script

```
sudo bash setup.sh
```

5. During the setup, provide the below mentioned inputs.
 - a. You will need to input the IP address that will be used for the swarm communication based on the NICs. See the "Preparations" section above.

```
#####  
  
Setting up docker swarm...  
  
#####  
  
Enter ip address which should be used for docker swarm  
communication:
```

- b. To setup PostgreSQL press "Y" (Not recommended for production setup). If you want to use some external database press "n".

```
#####  
  
setting up PostgreSQL...  
  
#####  
  
Do you want to setup PostgreSQL (Not recommended in  
production environment) [Y/n] Y  
  
Creating a random password  
Creating network pg_default  
Creating service pg_postgres  
Create a new password for all databases  
New Password:  
Verify New Password:
```

- c. Next it will setup reporting database using PostgreSQL.



```
#####  
Setting up reporting database...  
#####  
  
Creating new customize.conf for administration service..
```

d. It will now pull the docker images and start the services

```
#####  
Starting Digital Access...  
#####  
  
Creating network da_da-overlay  
Creating service da_admin  
Creating service da_authentication  
Creating service da_distribution  
Creating service da_policy  
Creating service da_accesspoint  
Successfully deployed  
  
#####  
Setup script execution completed successfully  
#####
```

e. Wait for all the images to get downloaded. To check status of downloaded images, use this command:

```
Check images  
  
sudo docker images
```

| REPOSITORY | TAG | IMAGE ID | CREATED | SIZE |
|--|-------------|--------------|------------|--------|
| smartid-digitalaccess/distribution-service | 6.0.7.73936 | c626415de8b6 | 2 days ago | 588MB |
| smartid-digitalaccess/authentication-service | 6.0.7.73936 | 8d5b56cbf7da | 2 days ago | 589MB |
| smartid-digitalaccess/policy-service | 6.0.7.73936 | 4ed8d050e2f7 | 2 days ago | 662MB |
| smartid-digitalaccess/administration-service | 6.0.7.73936 | fe2774f4da20 | 2 days ago | 1.12GB |
| smartid-digitalaccess/access-point | 6.0.7.73936 | f65edd4ac6b4 | 2 days ago | 97MB |

f. After all the images are downloaded from the repository, the containers will start automatically. It can take a few minutes to start all the containers. To check the container status, use this command:

```
Check container status  
  
sudo docker ps
```

g. Make sure the status of each container is healthy.



Offline setup

Offline setups consist of two steps:

First step

For the first step you need a system with network connectivity. We will download docker images on this system.

1. Make sure docker is installed.
2. Download the images using the command docker pull.

pull images

```
sudo docker pull nexusimages.azurecr.io/smartid-digitalaccess
/administration-service:6.0.7.73936
sudo docker pull nexusimages.azurecr.io/smartid-digitalaccess/policy-
service:6.0.7.73936
sudo docker pull nexusimages.azurecr.io/smartid-digitalaccess
/authentication-service:6.0.7.73936
sudo docker pull nexusimages.azurecr.io/smartid-digitalaccess
/distribution-service:6.0.7.73936
sudo docker pull nexusimages.azurecr.io/smartid-digitalaccess/access-
point:6.0.7.73936
```

3. To check if all images are downloaded or not run this command.

Check images

```
sudo docker images
```

4. Save all images in **tar** files. Run these commands:

image to tar

```
sudo docker save nexusimages.azurecr.io/smartid-digitalaccess
/administration-service:6.0.7.73936 > admin.tar
sudo docker save nexusimages.azurecr.io/smartid-digitalaccess
/authentication-service:6.0.7.73936 > authentication.tar
sudo docker save nexusimages.azurecr.io/smartid-digitalaccess
/distribution-service:6.0.7.73936 > distribution.tar
sudo docker save nexusimages.azurecr.io/smartid-digitalaccess/policy-
service:6.0.7.73936 > policy.tar
sudo docker save nexusimages.azurecr.io/smartid-digitalaccess/access-
point:6.0.7.73936 > accesspoint.tar
```

5. Copy all the **tar** files onto the system where you want to setup Digital Access.

Second step

1. On the system where the Digital Access setup shall be done:
 - a. Go to the folder where all the tar files are copied.
 - b. Load the images to a local docker repository using the command:



load Images

```
sudo ls -l *.tar | xargs --no-run-if-empty -L 1 sudo docker  
load -i
```

- c. Run the installation process, and follow the steps mentioned in "**Step-by-step instruction**" section above.