

RStudio Containers (R 4.0.0 and above)

2021-August

Introduction

RStudio has been installed as a container on Seadragon. This particular container requires some special work, so a submission script is provided to submit the job. There are some things you should know about accessing R in the RStudio container:

- The version of R is located inside the container. It is not the same as the version available through modules.
- You will need to have a personal R library for the version of R used in the container.
 - (See <http://hpcweb.mdanderson>, select "User Guides," then "Personal application libraries.")
 - Use the corresponding version of R available in modules to create your library

To access RStudio, you will need to log into Seadragon with X forwarding.

- See <http://hpcweb.mdanderson>, select "User Guides," then "X Windows"

Process

```
$ module load rstudio-sing/4.0.0
or
$ module load rstudio-sing/4.0.2
or
$ module load rstudio-sing/4.0.4
```

Note: At the time of this writing, Seadragon has RStudio modules for R versions 4.0.0, 4.0.2, and 4.0.4. Here, we use 4.0.2 as an example for reference.

1. Copy `/risapps/singularity/repo/RStudio/4.0.2/rserver_4.0.2.lsf` to your `$HOME`.
2. Log into the cluster using X forwarding (see documentation on hpcweb).
3. Edit the `bsub` commands to match your needs.
4. Submit your job using the edited submit script. (For example, "`bsub < rserver_4.0.2.lsf`")
5. After submission, run "`bpeek <jobid>`", this provides you information that you need in next step.
6. Follow the information on the screen. Be sure to log in from your workstation, NOT Seadragon.
7. When you are done:
 - a. exit RStudio by pressing the red button in the upper right, and close web browser window
 - b. run the `bkill <jobid>` command on the cluster login terminal

- c. Type CTRL-C on your workstation terminal.

Examples

For login to Seadragon with VPN Macintosh user

With XQuartz installed and open, from terminal window

```
$ login ssh -Y username@seadragon
```

1. After modifying the job submission script rserve_4.0.2.lsf, submit the job in terminal:

```
[ryao@ldragon2 rstudio4.0.2]$ bsub < rstudio4.0.2.lsf
Job <144577> is submitted to queue <short>.
<<ssh X11 forwarding job>>
<<Waiting for dispatch ...>>
```

2. Use bpeek <jobid> to retrieve information

```
[ryao@ldragon2 rstudio4.0.2]$ bpeek 144570
Job <144570> is not found
[ryao@ldragon2 rstudio4.0.2]$ vi rstudio4.0.2.lsf
[ryao@ldragon2 rstudio4.0.2]$ bsub < rstudio4.0.2.lsf
Job <144577> is submitted to queue <short>.
<<ssh X11 forwarding job>>
<<Waiting for dispatch ...>>
[ryao@ldragon2 rstudio4.0.2]$ bpeek 144577
<< output from stdout >>
1. SSH tunnel from your workstation using the following command:

ssh -N -L 8787:cdragon296.cm.cluster:54451 ryao@seadragon

and point your web browser to http://localhost:8787

2. log in to RStudio Server using the following credentials:

user: ryao
password: 21HmSjlxEZnlGUSSqyOP

When done using RStudio Server, terminate the job by:

1. Exit the RStudio Session ("power" button in the top right corner of the RStudio window)
2. Issue the following command on the login node:

bkill 144577

The following have been reloaded with a version change:
1) gcc/7.2.0 => gcc/7.1.0

singularity exec --bind /rsrch3 --bind /home/ryao/logs/rstudio/RServer:/var/run/rstudio-server /risap
ps/singularity/repo/RStudio/4.0.2/rstudio_4.0.2.sif rserver --www-port 54451 --auth-none=0 --auth-pam
-helper-path=pam-helper
```

