**Installing MTS in a VirtualBox VM.**

**Version control**

|  |  |
| --- | --- |
| Created by | Marcos Lara |
| When | 30-Feb-2025. |
| Objective | Provide a step-by-step document to guide the installation of MTS on the VirtualBox machine. |
| Observation | This is a draft version. I created this from memory, so I need to perform these steps to validate the entire document. |

**Steps:**

* **Request the ‘Administrator’ user from your laptop (Optional)**
* **Download the VirtualBox app from the Company Portal.**
* **Install VirtualBox on your Laptop**

Recommendation: Create a VM in your VirtualBox (recommended)

Memory: 8GB or 16GB

Processors: 2

Disk: 80GB

* **Create a developer account on the RedHat website.**

This will allow you to download the Red Hat 8 ISO installation file for free.

* **Download the Red Hat 8 or Red Hat 9 ISO installation file.**

Alternatively, you can copy from my OneDrive, (C:\Users\brandinodelaram\OneDrive - ACI Worldwide Corp\Documents\MTS Group\rhel-8.10-x86\_64-boot.iso)

***Note:*** One of the steps in the installation requires registration; however, Global Connect running on the ACI laptops will prevent you from registering your VM in Red Hat. You will need to disconnect from Global Connect temporarily.

Once your VM is installed, you need to install VirtualBox Guest Addition.

Select ‘Devices’ and ‘Insert Guest Additions CD image.’ If you don’t have this installation, it can also be downloaded from the ‘Company Portal.’

* **Add a Host-Only interface.**

When you install a new virtual machine (VM), VirtualBox automatically creates a Network Address Translation (NAT) interface that provides Internet access for your VM and enables basic connectivity with your Windows host. However, to access the outside world, your VM needs a ‘Host-Only’ connection.

* **Configure your ‘Host-Only’ network interface.**

In the VirtualBox application, configure only one ‘Host-Only’ network interface. If you use a static IP, the IP assigned to this interface must not be used in any other VM; instead, the VMs should utilise a sequential one. For example:

If your Host-only has this IP:

192.168.20.20/24 (do not use it in any of your VMs)

Your VMs can use up to 24 IPs from there, so your first VM should use

192.168.20.21/24

The second:

192.168.20.22/24

The Thirty

192.168.20.23/24

So on...

* **Configuring the network interface in Red Hat**

$ sudo vi /etc/sysconfig/network-scripts/ifcfg-enpleg

TYPE=Ethernet

PROAY\_METHOD=none

BROWSER \_ONLY=no

BOOTPROTO=none

DEFROUTE=yes

TPV4\_FAILURE\_FATAL=no

TPV6TNIT=yes

EWG 1 AUTOCONF=yes

TRv6\_1 DEFROUTE=yes

DEFROUTE=yes

IpV4\_FAILURE\_FATAL=no

IPV6INIT=yes

IPV6\_AUTOCONF=yes

IPV6\_DEFROUTE=yes

Ipv6\_FAILURE\_FATAL=no

IPV6\_ADDR\_GEN\_MODE=eui64

NAME=**enp0s8**

DEVICE

ONBOOT=yes

TPADDR=**192.168.20.21**

Then:

$ sudo systemctl restart NetworkManager

* **Verify that your IP address was accepted.**

$ ip a | grep 192

Your **enp0s8** network device should have the IP you set up.

Once you have it installed.

Follow the ACI MTS installation guide.

Or follow the steps below:

* **Installing MTS**

***Note:*** The original ‘MTS installation guide’ asks us to create at least two volumes as follows:

/usr/intranet

/ust/locaVintranet

You may follow the manual, but I am installing everything in the same volume as this document.

*Note:* This will take some time.

Copy the bundle from the MTS QA machine we are using or from any working copy of MTS.

For example, you can use the host **cov3imtssve10vm** as your starting point

In your new VirtualBox VM, the DNS won't work. Therefore, use the IP from the host machine where you will copy the configuration and library files. For example, **cov3imtssve10vm** is **10.5.26.201**. If you decide to use another machine as your starting point, check its IP:

$ nslookup <DNS>

Example:

$ nslookup cov3imtssve10vm

Alternatively, you can ping your host, for example:

**From MS-DOS:**

C:\> ping cov3imtssve10vm

***Note:*** As a base, I am using the ASB bank set up in the above host.

**Instructions:**

* **In the new VM, log as the root**

$ sudo mkdir -p /usr/intranet/ship/asb/60/0015

@ files from: cov3lmtssve10vm,(10.5.26.201) to your new directory created

/usr/intranet/ship/asb/60/0015

(This will take a while; you can take a nap while you wait)

Copy the pakman utility from cov3lmtssve10vm. This usually comes with the installation packages, but you will need it to unpack the installation files.

