Notes from the AWS RDQM deployment

Please Note: The supplied template creates a public and private Network load balancer which to date I have not tested successfully. These components may e safely removed from the template.

The following AWS image was used :

RHEL:3.10.0-1062.1.2.el7.x86_64

The following MQ version was used:

Name: IBM MQ
Version: 9.2.0.0
Level: p920-L200710.TRIAL
BuildType: IKAP - (Production)
Platform: IBM MQ for Linux (x86-64 platform)
Mode: 64-bit
O/S: Linux 3.10.0-1062.1.2.el7.x86_64
O/S Details: Red Hat Enterprise Linux Server 7.7 (Maipo)

A VPC and subnets were created as shown below.

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Availability Zone	Availability Zone ID
Private1	subnet-08eddc7cb569cae65	available	vpc-873601e2	172.31.80.0/20	4091	-	ap-southeast-2c	apse2-az2
Private2	subnet-08a070d0ea7988800	available	vpc-873601e2	172.31.48.0/20	4091	-	ap-southeast-2a	apse2-az3
Private3	subnet-06ca1fa1b8fdca09c	available	vpc-873601e2	172.31.64.0/20	4091	-	ap-southeast-2b	apse2-az1
Public1	subnet-39f9e37f	available	vpc-873601e2	172.31.16.0/20	4089	-	ap-southeast-2c	apse2-az2
Public2	subnet-b35741c4	available	vpc-873601e2	172.31.32.0/20	4091	-	ap-southeast-2a	apse2-az3
Public3	subnet-ff9a6f9b	available	vpc-873601e2	172.31.0.0/20	4090	-	ap-southeast-2b	apse2-az1
RDQMPrivate1	subnet-091765a43bb4d5b6a	available	vpc-873601e2	172.31.128.0/20	4091	-	ap-southeast-2c	apse2-az2
RDQMPrivate2	subnet-05395e2bd880d6e35	available	vpc-873601e2	172.31.96.0/20	4091	-	ap-southeast-2a	apse2-az3
RDQMPrivate3	subnet-052de64f0c28afdb8	available	vpc-873601e2	172.31.112.0/20	4091	-	ap-southeast-2b	apse2-az1

Process for Installation

1. Follow the steps as per the instructions at https://github.com/ibm-messaging/mq-rdqm/tree/production_deployment/cloud/aws
2. In the section 'Installing IBM MQ, I deviated after the instruction 'CDMQ Server' and performed the following
   rpm -ivh MQSeriesRuntime*
   rpm -ivh MQSeriesServer*
   rpm -ivh MQSeriesGSKit*
3. Install the prereqs for the RDQM supporting packages
   sudo yum install libtool
   sudo yum install nfs-utils
   sudo yum intsalll net-snmp-utils
   sudo yum install perl-TimeDate
   sudo yum install perl-DateTime

sudo yum install psmisc
sudo yum install -y cifs-utils
sudo yum install lvm2
sudo yum install libtool-ltdl-devel

4. Install the RDQM packages
   cd Advanced/RDQM/Prereqs/pacemaker-1
   *rpm -ivh *.rpm*

   cd ../ kmod-drbd-9/
   *rpm -ivh *.rpm*

   cd ../drbd-utils-9/
   *rpm -ivh *.rpm*

5. Execute (based on requirements)
   sudo echo 'fs.file-max = 524288' >> /etc/systl.conf
6. Resume the instructions in the Github README in step 1 from
   /opt/mqm/bin/setmqinst -i -p /opt/mqm
7. Before you Create the AMI, make sure you copy the *setupRdqminstance* script and the *configureRdqm* script to /root, ensuring they are set as executable (chmod a+x)
8. Referring to the section in the Github README 'Deploy the Template', I used the following command

   aws cloudformation deploy --template-file aws_mq_template.yaml --stack-name rdqmStack --no-execute-changeset --capabilities CAPABILITY_NAMED_IAM

9. Configure the RDQM instances as per the instruction 'Create an RDQM'. Note that you may have to login to the secondary instances and use the crtmqm -sxs <QMNAME> to create the secondary instances.