3. On your workstation, open a new terminal, use the information from bpeek <jobid> output

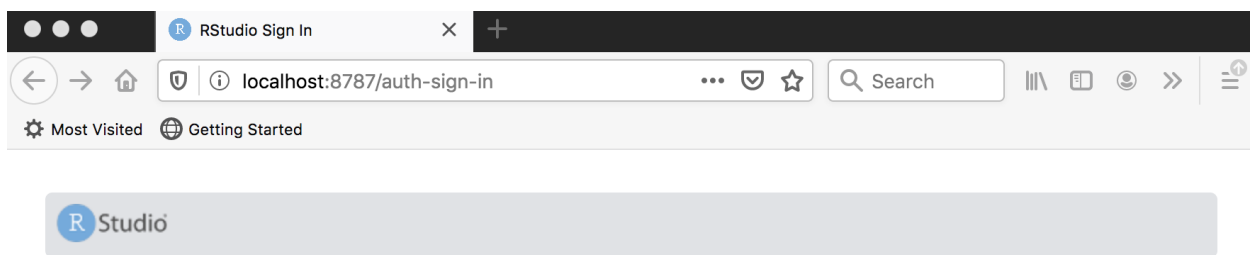
```
MDACM0CL5029737:~ ryao$ ssh -N -L 8787:cdragon296.cm.cluster:54451 ryao@seadragon
```

It prompts for Seadragon password, enter Seadragon password:

```
*** This system uses Active Directory authentication ***

ryao@seadragon's password:
█
```

4. Open a new web browser window on workstation, type localhost:8787 in address line, use the user and password information as seen in the above bpeek output:



Sign in to RStudio

Username:

ryao

Password:

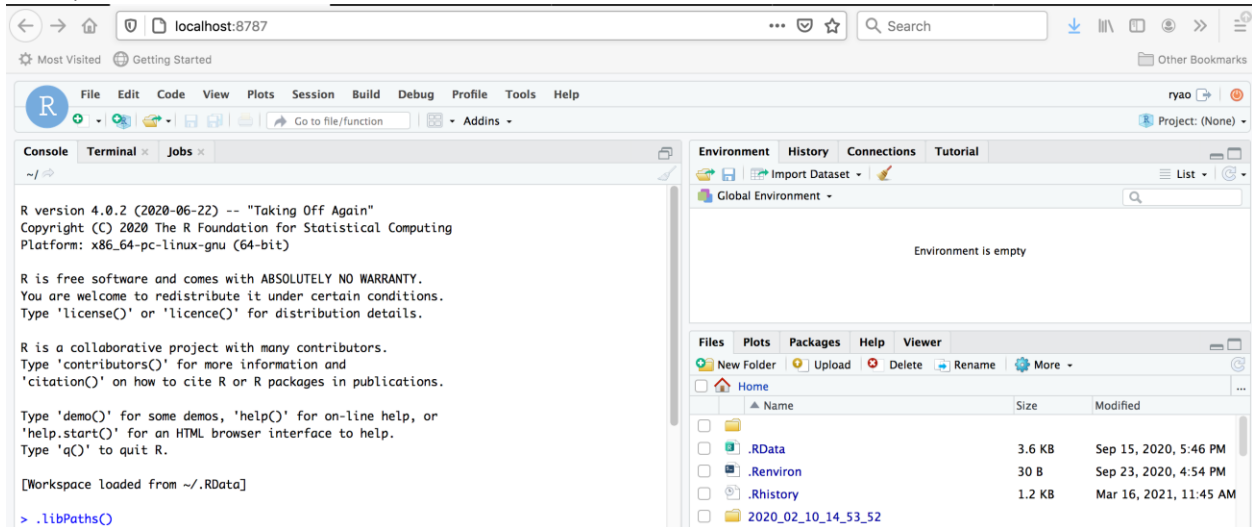
.....

You will automatically be signed out after 60 minutes of inactivity.

☐ Stay signed in when browser closes

Sign In

Ready to use RStudio in local web browser.

