# MultiChain on Mac OS X

# Install virtualbox

# Download Vagrant

# Create Vagrant Init File

Run vagrant init

this creates a vagrant file.

change this file to have multiple boxes:

*See point 8, here is the full updated vagrant file.*

Vagrant.require\_version ">= 1.6.2"

Vagrant.configure(2) do |config|

config.vm.box = "ubuntu/trusty64"

# config.vm.box = "https://cloud-images.ubuntu.com/vagrant/wily/current/wily-server-cloudimg-amd64-vagrant-disk1.box"

# config.vm.hostname = "Multichain"

# config.vm.network :private\_network, :ip => '10.4.4.4'

number\_of\_nodes = 3

number\_of\_nodes.times do |i|

config.vm.define "multichain-node-#{i}" do |node|

node.vm.hostname = "multichain-node-#{i}"

node.vm.network :private\_network, :ip => "10.4.4.#{i}"

node.vm.provider :virtualbox do |vb|

#vb.gui = true

vb.customize ["modifyvm", :id, "--memory", "512"]

vb.customize ["modifyvm", :id, "--natdnshostresolver1", "off"]

vb.customize ["modifyvm", :id, "--natdnsproxy1", "off"]

vb.name = "MultiChain-#{i}"

end

end

i += 1

end

config.ssh.forward\_agent = true

config.ssh.forward\_x11 = true

config.vm.provision "shell", :inline => "wget http://www.multichain.com/download/multichain-1.0-alpha-16.tar.gz"

config.vm.provision "shell", :inline => "tar -xvzf multichain-1.0-alpha-16.tar.gz"

config.vm.provision "shell", :inline => "cd multichain-1.0-alpha-16 && mv multichaind multichain-cli multichain-util /usr/local/bin"

end

# 4. Start up both Vagrant Boxes

$: vagrant up

this starts both machines

# 5. Connect to Blue Machine

$: vagrant ssh multichain-node-1

(we will see later that we can reach this with 10.4.4.2:6815)

# 6. Connect to Red Machine

$: vagrant ssh multichain-node-2

# 7. on each machine, download multichain as follows:

this has been performed automatically in the vagrant file so no need to worry about it anymore. Just putting the full instructions for completeness.

http://www.multichain.com/download-install/

Installing MultiChain – Linux

su (enter root password)

cd /tmp  
wget <http://www.multichain.com/download/multichain-1.0-alpha-27.tar.gz>  
tar -xvzf multichain-1.0-alpha-27.tar.gz  
cd multichain-1.0-alpha-27

sudo chmod 777 /usr/local/bin  
mv multichaind multichain-cli multichain-util /usr/local/bin (to make easily accessible on the command line)

# 8. on first machine Launch MultiChain and keep info.:

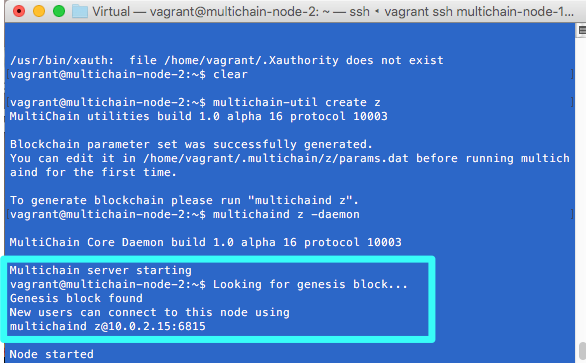
http://www.multichain.com/getting-started/

$: multichaind chain1 -daemon

this starts up multichain.

to see the beginning :

$: cat ~/.multichain/chain1/params.dat



Important is : Other nodes can connect to this node using:

multichaind chain1@10.0.2.15:6815

I found that this IP address cannot be used. (chain name and port are ok though)

instead I replaced the vagrant file with one a bit more complex, forcing an ip address, which I can find back when writing $ ip a

vagrant@multichain-node-2:~$ ip a

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00

inet 127.0.0.1/8 scope host lo

valid\_lft forever preferred\_lft forever

inet6 ::1/128 scope host

valid\_lft forever preferred\_lft forever

2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

link/ether 08:00:27:14:14:c5 brd ff:ff:ff:ff:ff:ff

inet 10.0.2.15/24 brd 10.0.2.255 scope global eth0

valid\_lft forever preferred\_lft forever

inet6 fe80::a00:27ff:fe14:14c5/64 scope link

valid\_lft forever preferred\_lft forever

3: eth1: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP group default qlen 1000

link/ether 08:00:27:a4:84:3c brd ff:ff:ff:ff:ff:ff

inet **10.4.4.2**/24 brd 10.4.4.255 scope global eth1

valid\_lft forever preferred\_lft forever

inet6 fe80::a00:27ff:fea4:843c/64 scope link

valid\_lft forever preferred\_lft forever

here is the full updated vagrant file

Vagrant.require\_version ">= 1.6.2"

Vagrant.configure(2) do |config|

config.vm.box = "ubuntu/trusty64"

