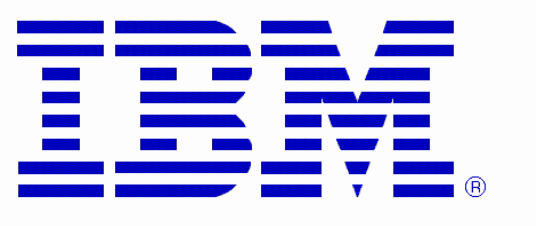


WebSphere Application Server Troubleshooting and Performance Lab on Docker - Quick Start

Authors

- Kevin Grigorenko (kevin.grigorenko@us.ibm.com)

Contents

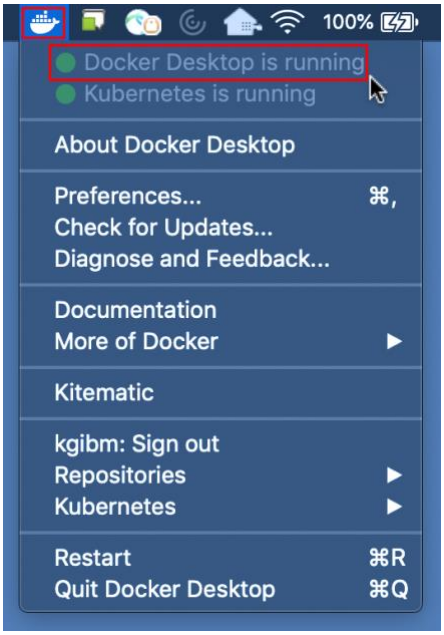
1 Install Docker

1. Install Docker:
    a. Windows ("Requires Microsoft Windows 10 Professional or Enterprise 64-bit.")
        - Download: https://hub.docker.com/editions/community/docker-ce-desktop-windows
        - For details, see https://docs.docker.com/docker-for-windows/install/
    b. Mac ("Requires Apple Mac OS Sierra 10.12 or above")
        - Download: https://hub.docker.com/editions/community/docker-ce-desktop-mac
        - For details, see https://docs.docker.com/docker-for-mac/install/
    c. For a Linux host, simply install and start Docker (sudo systemctl start docker):
        - For an example, see https://docs.docker.com/install/linux/docker-ce/fedora/

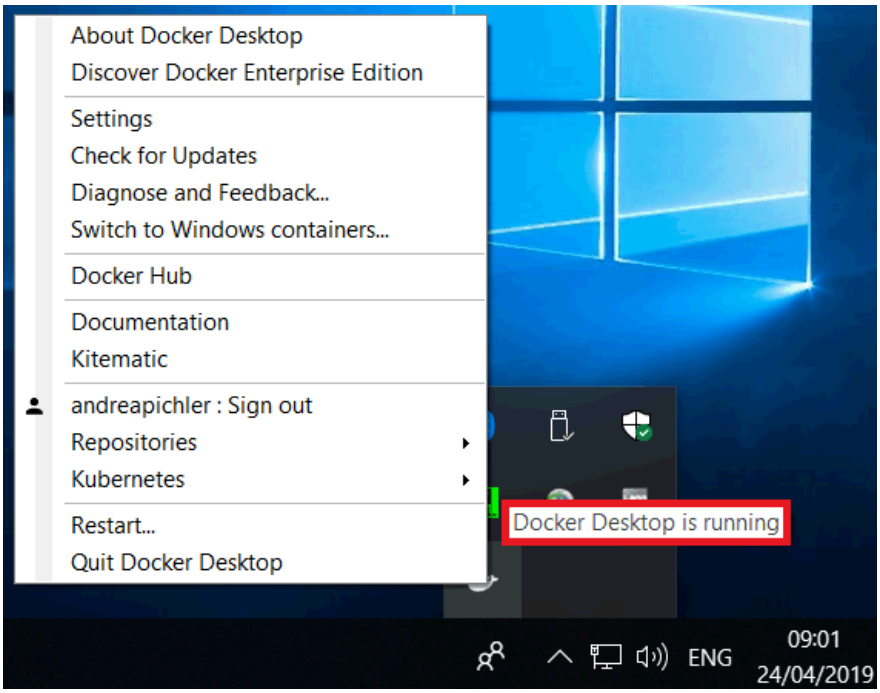
2 Configure Docker

1. Ensure that Docker is started. For example, start Docker Desktop and ensure it is running:

macOS:

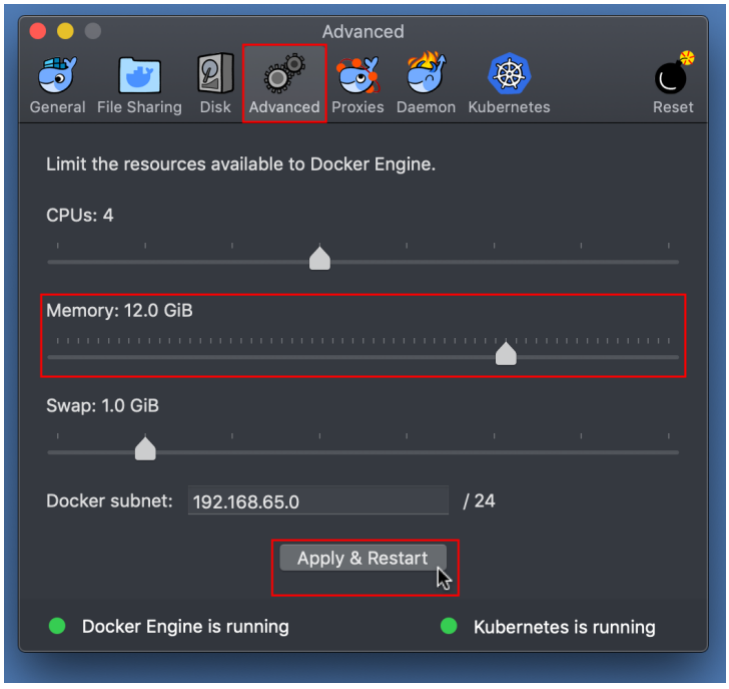


Windows:

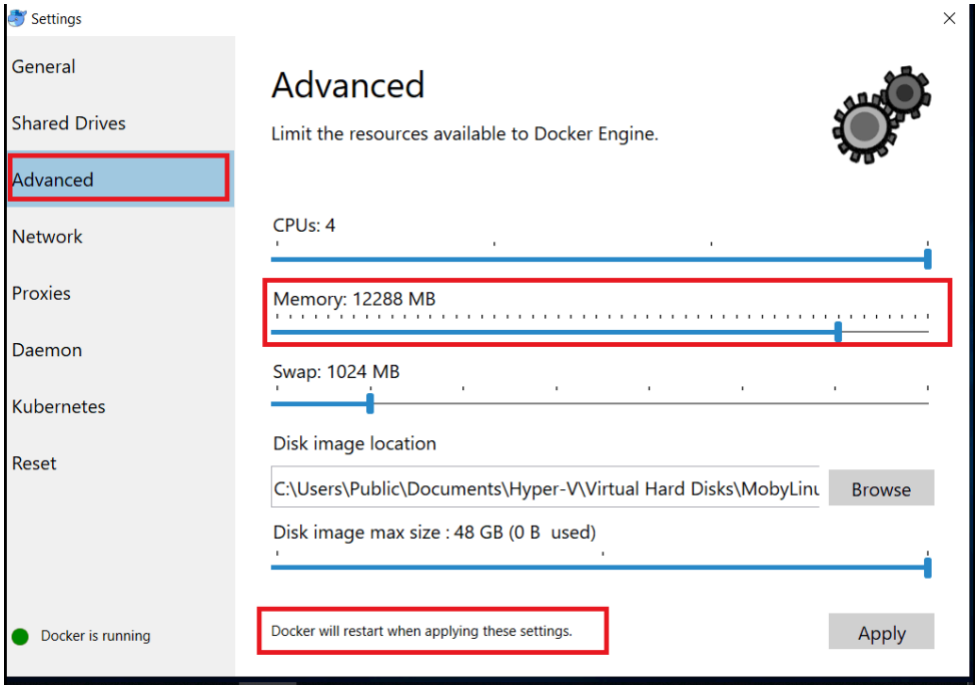


2. Ensure that Docker receives sufficient resources, particularly memory:
    a. Click the Docker Desktop icon and select "Preferences…" (on macOS) or "Settings" (on Windows)
    b. Select the Advanced tab.
    c. Increase Memory is at least 4GB and, ideally, at least 8GB.
    d. Click Apply

    macOS:



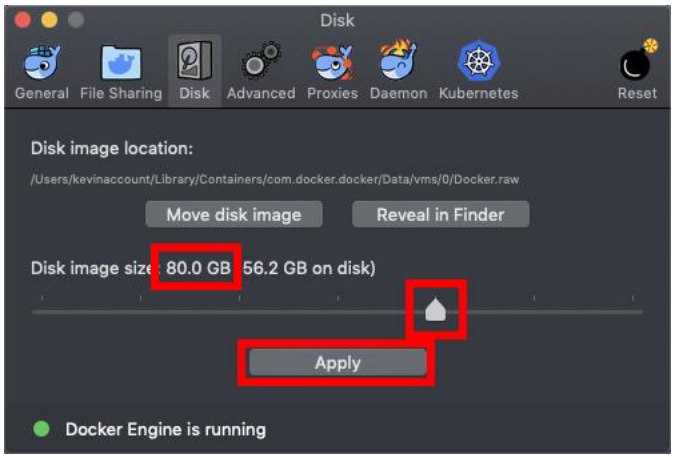
Windows:



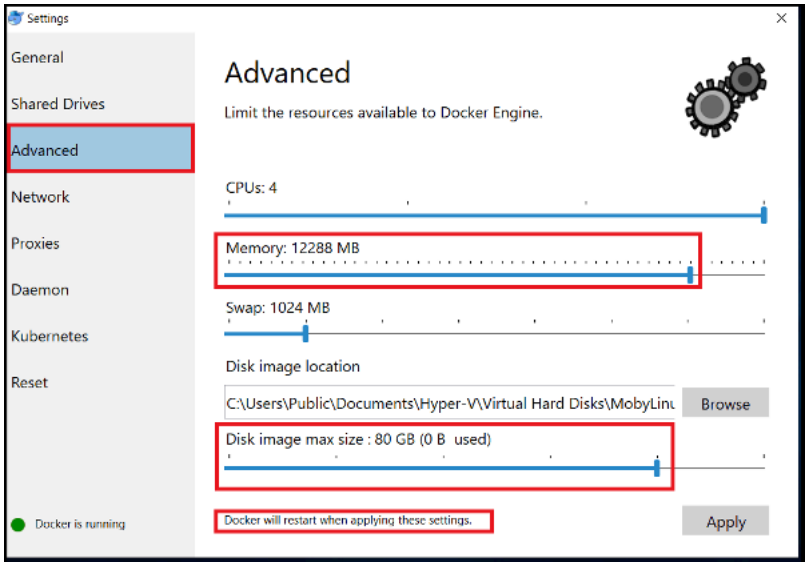
e. Select the **Disk** tab.

f. Increase the **Disk image size** to at least 80GB and click **Apply**:

macOS:

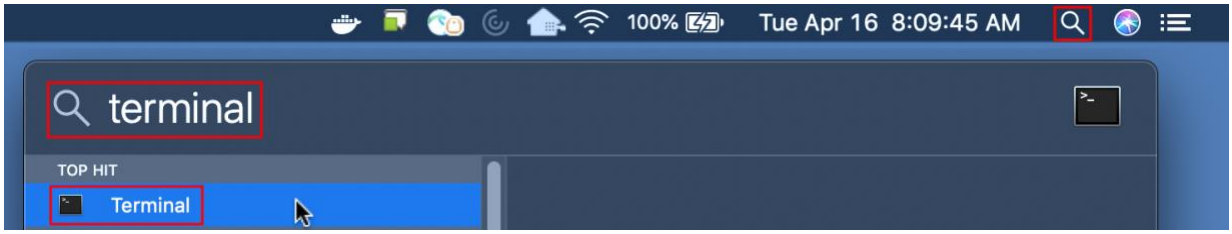


Windows:

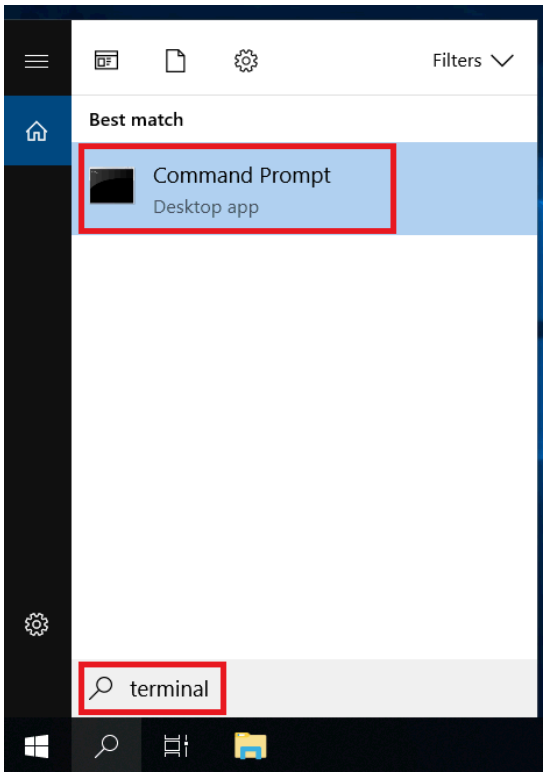

Settings
General
Shared Drives
Advanced
Network
Proxies
Daemon
Kubernetes
Reset
Docker is running
Advanced
Limit the resources available to Docker Engine.
CPUs: 4
Memory: 12288 MB
Swap: 1024 MB
Disk image location
C:\Users\Public\Documents\Hyper-V\Virtual Hard Disks\MobyLinu
Browse
Disk image max size : 80 GB (0 B used)
Docker will restart when applying these settings.
Apply

3. Open a terminal or command prompt:

macOS:



Windows:



3 Download the Image

```
docker pull kgibm/fedorawasdebug
```

Note that these images are > 20GB. If you plan to run this in a classroom setting, consider performing all the steps up to and including this item before arriving at the classroom.

4 Run the Image

1. Start the lab by starting the Docker container from the command line:

```
docker run --cap-add SYS_PTRACE --ulimit core=-1 --ulimit memlock=-1 --ulimit
stack=-1 --shm-size="256m" --rm -p 9080:9080 -p 9443:9443 -p 9043:9043 -p 9081:9081
-p 9444:9444 -p 5901:5901 -p 5902:5902 -p 3390:3389 -p 22:22 -p 9082:9082 -p
9445:9445 -p 8080:8080 -p 8081:8081 -p 8082:8082 -p 12000:12000 -p 12005:12005 -it
kgibm/fedorawasdebug
```

2. Wait about 2 minutes until you see the following in the output (if not seen, review any errors):

```
=========
= READY =
=========
```