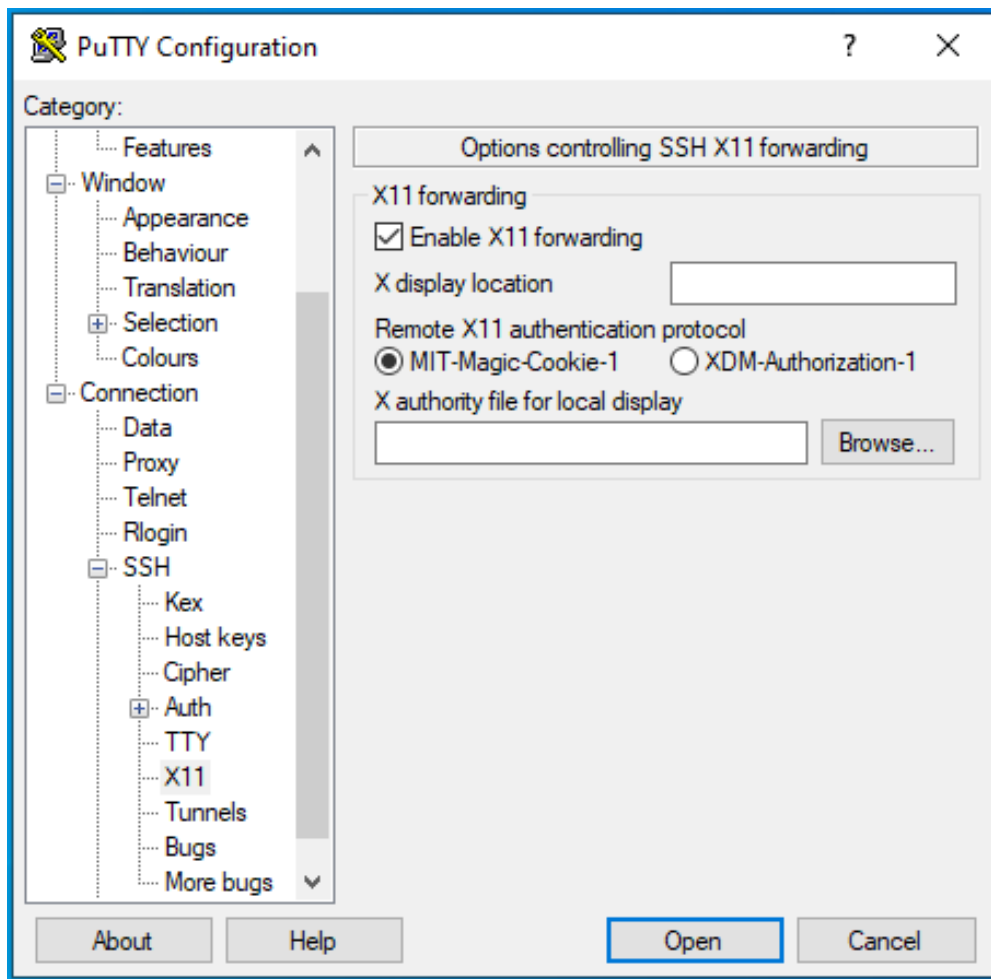


5. Exit RStudio.

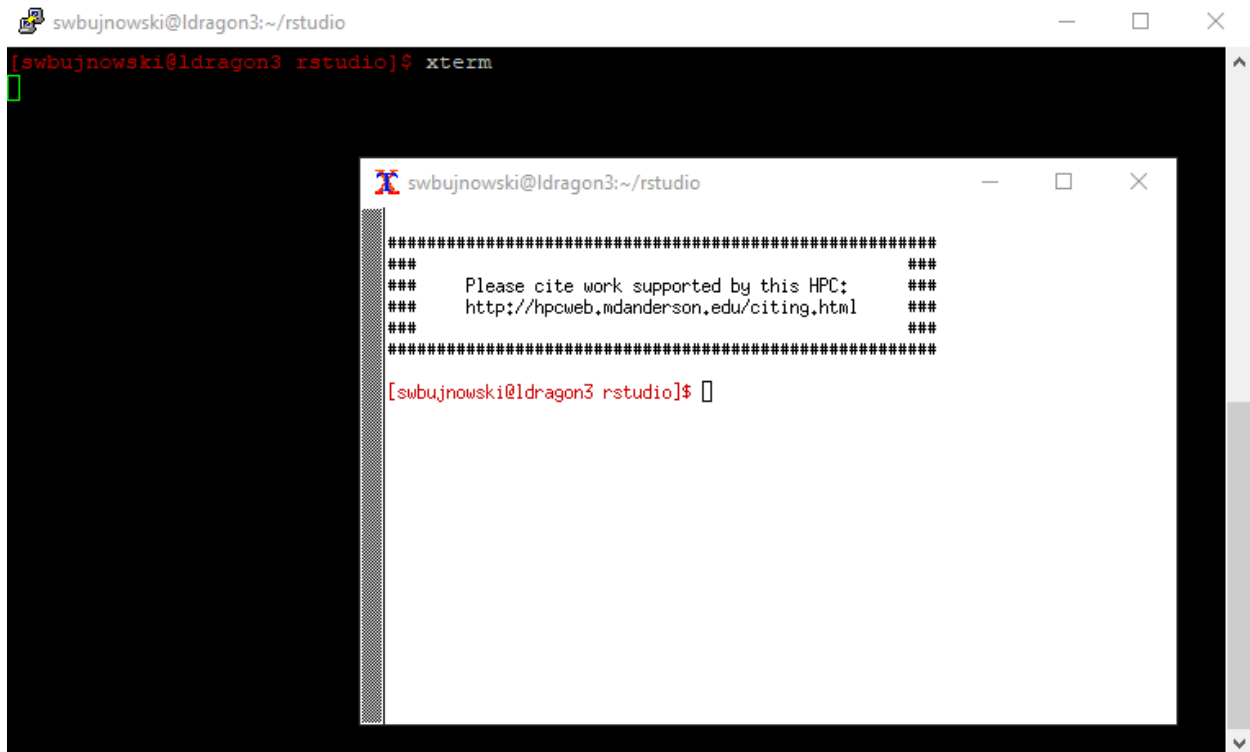
- exit RStudio by pressing the red button in the upper right, and close web browser window
- run the `bkill <jobid>` command on the cluster login terminal
- Type CTRL-C on your workstation terminal.