# config.vm.box = "https://cloud-images.ubuntu.com/vagrant/wily/current/wily-server-cloudimg-amd64-vagrant-disk1.box"

# config.vm.hostname = "Multichain"

# config.vm.network :private\_network, :ip => '10.4.4.4'

number\_of\_nodes = 3

number\_of\_nodes.times do |i|

config.vm.define "multichain-node-#{i}" do |node|

node.vm.hostname = "multichain-node-#{i}"

node.vm.network :private\_network, :ip => "10.4.4.#{i}"

node.vm.provider :virtualbox do |vb|

#vb.gui = true

vb.customize ["modifyvm", :id, "--memory", "512"]

vb.customize ["modifyvm", :id, "--natdnshostresolver1", "off"]

vb.customize ["modifyvm", :id, "--natdnsproxy1", "off"]

vb.name = "MultiChain-#{i}"

end

end

i += 1

end

config.ssh.forward\_agent = true

config.ssh.forward\_x11 = true

config.vm.provision "shell", :inline => "wget http://www.multichain.com/download/multichain-1.0-alpha-16.tar.gz"

config.vm.provision "shell", :inline => "tar -xvzf multichain-1.0-alpha-16.tar.gz"

config.vm.provision "shell", :inline => "cd multichain-1.0-alpha-16 && mv multichaind multichain-cli multichain-util /usr/local/bin"

end

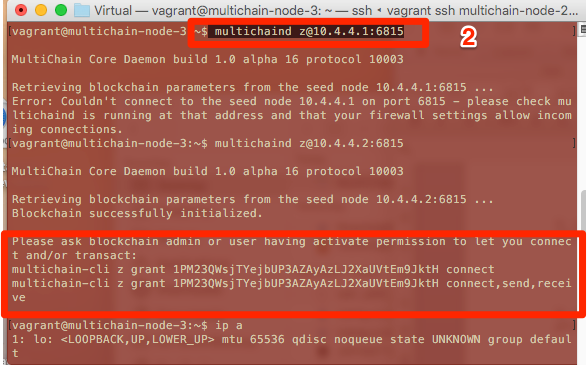
## 9. On second (Red) machine : multichain-node-2 : I try to connect towards the blue one : multichain-node-1

multichaind z@10.4.4.1:6815

where z = multichain name

where 10.4.4.1 = ip address

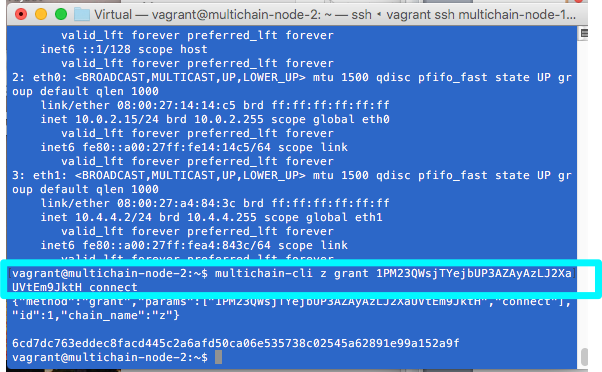
where 6815 = port provided in multichain info



We get an error . This is expected, because we don't have rights:

## 10. Back on blue server : grant permissions to red machine to connect

$ multichain-cli z grant 1PM23QWsjTYejbUP3AZAyAzLJ2XaUVtEm9JktH connect



## 10. Back on red server : connect to the multichain z

$ multichaind z -daemon

now the message is better:

MultiChain Core Daemon build 1.0 alpha 16 protocol 10003

Multichain server starting

vagrant@multichain-node-3:~$ Chain z already exists, adding 10.4.4.2:6815 to list of peers

New users can connect to this node using

multichaind z@10.0.2.15:6815

Node started

Blue Machine: I create an asset on the red machine with 1500.00 USD:

$ multichain-cli z issue 1PM23QWsjTYejbUP3AZAyAzLJ2XaUVtEm9JktH dollar 1500.00

Red Machine : I check the asset:

$ multichain-cli z listassets

vagrant@multichain-node-3:~$ multichain-cli z listassets

{"method":"listassets","params":[],"id":1,"chain\_name":"z"}

[

{

"name" : "dollar",

"issuetxid" : "c795172999f10f40a23a641c9e13decddcf008b33af94c6f192439b9a03bd76e",

"assetref" : "116-265-38343",

"multiple" : 1,

"units" : 1,

"details" : {

},

"issueqty" : **1500.00000000**,

"issueraw" : 1500

}

]

Blue Machine : Get the address:

$ multichain-cli z getaddressesbyaccount ""

{"method":"getaddressesbyaccount","params":[""],"id":1,"chain\_name":"z"}

[

"1BJL9q8v4XLawteW9qeuCXE8qxLpsvrCBzBS6j"

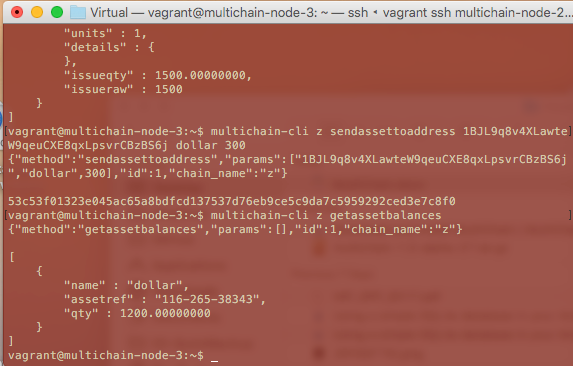
]

Red Machine : send money to blue :

$ multichain-cli z sendassettoaddress 1BJL9q8v4XLawteW9qeuCXE8qxLpsvrCBzBS6j dollar 300

Red Machine : check balance :

$ multichain-cli z getassetbalances



Blue Machine : check balance :

$ multichain-cli z getassetbalances