(assuming you are in the same directory 00015)

$ cd /usr/intranet/util

$ scp username@10.5.26.201:/usr/intranet/ship/kiw/60/0001/pakman\*

* **Create an MTS account**

$ sudo useradd mtsadmin

* **Create an MTS group**

$ sudo groupadd mts

* **Add the mtsadmin to the group mts**

$ sudo usermod -a -G mts mtsadmin

* **Give the root privileges to the misadmin user**

$ sudo usermod -aG wheel mtsadmin

* **Disable password when you run mtsadmin commands**

$ sudo visudo

Modify his: %wheel ALL=(ALL) ALL

To this: %wheel ALL=(ALL) NOPASSWD: ALL

* **Install 32-bit compatibility libraries:(MTS runs in 32 bits, so we need some compatibility libraries).**

$ sudo dnf install epel-release (this is to add repo)

$ sudo dnf install glibc.i6686

$ sudo dnf install libstdc++.i686

$ sudo dnf install ncurses-libs.i686

(ASB needed to install this manually: ncurses-compat-libs.i686 )

* **Check the libraries added:**

$ rpm -ga | grep -i glibc

* **Create the entia directory**

$ sudo mkdir -p /usr/intranet/areas

$ sudo chown mtsadmin areas

$ cd /usr/intranet/areas

$ sudo mkdir -p entia/dev

$ sudo chown -R mtsadmin entia

$ sudo chgrp -R mts entia

$ sudo mkdir -p /usr/intranet/util

$ sudo mkdir -p /usr/intranet/install

$ mkdir -p /usr/local/intranet/areas

Procedures for creating a link to avoid SMP error:

$ cd /usr/local/intranet

$ chown mtsadmin areas

$ chgrp mts areas

$ sudo mkdir /usr/local/intranet/tmp

Copy the libraries below from a functional copy of MTS to your newly created directory, tmp.

[mesadmin@localhost tmp]$ ls -ltr

x--. 1 mtsadmin mtsadmin 136660 Oct 11 00:24 libhciecbl32.so

t-—. 1 mtsadmin mtsadmin 968888 Oct 11 00:24 libhciecbl1V32.so

$ cd /usr/local/intranet

$ chown mtsadmin areas

$ chgrp mts areas

$ chown mtsadmin entia

$ chgrp mts entia

* **Check if the links are correct for these libraries.**

$ **ldd** libhciecb132.so

linux-gate.so.1 (0xf7f£cS000)

libncurses.so.S => not found

libtinfo.so.5 => not found

dc++.s0.6 => /usr/lib/libstdet++.so.6 (0xf7e10000)

libm.so.6 => /ugr/lib/libm.so.6 (0xf7d3e000)

libe.so.6 => /usr/lib/libe.so.6 (Ox£7b95000)

libgce\_s.s0.1 => /usr/lib/libgec\_s.so.1 (0xf7b78000)

/iib/ld-linux.so.2 (0xf7f£c7000)

As you can see above, there are two libraries’ links ‘not found’ for fixing it:

$ mkdir libs

(Copy the specific libraries)

libneurses:so.S

Llibtinfo.so.5

ld-linux.so.2

libncurses.so-5

libtinfo.so.5

id-Linux.so.2

* **Create the links.**

$ In -sf /lib/libs/libncurses.so.5 libncurses.so.5

$ In -sf /lib/libs/libtinfo.so.5 libtinfo.so.5

* **Recheck your libraries**

$ cd /usr/local/intranet/tmp

$ ldd libhciecb132.so

iX-gate.so.1 (0xf7f£70000)

inses.so.5 => /lib/libncurses.so.5 (0xf7£26000)

