RStudio Server on a NeCTAR cloud VM

Joey Gerlach - joey.gerlach@ersa.edu.au

21 July 2016

RStudio Server enables you to provide a browser based interface (the RStudio IDE) to a version of R running on a remote Linux server. Deploying R and RStudio on a server has a number of benefits, including:

- The ability to access your R workspace from any computer in any location;
- Easy sharing of code, data, and other files with colleagues;
- Allowing multiple users to share access to the more powerful compute resources (memory, processors, etc.) available on a well equipped server; and
- Centralized installation and configuration of R, R packages, TeX, and other supporting libraries.

There is documentation on setting up a VM running RStudio-server from scratch on the NeCTAR support page https://support.ehelp.edu.au/support/solutions/articles/6000098750-pawsey-installing-r-and-rstudio-in-the-cloud

This workshop will involve launching a VM using an image in which most of the set-up and installation is already done. Most of the steps in setting up the image are the same as the steps outlined in the documentation, and a quick list of the main commands used in setting up the image is included in the appendices of this guide.

By launching a VM with a prepared image, we can skip many of the steps that require some familiarity with BASH commands, and we can access RStudio almost immediately.

Launch a virtual machine with pre-installed RStudio-Server

1. Log on to the NeCTAR dashboard https://dashboard.rc.nectar.org.au/project

2. Select the correct project allocation in the top bar (if it starts with "pt-", it is your default trial allocation)

3. Select "**Access & Security**" under the "Compute" subheading in the left side main menu.

   1. Click "Create Security Group"

   2. Name the security group 'RStudioServer', with the description "port 8787 and 22". Click "Create Security Group"

   3. Click "**Manage Rules**" in the "Actions" drop-down menu.

   4. Click "**Add Rule**". Type "**8787**" under "**Port**". Under "**CIDR**", you choose a range of IP adresses that can access your VM through port 8787 (password protected). If you keep it as "0.0.0.0/0", you can access your VM from any computer, but anyone else might find a way to access it also. Here are the IP ranges for SA University networks. Use one of these, and the VM can only be accessed form a University computer (including an external computer using a VPN to access the University network).

      - 129.127.0.0/16 - University of Adelaide
      - 129.96.0.0/16 - Flinders
      - 130.220.0.0/16 - UniSA

   5. Repeat the last step for port "**22**" (which allows SSH access to the VM)

   6. Select "**Key Pairs**" and use "**Import Key Pair**" to upload the public key of a keypair you have created in PuttyGen (Windows users)
   or in a Mac/Linux terminal with `ssh-keygen -f ~/.ssh/keyname`.
   See https://support.ehelp.edu.au/support/solutions/articles/6000055376-launching-virtual-machines for instructions on setting up keypairs.

4. Select "**Images**" in the left side main menu. Click the "**Public**" tab at the top of the window.

5. Find the image named "**RstudioServer__eRSA__**" (select the most recent date if there is more than one) and select "**Launch Instance**" from the "Actions" drop-down menu.

6. Give your VM a name and choose a "**Flavor**" (size of the VM, e.g. m1.small).

7. Select the "**Access & Security**" tab. Select your Key Pair name, and select the "**RStudioServer**" security group you just created.

8. OPTIONAL: If you might use a volume storage attachment at some point, select the "**Availability Zone**" and choose "**sa**" from the drop-down menu.

9. Click "**Launch**". It may take a few minutes for the instance to boot.

Connect to the VM with SSH

See the support page:

Copy the IP Address of the instance from the NeCTAR dashboard.

- Windows - Set up a PuTTY connection with the IP address

- Mac/Linux - In the terminal app, enter

```
ssh -i <keyname> ubuntu@<IPaddress>
```

Set up the RStudio-Server VM

On your first connection to the VM on the command line, a startup script will automatically run to complete set-up:

You will be prompted to provide a password for user: "ubuntu" . This password will be used to access the RStudio-Server through a web browser.

This script ensures the storage disc mounted at /mnt is writable, and provides a symbolic link to this disc in the home directory. The pre-installed RStudio-Server is started. The shell script then deletes itself as it is designed to only run once, at set-up.