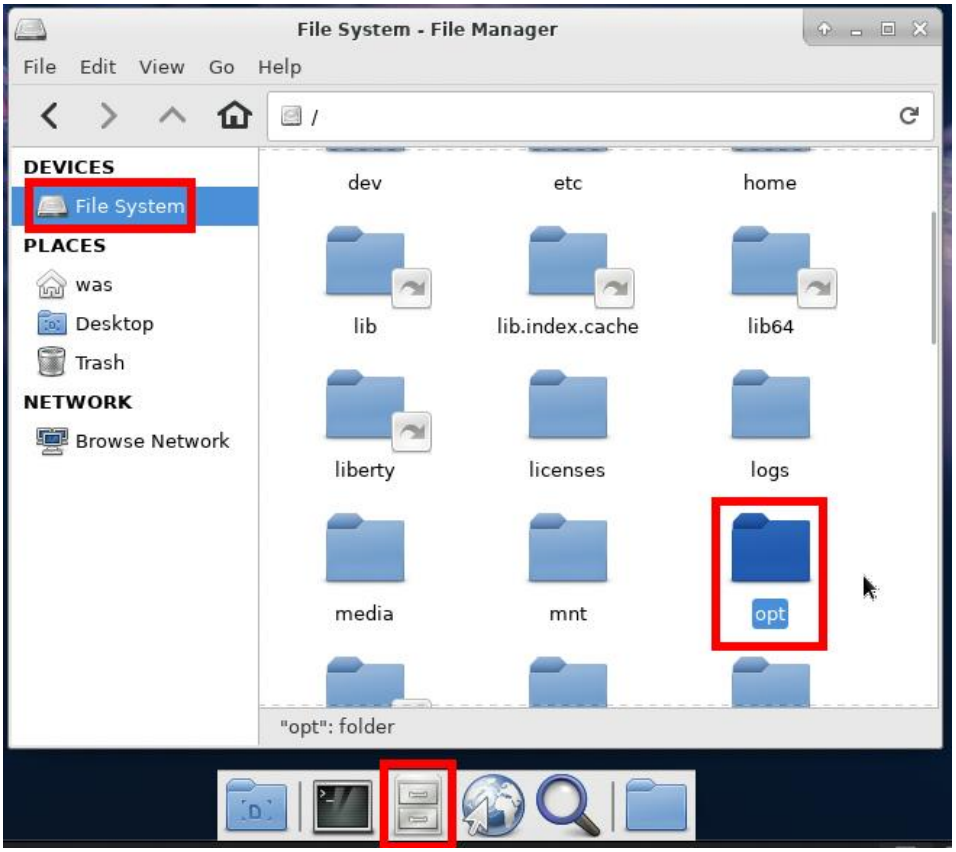
5 Remote into the Image

1. VNC or Remote Desktop into the container:
    a. macOS built-in VNC client:
        i. Open another tab in the terminal and run:
            1. open vnc://localhost:5902
            2. Password: **websphere**
    b. Linux VNC client:
        i. Open another tab in the terminal and run:
            1. vncviewer localhost:5902
            2. Password: **websphere**
    c. Windows 3rd party VNC client:
        i. If you are able to install and use a 3rd party VNC client (there are a few free options online), then connect to **localhost** on port **5902** with password **websphere**.
    d. Windows Remote Desktop client:
        i. Windows requires a few steps to make Remote Desktop work with a Docker container. See Appendix: Windows Remote Desktop Client for instructions.
    e. SSH:
        i. If you want to simulate a production-like environment, you can SSH into the container (e.g. using terminal ssh or PuTTY) although you'll need one of the GUI methods above to run most of this lab:
            1. ssh was@localhost
            2. Password: **websphere**
2. Test WAS Liberty by going to http://localhost:9080/daytrader/ in your host browser or the remote desktop/VNC browser.
3. Test Traditional WAS by going to http://localhost:9081/daytrader/ in your host browser or in the remote desktop/VNC browser.
    a. Test the Traditional WAS Administrative Console by going to https://localhost:9043/ibm/console in your client browser or in the remote desktop/VNC browser.
        i. User: **wsadmin**
        ii. Password: **websphere**

