RESim Remote Access Guide

Configuring RESim servers and clients

August 26, 2022

1 Introduction

This guide provides instructions for accessing RESim remotely on a RESim server

These instructions assume your remote RESim server is bladet9. Adjust as needed based on the server assigned to you.

2 Client setup

Perform these steps on the computer that you will use to access RESim. These instructions assume a Linux host upon which IDA Pro is installed, e.g., a VM.

Use `ssh-keygen` to create an ssh key pair, using the `-t ed25519` option to avoid problems with ssh clients that reject sha1 keys by default. Send the public key to mfthomps@nps.edu along with a requested user ID. You will then receive an assigned server name and IP address. After receiving the reply, do the following:

- Add the assigned blade and IP address to your /etc/hosts:

```
10.20.200.159 bladet9
```

- Set up your /.ssh/config file to identify the gateway and RESim server. An example assuming your assigned RESim server is bladet9. Replace that, and the ProxyCommand IP with those assigned to you. Also replace the IdentifyFile with your own, and replace both User fields with your assigned login ID.:

```
Host cgc-gw
   IdentityFile        ~/.ssh/id_rsa
   User                mfthomps
   ServerAliveInterval 1
   ServerAliveCountMax 60
   TCPKeepAlive        yes
   HostName            205.155.65.172

Host bladet9
  HostName             bladet9
  User                 mike
  IdentityFile         ~/.ssh/id_rsa
  ProxyCommand         ssh -q cgc-gw nc 10.20.200.159 22
```

- Create an ssh agent on your local Linux. E.g., source this script (using your ssh id file):

```
eval ‘ssh-agent‘
ssh-add ~/.ssh/id_rsa
```

- You should now be able to ssh to the RESim server, e.g.,

```
ssh -Y bladet9
```

(Using -X seems to eventually time out because of temporary permissions?)

2.1 IDA

See the `RESim-UsersGuide.pdf` for information on installing and configuring IDA for use with RESim.

Clone the RESim repo onto the computer from which you'll run IDA (need this for RESim IDA Python plugins.) Set the RESIM_DIR, RESIM_IDA_DIR and IDA_DIR per the guide.

When starting IDA with the `runIda.sh` command per the guide, you will provide the name of your blade server as the final argument, e.g.,,

```
runIda.sh foo bladet9
```

Be sure you have an ssh agent running in the shell before using the runIda.sh command.

The runIda.sh command will create a tunnel to your blade server. You should be able to see this tunnel with the `ps aux | grep ssh` command. It will look something like:

```
ssh -fN -L 9123:localhost:9123 -oStrictHOstKeyChecking=no -oUserKnownHostsFile=/dev/null bladet9
```

3 Configure RESim Server

See section 6 if this RESim server has not yet been configured.

Steps below are implemented in RESim/simics/setup/config-resim-user.sh The following steps are taken for each user on the RESim server.

- Add to your .bashrc:

```
export RESIM_DIR=~/git/RESim
export SIMDIR=/mnt/simics/simics-4.8/simics-4.8.170
```

- Add to your .profile:

```
export PATH=$RESIM_DIR/simics/bin:$PATH
```

- Start a new shell to inherit that variable:

```
bash -l
```

- Use git to clone the RESim repo on the RESim server, e.g.,

```
mkdir ~/git
cd ~/git
git clone https://github.com/mfthomps/RESim.git
```

- Create a "workspace" directory, and cd to it. Then initialize it as a Simics workspace:

```
resim-ws.sh
```

- Copy the files in `git/RESim/simics/workspace` to your workspace, and follow the README instructions.

4 Workspaces

The steps above are intended to allow you to run tests using the CADET01 service. For each new project, create and initialize a new workspace. Simply create a directory with an informative name, cd to it and run the workspace or project setup command as described above.

5 Simics 5 procedures

This section addresses bladet10 which has both Simics 4.8 and Simics 5. When creating a new workspace, initialize it with:

```
resim-ws.sh
```

Two scripts in /usr/bin are used to control the license server. Use simics5-license.sh and simics4-license.sh to set the server for the version of Simics that you wish to run. Those scripts need only be run when changing versions.

6 System Setup

These steps are only required once for each new server. **Steps below are implemented in RESim/simics/setup/config-resim-server.sh**

- Add the following to the server /etc/hosts:

```
10.20.200.41 webproxy
```

- Confirm the /etc/apt/sources.list refers to the proper mirror, e.g.,

```
deb http://us.archive.ubuntu.com/ubuntu trusty universe
deb http://us.archive.ubuntu.com/ubuntu trusty main restricted
deb http://us.archive.ubuntu.com/ubuntu trusty-updates main restricted
```

- Create a mount point and add entry to the /etc/fstab:

```
sudo mkdir /mnt/re_images
sudo chmod a+rwx /mnt/re_images/
Add to /etc/fstab:  webproxy:/ubuntu_img /mnt/re_images nfs4 auto 0 0
```

- Create link to shared images:

```
sudo mkdir /eems_images
cd /eems_images
sudo ln -s /mnt/re_images ubuntu_img
```

- Install python-magic from gz file: pip install ¡path¿

```
sudo pip install /mnt/re_images/python_pkgs/python-magic-0.4.15.tar.gz
```

- Install xterm

```
apt-get install xterm
```

- Install git

```
apt-get install git
```

7 Configure Simics licenses

These steps are automated in the RESim/setup/config-simics.sh script.

- Get the Simics license server running (name the license file that matches your ethernet MAC address:

```
./simics-gui -license-file /mnt/simics/simics-4.8.75/licenses/24B6FDF7BB64.lic
```

- Then quit. Use

```
ps aux | grep lmgrd
```

  to confirm

- Install the vmx kernel module (Simics VMP)

```
bin/vmp-kernel-install
(follow instructions to enable on reboot)
```