9.50.5 => /lib/libtinfo.so.5 (0xf7£04000)

libncurses.so.5

libtinfo.so.5

1d-linux.so.2

* **Create the follow links.**

$ cd /lib

$ In -sf /1ib/libs/libncurses.so.5 libncurses.so.5

$ In -sf /lib/libs/libtinfo.so.5 libtinfo.so-.5

$ cd /usr/local/intranet/tmp

$ ldd libhciecbl32.so

\* linux-gate.so.1 (0xf7£70000)

jibncurses.so.5 => /lib/libncurses.so.5 (0xf7£26000)

libtinfo.so.5 => /lib/libtinfo.so.5 (0xf7£04000

libstdc++.so.6 => /lib/libstdct+.so.6 (0xf7d71000)

libm.so.6 => /lib/libm.so.6 (0xf7c9f000)

libe.so.6 => /lib/libc.so.6 (0xf7af6000)

libgec\_s.s0.1 => /lib/libgcc\_s.so.1 (Ox£7ad9000) I

libdi.so.2 => /lib/libdl.so.2 (0xf7ad4000)

/lib/lid-linux.so.2 (0xf7£72000

* **Check whether you possess the necessary MQ libraries; ideally, the maslient package should be installed.**

$ cd /usr/local/intranet/areas/dev/lib

$ ldd /usc/intranet/dev/nex/60/lib/libnex\_mq\_client.so

ldd: warning: you do not have execution permission for

\*/usr/intranet/dev/nex/60/1ib/libnex\_mq\_client.so’

linux-gate.so.1 (0xf7ed4000) ;

libmgic.so => /usr/lib/libmgic.so (0xf7ec3000)

libentia.so => /usr/intranet/dev/nex/60/lib/libentia.so (0xf7d£5000)

libnex\_message.so => /usr/intranet/dev/nex/60/1lib/libnex\_message.so (Oxf 7dead00)

libe.so.6 => /usr/lib/libc.so.6 (0xf7c41000) 3

libmge.so => /usr/lib/libmge.so (0xf72db000)

Jibdl.s0.2 => /usr/lib/libdl.so.2 (0x£72d6000)

libm.so.6 => /usr/lib/libm.so.6 (0xf£7204000)

rt.so.1 => /usr/lib/librt.so.1 (0xf71f£a000)

j-s0 => /usr/intranet/dev/nex/60/lib/libsbj.so (O0xf£6£57000)

ld-linux.so.2 (Oxf7ed6000)

jibmge.so => /usr/lib/libmge.so (0xf72db000)

libdl.so.2 => /usr/lib/libdl.so.2 (0xf72d6000)

libm.so.6 => /ugx/1ib/libm. so.6 (0xf£7204000

librt.so.1 => 7asr/lib/librt. so.1 (0xf71£a000)

libsbj.so => Jusr/intranet/dev/nex/60/lib/libsb}. so (0xf6£57000)

/lib/ld-linux.so.2 (Oxf7ed6000)

libstdc++.so.6 => /usr/lib/libstdct+.s0.6 (0xf6dc4000)

libpthread.so.0 => 7usr/1ib/libpthread. s0.0 (Oxf6da4000)

libace\_subs.so => /usr/intranet/dev/nex/60/1ib/libace\_ subs.so (Oxf6dal00 )

libnex \_entpf. so => /usr/intranet/dev/nex/60/lib/libnex\_entpf.so eebeaas0c )

libvid.so => /usr/intranet/dev/nex/60/lib/libvid.so (Oxfécbb000)

Jiblock.so => /usr/intranet/dev/nex/60/lib/liblock.so (Oxf£6EbS5000)

libgee 5.50.1 => /ugsr/lib/libgcc\_s.so.1 (0xf6c98000)

» libnex\_crypt.so => /usr/intranet/dev/nex/60/lib/libnex\_crypt.so (0xf6c93000)

libgce\_ 5.50.1 => /usr/lib/libgcc\_s.so.1 (Ox£6c98000) o

libnex\_crypt.so => Jusx/intranet/dev/nex/60/lib/libnex\_crypt.so (Oxf6c9

If you encounter any ‘**not found**’ messages, you should check your MQ libraries.

$ cd /opt/mqm/lib

$ ls -ltr | grep -l (filter only links)

libimgb23gl.s0 -> /opt/pgm/lib/libimqb23gl.s0.4.1

Libimgb23gl\_r.so -> /opt/mgm/lib/libimqb23gl\_r.s0.4.1

libimgs23gl.so -> /opt/mgm/lib/libimqs23gl.s0.4.1

Libimqs23gl\_r.so -> /opt/mgp/lib/libimqs23gl\_r.so.4.1

libimge23gl.so -> /opt/mam/lib/libimqe23gl.s0.4.1

Libimqe23gl\_r.s0 -> /opt/mgm/lib/libimqe23gl\_r.a0.4.1

amagéud -> /opt/mgn/libé4/amgefnd

amaziy -> /opt/mgm/libé4/amgzty

* If you decide to copy the MQ installation:

$ cd /opt/mqm

$ cd lib64

$ cd lib64

$ ls -ltr

total 500

-r-xr-xr--. 1 root root 6240 Oct 13 18:17 amqzfu

-r-xr-xr--. 1 root root 501832 Oct 13 18:17 amqzfd

(Copy these files if you don’t already have them.)

