Gladius Desktop Client on Docker Toolbox running on Windows 7

Hello All

This is an early exposure of the Gladius Desktop Client running on Docker Toolbox on Windows 7. Took a little bit of work but with the help of Alex at Gladius I got it to work. Very promising platform.

The file for download is an OVF file for virtualization.

User: gladius Pass: gladiuspw

root password is also gladiuspw

The IP address is configured to be static. You can change the network interface setup to suite your purpose.

```
root@gladius1:/etc/network# cat interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
# iface eth0 inet dhcp
iface eth0 inet static
address 192.168.1.66
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameservers 8.8.8.8

root@gladius1:/etc/network#
```

Upon boot up login. Then change to the gladius-node-client folder.

```
Ubuntu 14.04.5 LTS gladius1 tty1

gladius1 login: gladius
Password:
Last login: Sat Nov 25 17:26:18 EST 2017 on tty1
Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Sat Nov 25 18:08:40 EST 2017

  System load:  0.08              Processes:           126
  Usage of /:   46.6% of 8.50GB   Users logged in:     0
  Memory usage: 8%                IP address for eth0: 192.168.1.66
  Swap usage:   0%

  Graph this data and manage this system at:
    https://landscape.canonical.com/

0 packages can be updated.
0 updates are security updates.

New release '16.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2019.
gladius@gladius1:~$ sudo -s
[sudo] password for gladius:
root@gladius1:~# ls
gladius-node-client  hello-world
root@gladius1:~# cd gladius-node-client
root@gladius1:~/gladius-node-client#
```

Now start the Docker platform.

```
root@gladius1:~/gladius-node-client# docker-compose up -d --build
```

The first time you run this it will take about five minutes if you are setting it up yourself. However, now that it has been run once running this now will now take a few seconds.

```
root@gladius1:~/gladius-node-client# docker-compose up -d --build
Building web
Step 1/6 : FROM node:boron
 ---> e8c6ec267732
Step 2/6 : WORKDIR /usr/app
 ---> Using cache
 ---> 067337223581
Step 3/6 : COPY package.json .
 ---> Using cache
 ---> baa6a482ae01
Step 4/6 : RUN npm install --quiet
 ---> Using cache
 ---> 5585011f2bde
Step 5/6 : COPY . .
 ---> Using cache
 ---> c64594f4585c
Step 6/6 : RUN npm run setup
 ---> Using cache
 ---> 695cce4916d8
Successfully built 695cce4916d8
Successfully tagged gladiusnodeclient_web:latest
Starting gladiusnodeclient_web_1 ...
Starting gladiusnodeclient_varnish_1 ...
Starting gladiusnodeclient_web_1
Starting gladiusnodeclient_varnish_1 ... done
root@gladius1:~/gladius-node-client#
```

I had to do this next step initially with the new setup but you should not have to. I include it just in case and for your additional knowledge. There was a bug in the current Gladius Desktop Client that Alex shared with me and the following steps are the fix in case you are setting this up from scratch on your own. Once the Docker instance is running you need to do an additional step within the Docker instance itself. First you need to identify the Docker instance with the following command.

```
root@gladius1:~/gladius-node-client# docker container ls
CONTAINER ID        IMAGE                  COMMAND                  CREATED             STATUS
        PORTS                                        NAMES
3ff965c5935e        gladiusnodeclient_web  "npm start"              About an hour ago   Up About a
minute   0.0.0.0:3000->3000/tcp                      gladiusnodeclient_web_1
d71e831a3aa8        ceacms/varnish         "/usr/local/bin/chap…"   20 hours ago        Up About a
minute   0.0.0.0:80->6081/tcp, 0.0.0.0:81->6082/tcp  gladiusnodeclient_varnish_1
root@gladius1:~/gladius-node-client# _
```

The above command will identify the Docker instance for Gladius. Then you need to BASH into the Docker instance. Every time you start a new instance of Docker for Gladius it will have a new Docker ID. You will use that Docker instance ID in the next command.

```
root@gladius1:~/gladius-node-client# docker container ls
CONTAINER ID        IMAGE                  COMMAND                  CREATED             STATUS
        PORTS                                       NAMES
3ff965c5935e        gladiusnodeclient_web  "npm start"              About an hour ago   Up About a
minute   0.0.0.0:3000->3000/tcp                     gladiusnodeclient_web_1
d71e831a3aa8        eeacms/varnish         "/usr/local/bin/chap…"   20 hours ago        Up About a
minute   0.0.0.0:80->6081/tcp, 0.0.0.0:81->6082/tcp   gladiusnodeclient_varnish_1
root@gladius1:~/gladius-node-client# docker exec -it d71e831a3aa8
"docker exec" requires at least 2 arguments.
See 'docker exec --help'.

Usage:  docker exec [OPTIONS] CONTAINER COMMAND [ARG...] [flags]

Run a command in a running container
root@gladius1:~/gladius-node-client# docker exec -it d71e831a3aa8 /bin/bash
root@d71e831a3aa8:/# _
```

You are now inside the Docker instance that is running the Gladius client software you started earlier.

This is only necessary if you are setting up for the first time. Evidently it is as a result of a bug in the current version of the Gladius Desktop Client. Significant improvements are going to be made in the next versions and eventually Docker will not be included. But I wanted everyone to see the progress that Gladius is making on this initiative so I invested some time to create an easily implemented working version. The Gladius Team is still working on a Mining Pool but that will come soon. My goal is to have my own mining pool in my datacenter.

So anyway, this command seems to only have to be run once within the Gladius Docker Instance.

```
root@gladius1:~/gladius-node-client# node setup.js
root@gladius1:~/gladius-node-client#
```

Once that is done you should be able to open a web browser.

http://192.168.1.66:3000/

Soon the Gladius team will have some Mining Pools setup so that you can use the Gladius Desktop Client. I also plan to setup my own Gladius Mining pool so others can use it.

I look forward to the success of Gladius.

Arnold Villeneuve