For login to Seadragon from a Windows PC using X-Win32:

- Launch X-Win32
- Login with PuTTY, but first make sure that X11 forwarding has been enabled:



3. As a test that X11 (X-Windows) is being forwarded properly, you should now be able to open “xterm” from your terminal window.



4. In the terminal session that opens up, modify the job submission script `rserver_4.0.2.lsf`, and then submit the job in the terminal window:

```
[swbujnowski@ldragon3 rstudio]$ bsub < rserver_4.0.2.lsf
Job <6072136> is submitted to queue <short>.
<<ssh X11 forwarding job>>
<<Waiting for dispatch ...>>
```

5. Use `bpeek <jobid>` to retrieve information:

```
[swbujnowski@ldragon3 rstudio]$ bpeek 6072136
<< output from stdout >>
```

1. SSH tunnel from your workstation using the following command:

```
ssh -N -L 8787:cdragon287.cm.cluster:59200
swbujnowski@seadragon
```

and point your web browser to `http://localhost:8787`

2. log in to RStudio Server using the following credentials:

```
user: swbujnowski
```

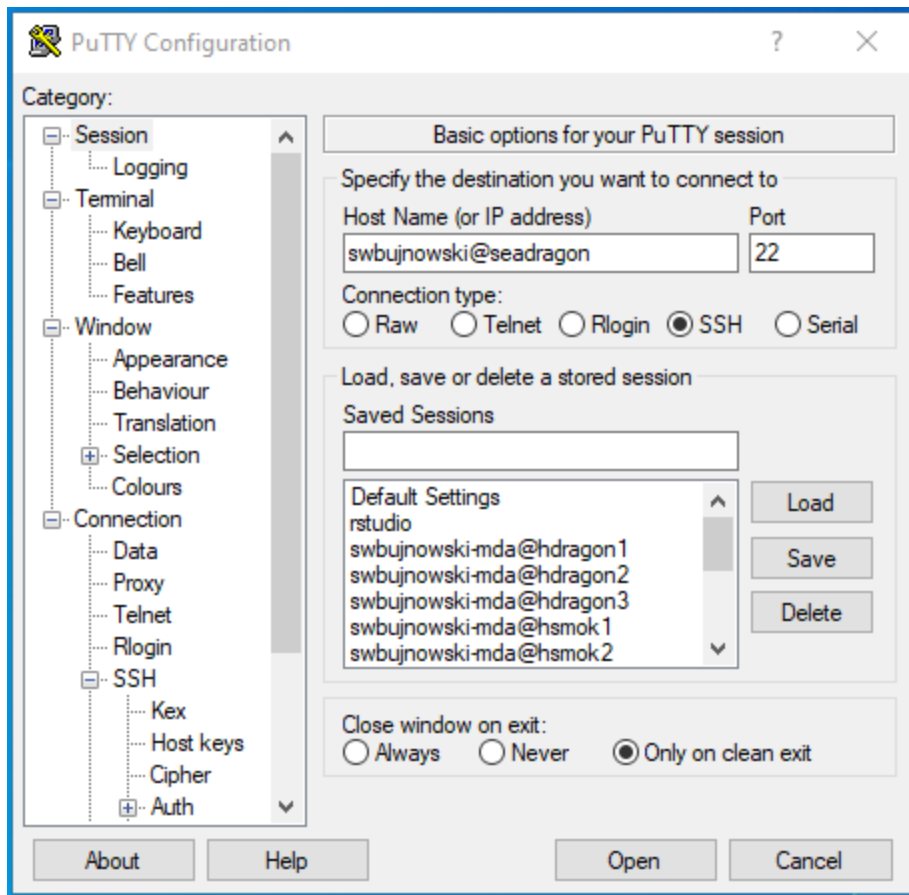
```
password: TERztR80URFmUutG4+99
```

When done using RStudio Server, terminate the job by:

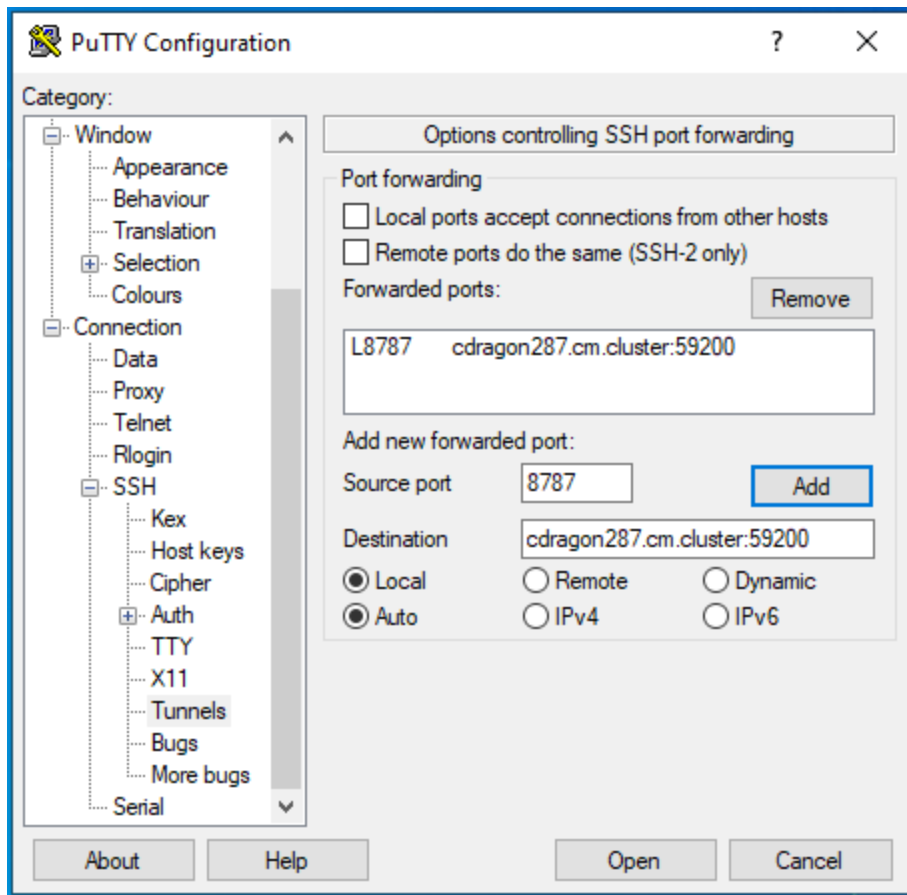
1. Exit the RStudio Session ("power" button in the top right corner of the RStudio window)
2. Issue the following command on the login node:

```
bkill 6072136
singularity exec --bind /rsrch3 --bind
/home/swbujnowski/logs/rstudio/RServer:/var/run/rstudio-
server
/risapps/singularity/repo/RStudio/4.0.2/rstudio_4.0.2.sif
rserver --www-port 59200 --auth-none=0 --auth-pam-helper-
path=pam-helper
[swbujnowski@ldragon3 rstudio]$
```