You can install the MQ client, but if you don’t have it, you can copy it from a machine where it is already installed.

MTS needs these links:

$ cd /usr/lib

Add the configuration below to your limits file:

$ sudo vi /etc/security/limits.conf

# Users need to be able to create more posix queues than the default 819200 would allow

hard msgqueue 21000000

soft msgqueve 21000000

hard nproc 32767

nofile 10240

fofile 10240

progadm Mproc 32767

progadm hproc. 32767

* **Run pakman as root.**

$ sudo pakman

ACI Worldwide Package Manager, version 2.06

Copyright 2001, 2009 ACI Worldwide. All Rights Reserved.

This Software is protected by United States Copyright Law and

international treaty provisions and its use is permitted in

accordance with the ACI Worldwide Software License Agreement as

further described in "help about”.

pakman >

* **Map the user and group.**

pakman> map group sys mts

pakman> map user sys mtsadmin

* **Load all packages.**

pakman > read /usr/intranet/ship/asb/60/0015/\*.gz

During the initial installation, you will be asked to create a username and a group name to access MTS. You can create the username' mtsadmin' and the group' mts' (or choose another name if you prefer).

* **List all packages loaded.**

pakman > list

* **Install the packages.**

Pakman > install 1 2 3 4 … (use spaces between the package numbers)

This will pop up a question about where you want to install MTS. Choose:

**dev** (for all questions. In this example, we are installing the **DEV** environment)

* **Verify your packages.**

pakman >verity 1 2 3 4\_..

* **Quit pakman.**

pakman > exit

* **Create your area.**

$ sudo su - mtsadmin

[mtsdmin@localhost ~]$ makearea dev

Pathname to product tree (e.g. /usr/intranet/test/mts/20):

**/usr/intranet/dev/mts/60** (set this)

Custom codes identifier (qa): **bnk**

Summary:

Area name: dev

Area directory: /usr/local/intranet/areas/dev

Product instance: dev

Product name: mts

Product version: 60

Custom id: **bnk**

NEX instance: dev

NEX version: 60

Build to area: <none>

Is this correct? (y/n): **y**

Area dev created.

* **Create the following soft links.**

$ which perl

/usr/bin/perl

$ cd /usr/local/bin

$ sudo ln -sf /usr/bin/perl perl

$ which bash

/usr/bin/bash

$ sudo ln -sf /usr/bin/bash bash

* **Check your area.**

$ showarea

Area name: dev

Area directory: /usr/local/intranet/areas/dev

Product name: mts

mts instance: dev

mts version: 60

mts customer: **bnk**

NEX instance:

NEX version: 60

***Note:*** You can initiate your MTS database from scratch or copy the database and the configuration files from one area that you have:

* **Copy the below configuration files to the ./config directory.**

$ cd config

cfg\_tab.dat

cfg\_ab.dat

mts.cfg

route.cfg

API1\_S0A\_1\_soa\_request\_types.properties

iso20022\_cashaccount38\_include\_bnk.xml

iso20022\_ctctdtls\_in\_include.xml

rfml\_msg\_history.xml

rfml\_outbound\_include\_011.xml

rfml\_soa\_inbound\_008.xml

rfml\_soa\_inbound\_008.xml

rtps\_transaction\_status\_interface\_request\_001.xml

$ scp username@**10.5.26.201**:/usr/local/intranet/areas/**mlara\_asb**/config/route.cfg

***Note:*** *In the example above, I am copying files from a functional host using the area name mlara\_asb.*

* **Create the link for MQ in the lib directory.**

$ ca dev

$ cd lib

$ ln -s /usr/intranet/dev/nex/60/1ib/libnex\_mq\_server.so libnex\_mq.so

* **Install OpenJDK (this is Java 32 and needs to be version 1.8)**

$ sudo yum update

$ sudo yum install java-1.8.0-openjdk-devel

$ java -version

Create JAVA link:

$ which java

/usr/bin/java

$ cd /usr

$ sudo mkdir java8

$ cd java8

$ sudo In -n /usr/bin/java java

**Note: License.** Follow the procedure in the document ‘MTS 6.0 ECobol License Installation’ to generate a license file.