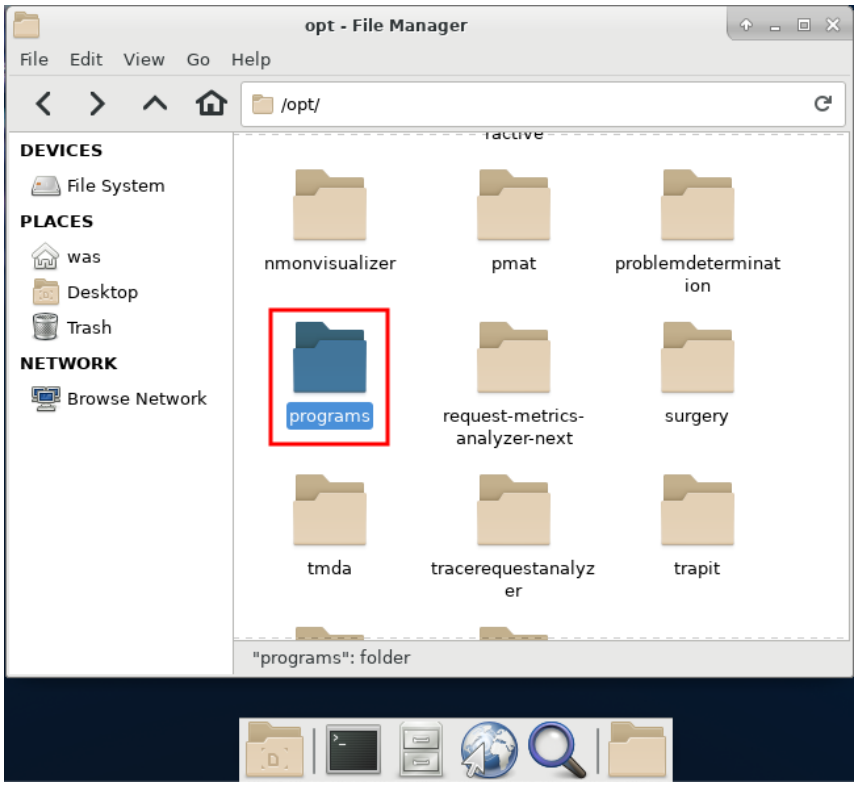
6 Run JMeter

Apache JMeter is a free tool that drives artificial, concurrent user load on a website. The tool is pre-installed in the lab image and we'll be using it to simulate website traffic to the DayTrader7 sample application pre-installed in the lab image.

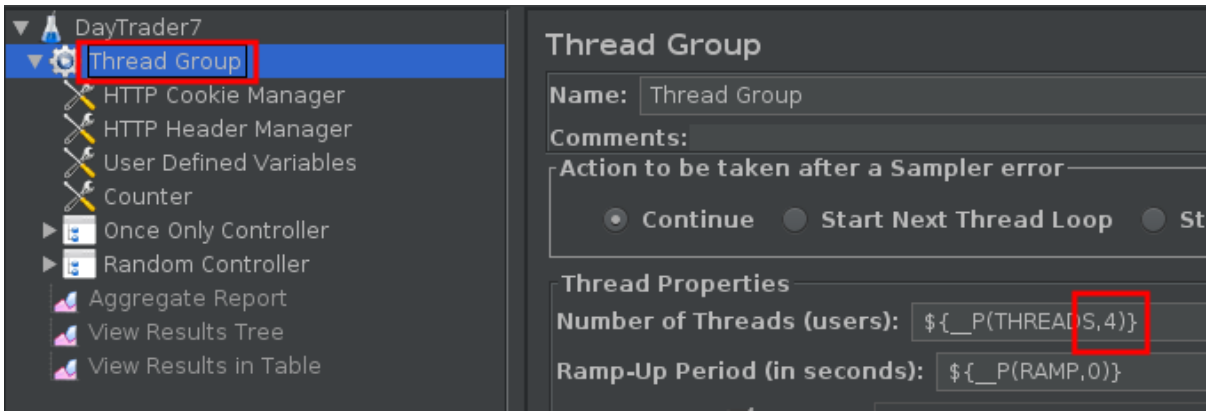
1. Open File Manager and navigate to /opt:



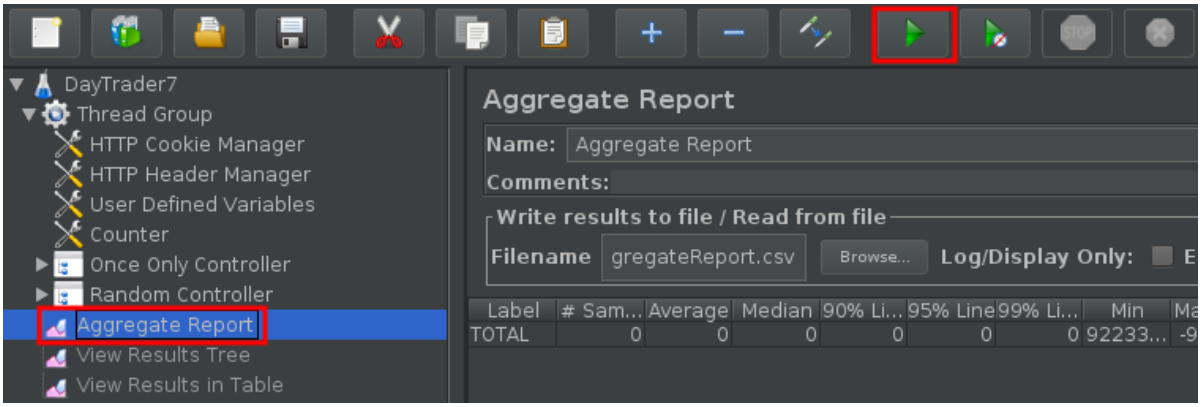
2. Navigate to **/opt/programs/**:



3. Double click on **JMeter**
4. Click **File → Open** and select **/opt/daytrader7/jmeter_files/daytrader7_liberty.jmx**
5. By default, the script will execute 4 concurrent users. You may change this if you want (e.g. based on the number of CPUs available):

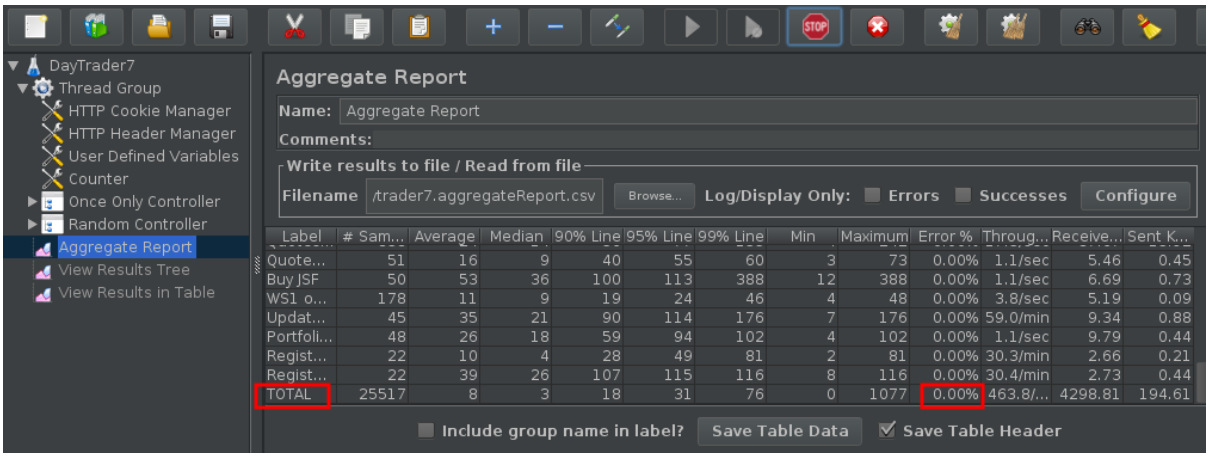


6. Click the green run button to start the stress test and click the **Aggregate Report** item to see the real-time results.



7. It will take some time for the responses to start coming back and for all of the pages to be exercised.

8. Ensure that the **Error %** value for the **TOTAL** row at the bottom is always 0%. If there are any errors, review the WAS logs.