The commands to check the status and run RStudio-server are:

```
sudo rstudio-server verify-installation
sudo rstudio-server status
sudo rstudio-server restart
sudo rstudio-server start
sudo rstudio-server stop
```

Access RStudio in your web browser

In the address bar of your web browser, paste the IP address of your VM (fromt the NeCTAR dashboard: Instances page) followed by ":8787".
`XXX.XXX.XXX.XXX:8787`

This should bring up the RStudio login window. Enter the user : ubuntu
and the password that you set up.

The primary storage of the VM (i.e. the home directory) is generally very small. Data storage and computation should take place in the mounted ephemeral storage which is located at */mnt*. The linked folder brings this storage to the home directory.

When you look at the *files* tab in RStudio, you will see that the home directory is loaded and that it contains a folder called *"Mounted_Storage"*. This 'folder' is a symbolic link to the mounted ephemeral storage (*/mnt*). It is recommended to create an *R Project* within this `Mounted_Storage` directory for data storage and computation.

OPTIONAL: Use the following commands to learn about your VM disk structure: In the **Tools** menu at the top of RStudio, select **Shell**, then enter

- `lsblk`
- `df -hT`
- `htop`

Play with RStudio - load an interactive tutorial

```
install.packages("swirl")
library(swirl)
```

Choose a tutorial to load:

Beginner:

```
install_from_swirl("R Programming")
install_from_swirl("Open Intro")
install_from_swirl("Data Analysis")
```

Intermediate:

```
install_from_swirl("Getting and Cleaning Data")
install_from_swirl("Exploratory Data Analysis")
install_from_swirl("Regression Models")
```

Advanced:

```
install_from_swirl("Statistical Inference")
```

Access the tutorial:

```
swirl()
```

Accessing the Shell from RStudio

If you need to change any VM settings on the Linux command line, you can often do this from within the RStudio session rather than needing to get SSH access through PuTTY or Terminal.

In the **Tools** menu at the top of RStudio, select **Shell**. Enter any BASH commands to the VM from this window. Examples of use that you might need are:

Adding a new user

In the *Shell*, use the following command to add users to ensure the new user has access to the ephemeral storage.

```
sudo adduser --ingroup Ruser <name>
```

The new user can use RStudio using the same URL, but enter their own username and password. The new user will have their own home folder, and will need a link to the mounted ephemeral storage. From the new user's account, open the *Shell* and enter:

```
ln -s /mnt ~/Mounted_Storage
```

Updating R version

Every now and then, you should check for updates in R (and other packages). In the *Shell* enter:

```
sudo apt-get update
sudo apt-get upgrade
sudo rstudio-server restart
```

Appendix

The commands used to set up the RStudio Server Instance

```
### Set up CRAN repository and install latest R
sudo nano /etc/apt/sources.list

### add the following line to the file on a new line (without ##):
### deb http://cran.ms.unimelb.edu.au/bin/linux/ubuntu trusty/

sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys E084DAB9
sudo apt-get update
sudo apt-get install r-base
sudo apt-get install r-base-dev

### make the R library writable for all users
sudo chmod 777 /usr/local/lib/R/site-library

### Download and install RStudio Server for Ubuntu
sudo apt-get install gdebi-core
wget https://download2.rstudio.org/rstudio-server-0.99.893-amd64.deb
sudo gdebi rstudio-server-0.99.893-amd64.deb
rm rstudio-server-0.99.893-amd64.deb

### Install some useful tools
sudo apt-get install texlive lmodern libcurl4-openssl-dev

### A script (~/.bash_profile) was created
### to run once on launching the instance,
### and then delete itself:

sudo groupadd Ruser
sudo usermod -g Ruser ubuntu
sudo chown :Ruser /mnt
sudo chmod 770 /mnt
ln -s /mnt ~/Mounted_Storage
echo "Set a password for user 'ubuntu' "
sudo passwd ubuntu
sudo rstudio-server verify-installation
echo "Set-up is complete. Enter <IPaddress>:8787 in your web browser to access RStudio"
wait && rm ~/.bash_profile
```