random.blackpacket.net /2021/01/10/docker-nfs-and-cifs-volumes.html

Docker NFS and CIFS Volumes

Jan 10, 2021 • Tyler • Docker • NFS • Samba • CIFS

It’s hard to find information about how to get Docker to mount NFS exports and CIFS shares as volumes.

This is very handy to do, because it allows any node in your Docker cluster to connect to the same volume, as opposed to local volumes, which are only available on one node.

When using these, beware of a couple caveats:

- Running `docker volume create` doesn’t actually mount anything. This only happens when a container first attaches to the volume, and that’s when the problems will appear if you got anything wrong. So be sure to test using an actual container. (eg: `docker run -it --rm -v my-first-volume:/test alpine:latest /bin/sh` and look in /test)
- If you are testing different configuratons, beware that Docker does not “update” volumes, so especially if you are using a docker-compose approach, those containers are typically left in tact, and any change you make in the docker-compose file are ignored. You need to delete the volume first: `docker volume rm my-first-volume`

NFS

To manually create the volume on the command line:

```
docker volume create --driver local --opt type=nfs --opt o=addr=server.hostname.com,nolock,soft,rw --opt device=:/server/export/path nfs-volume-test
```

In docker-compose:

```
volumes:
  nfs-volume-test:
    driver_opts:
      type: "nfs"
      o: "addr=server.hostname.com,nolock,soft,rw"
      device: ":/server/export/path"
```

CIFS

To manually create the volume on the command line:

```
docker volume create --driver local --opt type=cifs --opt o=username=username,password=secret,rw --opt device=//192.168.0.2/Share cifs-volume-test
```

In docker-compose:

```
volumes:
  cifs-volume-test:
    driver_opts:
      type: "cifs"
      o:
"username=username,password=secret,uid=1000,gid=1000,file_mode=0777,dir_mode=0777,noexec,nosuid,nosetu

      device: "//192.168.0.2/Share/"
```

NOTE: You need to use the IP address when doing CIFS mounts. Although you may be able to mount it using `mount -t cifs //hostname/share` on the command line, docker does not use the same userland tools to do the mount, and will fail if a hostname is provided.

You may not need the `vers=1.0`, that downgrades the SMB protocol to version 1, which can provide a much more stable connection, at the cost of poor security.