Add the below source in the .bashrc of the mtsadmin.

$ vi ~/.bashrc

source `find /usr/intranet/dev/nex/60/bin -name nexrc -print | tail -1`

* **Try to load your area.**

$ ca dev

You are in area “dev”.

* **Check your installation.**

[mtsadmin@localhost dev]$ ent p

Entia [dev]: is DOWN

* **Try to load MTS**

$ mts -r (If you prefer to create your database from scratch, you can run the below

command before running mts -r).

**Creating a clean database.**

Additionally, to copy all configuration files, you will need the file ‘que\_init.dat.’

$ scp username@10.5.26.201:/usr/local/intranet/areas/mlara\_asb/config/que\_init.dat

$ mts\_dbinit **bnk** (Use three letters of the bank you want to create the area database)

Need more details.

Firewall: To access videoclient, you need to enable the below ports:

$ sudo firewall-cmd --zone=public --add-port=**20142**/tcp –permanent

**success**

$ sudo firewall-cmd --zone=public --add-port=**20143**/tcp --permanent

**success**

* **Running RGW as a container**

Please check the document “Running RGW as a container.docx”.

* **Troubleshooting**

If you have the below error:

COBOL Stack trace unavailable.

Use ‘SMAP\_DIR’ environment variable to point to SMAP file location or ensure they are on $CLASSPATH.

~ EC-PROGRAM-NOT-FOUND: Cannot find callable program ‘NEX\_MQ\_INITIALIZE.

.

.

Error calling 'NEX\_MQ\_INITIALIZE': Cannot find callable program ‘NEX\_MQ\_INITIALIZE’.

This is probably associated with MQ libraries, but before you do anything, try this:

* **Run the below commands:**

$ sudo mkdir /usr/local/intranet/tmp

Copy the following libraries from a functional MTS area copy.

$ cd /usr/local/intranet/tmp

-rwxr-r--. 1 mtsadmin mtsadmin 136660 Oct 11 00:24 libhciecbl32.so

-rwxr-r--. 1 mtsadmin mtsadmin 98888 Oct 11 00:24 libhciecblV32.so:

If it doesn’t work, the issue is related to the MQ installation.

* **Check your MQ Installation.**

MTS relies heavily on MQ installation, so if it is not installed correctly, you will encounter numerous errors.

You can decide whether to install the complete package for the MQ client or copy from an existing installation.

* **Copying MQ libraries.**

$ cd /opt/mqm

$ sudo scp -p [username@**10.5.26.201**:/opt/mqm/\*](mailto:username@10.5.26.201:/opt/mqm/*) .

Where: 10.5.26.201 refers to the IP address of a functioning copy of MTS where the MQSeries client was installed. The reason I use the IP address is that DNS is not operational in our VM.

* **Create the links:**

$ sudo mkdir lib

$ sud mkdir lib64

$ cd lib

$ sudo ln -sf /opt/mqm/lib/libimqb23g1:38:4:1 libimdb23g1:38

$ sudo ln -sf /opt/mqm/lib/libimqb2391\_F:58:4:1 libimqb23g1\_f:38

$ sudo ln -st /opt/mqm/lib/libimgs23g1.38.4:1 libimgs23gi.so

$ sudo ln -sf /opt/mqm/lib/libimgs23g1\_f:58:4:1 libimgs23g1f:so

$ sudo ln -sf /opt/mqm/lib/libimqc23g1.s6.4.1 libimqc2391:38

$ sudo ln -sf /opt/mqm/lib/libimqc23g1\_r:50.4.1 libimgc23g1\_.so

$ sudo ln -sf /opt/mqm/lib64/amqzfu amqzfu

$ sudo ln -sf /opt/mqm/lib64/amqzfud amqzfud

$ cd ../lib64

$ sudo ln -sf /opt/mgm/1ib64/1ibimqb23gl\_r.so.4.1 1ibimqb23gl\_r.so

$ sudo ln -sf /opt/mqm/1ib64/libimqs23g1.so.4.1 libimqs23gl.so

$ sudo ln -sf /opt/mqm/1ib64/libimqs23gl\_r.so.4.1 libimqs23gl\_r.so

$ sudo ln -sf /opt/mqm/1ib64/libimqce23gl.so.4.1 libimqce23gl.so

$ sudo ln -sf /opt/mqm/1ib64/libimqc23gl\_r.so.4.1 libimqc23gl\_r.so