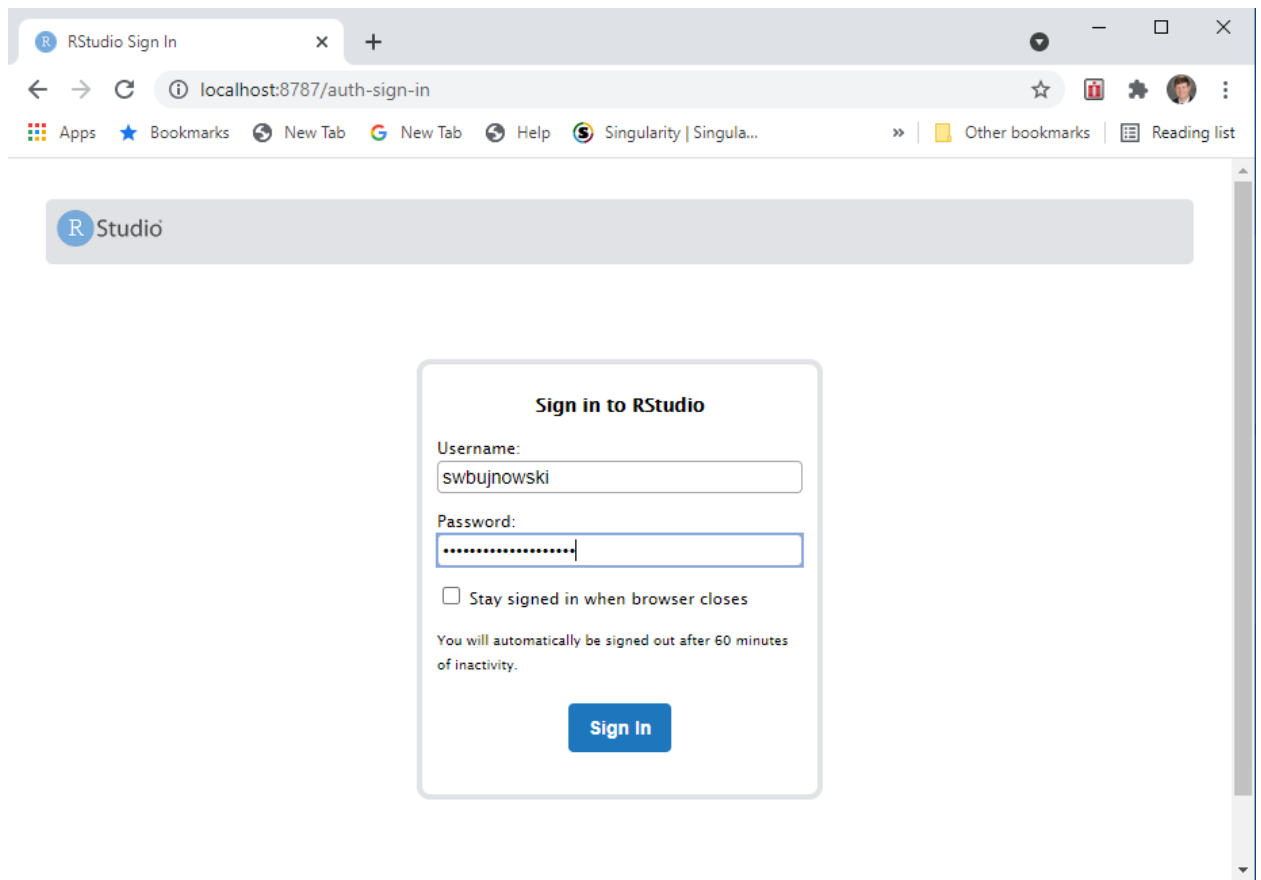
6. On your workstation, open a new terminal using PuTTY, using the information from bpeek <jobid> output.
 1. Your Host Name in the main PuTTY Configuration window should be <username>@seadragon