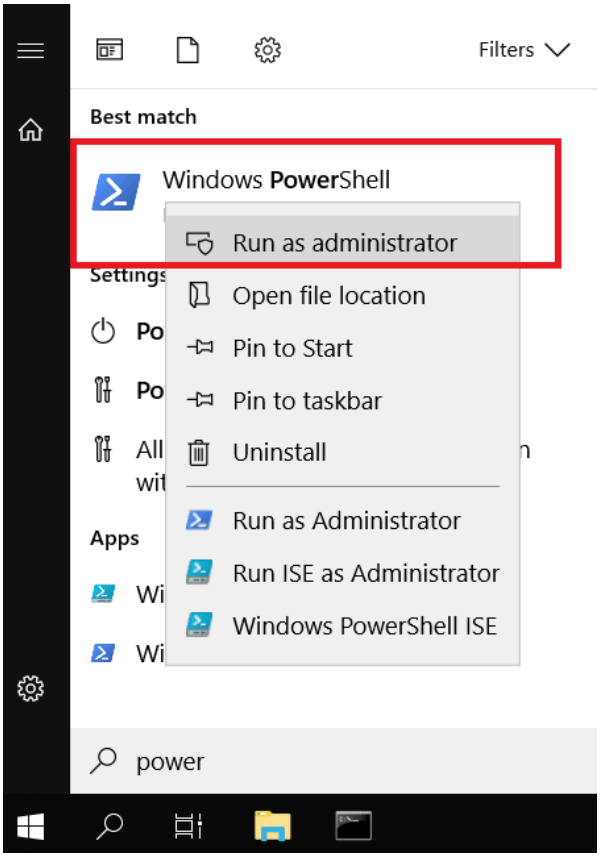


7 Appendix

7.1 Windows Remote Desktop Client

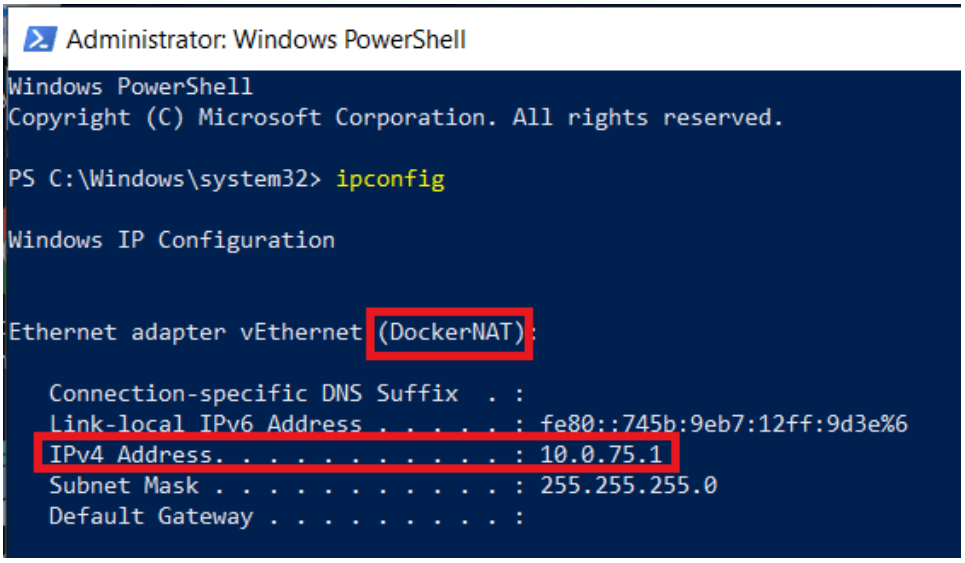
Windows requires extra steps to configure remote desktop to connect to a container[1]:

1. Open PowerShell as Administrator:



2. Run ipconfig and copy the IPv4 address of the DockerNAT adapter. For example:

[1] https://social.msdn.microsoft.com/Forums/en-US/872129e4-07a5-48c3-86f7-996854e7a920/how-to-connect-via-rdp-to-container?forum=windowscontainers



3. Run the following command in PowerShell:

```
New-NetFirewallRule -Name "myRDP" -DisplayName "Remote Desktop Protocol" -Protocol TCP -LocalPort @(3389) -Action Allow
```

4. Run the following command in PowerShell:

```
New-NetFirewallRule -Name "myContainerRDP" -DisplayName "RDP Port for connecting to Container" -Protocol TCP -LocalPort @(3390) -Action Allow
```