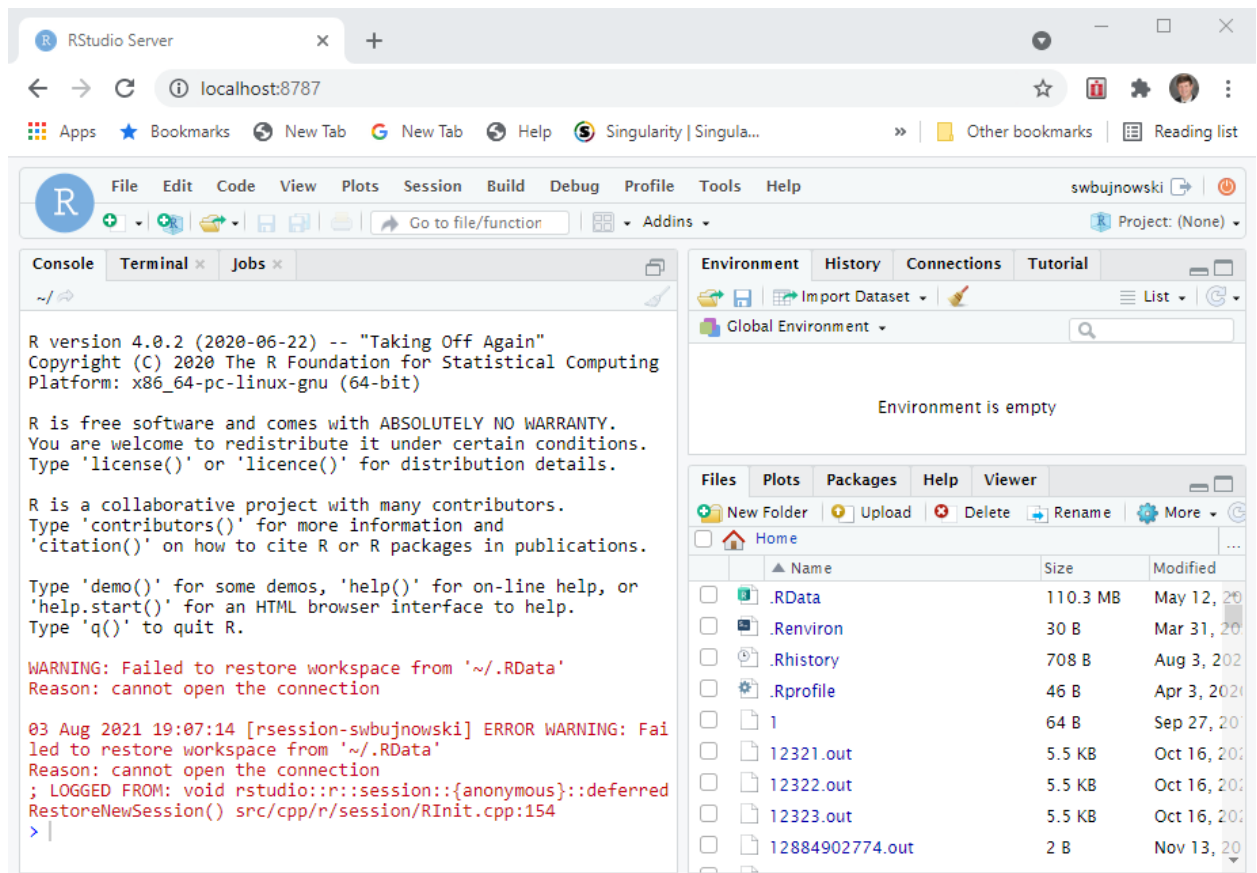
2. Click the “Tunnels” configuration item, type in a “Source port” of 8787, and enter the “Destination” as is given in the output of the “bpeek” command. When you have completed typing in this info click “Add”.



3. While still in this window, click “Open”, and enter your password when prompted to do so. Once logged in, leave this window open! You will not be inputting anything further using this window, but it needs to remain open so that X Windows can tunnel and forward results from Seadragon to your Windows PC.
7. Open a new web browser window on your Windows PC, type “localhost:8787” in the URL, and use the user and password information as given in the earlier “bpeek” command.



8. Click “Sign In”, and you are now in the RStudio GUI!



9. Exit RStudio:

- Exit RStudio by pressing the orange button in the upper right and close the web browser window by clicking "X" in the upper right-hand corner.
- Run the `kill <jobid>` command on the login terminal that you used to submit your job.
- Exit out of the terminal session that you created to set up the tunneling.

Install R packages inside container

Installing R packages inside the RStudio container is challenge, since the container has the Ubuntu operating systems, while Seadragon has RedHat 7 OS.

A practical approach is to install R packages into a container OS personal R library. When using RStudio container, set this library path inside RStudio console.

Using RStudio 4.0.2 as an example to install packages inside container:

1. [ryao@ldragon2 ~]\$ module load rstudio-sing/4.0.2
2. Create a new directory to put personal R library in Ubuntu OS
[ryao@ldragon2 ~]\$ mkdir -p /home/ryao/R/ubuntu/4.0.2
3. Install R packages by launch RStudio image on Seadragon login node
[ryao@ldragon2 ~]\$ singularity exec --bind /rsrch3 \ /risapps/singularity/repo/RStudio/4.0.2/rstudio_4.0.2.sif R

4. Set libPaths() in R

First, check the current libPaths(), see similar to this

```
> .libPaths()
[1] "/usr/local/lib/R/site-library"
[2] "/usr/local/lib/R/library"
[3] "/home/ryao/R/x86_64-pc-linux-gnu-library/4.0"
```

Next, set the libPaths to the personal library created in step 1.

```
> .libPaths("/home/ryao/R/ubuntu/4.0.2")
```

Check libPaths() again

```
> .libPaths()
[1] "/home/ryao/R/ubuntu/4.0.2"      "/usr/local/lib/R/site-library"
[3] "/usr/local/lib/R/library"
```

5. Installed the packages:

```
a) install BiocManager
> if (!requireNamespace("BiocManager", quietly =
TRUE))install.packages("BiocManager")
BiocManager::install(version = "3.12")

b) After BiocManager is installed successfully, install Bioconductor
packages as needed
> library(BiocManager)
Biocconductor version 3.12 (BiocManager 1.30.10),?BiocManager::install
for help
> BiocManger::install("AnnotationDbi")
...
Install the package and its dependencies

c) Install CRAN based packages, for example:
> install.packages("ggplot2")
```

.....
Install the package and its dependencies

Now, use the library in RStudio container:

1. Repeat the process to launch RStudio container, using rstudio 4.0.2 as an example,

ssh -Y seadragon

```
[ryao@ldragon2 ~]$ module load rstudio-sing/4.0.2
```

--- copy /risapps/singularity/repo/RStudio/4.0.2/rserver_4.0.2.lsf
to a directory under \$HOME, for example, in \$HOME/rstudio4.0.2

```
[ryao@ldragon2 rstudio4.0.2]$ bsub < rstudio4.0.2.lsf ]
Job <144577> is submitted to queue <short>.
<<ssh X11 forwarding job>>
<<Waiting for dispatch ...>>
[ryao@ldragon2 rstudio4.0.2]$ bpeek 144577 ]
<< output from stdout >>
1. SSH tunnel from your workstation using the following command:

    ssh -N -L 8787:cdragon296.cm.cluster:54451 ryao@seadragon

    and point your web browser to http://localhost:8787

2. log in to RStudio Server using the following credentials:

    user: ryao
    password: 21HmSjlxEZnlGUSSqyOP

When done using RStudio Server, terminate the job by:

1. Exit the RStudio Session ("power" button in the top right corner of the RStudio window)
2. Issue the following command on the login node:

    bkill 144577

The following have been reloaded with a version change:
1) gcc/7.2.0 => gcc/7.1.0

singularity exec --bind /rsrch3 --bind /home/ryao/logs/rstudio/RServer:/var/run/rstudio-server /risapp
s/singularity/repo/RStudio/4.0.2/rstudio_4.0.2.sif rserver --www-port 54451 --auth-none=0 --auth-pam-h
elper-path=pam-helper
```

2. On workstation terminal, Mac OS terminal or Putty, Cgywin on Windows 10

```
MDACM0CL5029737:~ ryao$ ssh -N -L 8787:cdragon296.cm.cluster:54451 ryao@seadragon
```

3. Open web browser to http://localhost:8787, login with the credentials from bpeek 144577;


The key is to set the personal library path to where the R packages are installed in the container:

```
.libPaths("/home/ryao/R/ubuntu/4.0.2")
```

Now, you can load the R packages that are installed, e.g. library(AnnotationDbi)

← → 🏠 🔒 localhost:8787

⚙️ Most Visited 🌐 Getting Started

 File Edit Code View Plots Session Build Debug Profile Tools Help

+ 🔍 📁 📄 🖨️ ➡️ Go to file/function 🧩 Addins

Console Terminal x Jobs x

~/ 🔍

```
R version 4.0.2 (2020-06-22) -- "Taking Off Again"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]

> .libPaths()
[1] "/home/ryao/R/x86_64-pc-linux-gnu-library/4.0"
[2] "/usr/local/lib/R/site-library"
[3] "/usr/local/lib/R/library"
> .libPaths("/home/ryao/R/ubuntu/4.0.2")
> .libPaths()
[1] "/home/ryao/R/ubuntu/4.0.2"      "/usr/local/lib/R/site-library"
[3] "/usr/local/lib/R/library"
> library(AnnotationDbi)
Loading required package: stats4
Loading required package: BiocGenerics
Loading required package: parallel
```