Instructions to log into the RedHawk machine

Step-by-step guide to login and clone the Git repository on RedHawk machine

Method 1:

Note: Following method gives the git-lfs space error. The git-lfs space is in RedHawk machine not on pslogin. Also, to clone repo on pslogin is not recommended.

- Step 1: How to do pslogin:
 - ° ssh to pslogin: "ssh username@pslogin.slac.stanford.edu"
 - After pslogin user can clone the git project repository in pslogin space (This space will be shared between pslogin and RedHawk machine)
- Step 2: How to shh to RedHawk machine:
 - ssh to RedHawk from "pslogin": "ssh psstuart"
 - This step will connect the user with RedHawk machine
 - ° Once the user logs into RedHawk machine, the user can see the same project repository cloned during the first step.

Method 2:

- Step 1: ssh to pslogin: "ssh username@pslogin.slac.stanford.edu" or "ssh pslogin -Y"
- Step 2: ssh to RedHawk from "pslogin": "ssh psstuart"
 - This step will connect the user with RedHawk machine
- Step 3: Once the user logs into RedHawk machine, user can clone the git repo by typing following command.
 ° git clone --recursive git@github.com:slaclab/user_repo.git

How to load the firmware programming files into RedHawk machine

- User cannot build the firmware on RedHawk machine because Vivado doesn't support the RedHawk machine distro.
 - User can just build it on Ubuntu then use the python script to program the generated .MCS file. To do that pleaseAD follow following steps:

 Step 1: Go to Ubuntu server and build the firmware. For more information on how to build the firmware of your project please follow the link given in Related articles.
 - Step 2: Once the FPGA programming files are ready. Copy those files to RedHawk machine by using following command.
 - scp -r /u/gu/adave/Project/snl-trans-FES/firmware/targets/SnlTransFesKcu1500/images pslogin:/cds/home/a/adave/Projects/snltrans-FES/firmware/targets/SnlTransFesKcu1500
 - Here the FPGA image files from rdsrv310 are being copied and pasted on RedHawk machine's mount-point dir.

How to reprogram the PCIe firmware via Rogue software

- Step 1: Go to software folder by typing following command.
 ° cd user_project/software
- Step 2: To setup the rogue environment please follow the following command.
 - ° pskube-minion03 ~\$ bash
 - bash-4.2\$ source ~ruckman/anaconda3/etc/profile.d/conda.sh bash-4.2\$ conda activate rogue_v5.15.3 (rogue_v5.15.3) bash-4.2\$
 - This will activate the conda rogue environment
- Step 3: Run the PCIe firmware update script

o \$ python scripts/updatePcieFpga.py --path <PATH_TO_IMAGE_DIR>

where <PATH_TO_IMAGE_DIR> is path to image directory (example: ../firmware/targets/SnIBesKcu1500/images/)

Select image number when following message is shown:

```
<sup>o</sup> Rogue/pyrogue version v5.10.0. https://github.com/slaclab/rogue
  Basedir = /afs/slac.stanford.edu/g/controls/development/users/ktkim/Project/EdgeML/snl-bes-dev
  /firmware/submodules/axi-pcie-core/scripts
  Start: Started zmqServer on ports 9099-9101
  *****
  Current Firmware Loaded on the PCIe card:
  *****
  Path
             = PcieTop.AxiPcieCore.AxiVersion
  FwVersion = 0x1010000
UpTime = 0:56:11
             = 0x35cdbac182394d890796f4114e5a85f7e9a87966
  GitHash
  XilinxDnaId = 0x4002000101181bc40480a145
 FwTarget = SnlBesKcu1500
BuildEnv = Vivado v2021.1
  BuildServer = rdsrv317 (Ubuntu 20.04.3 LTS)
  BuildDate = Thu 09 Dec 2021 12:33:26 PM PST
Builder = ruckman
  ******
  0 : ../../assets/SnlBesKcu1500/v1.1.0/SnlBesKcu1500-0x01010000-20211209123326-ruckman-35cdbac
  1 : Exit
  Enter image to program into the PCIe card's PROM: <IMAGE NUMBER>
```

- Step 4: Reboot the RedHawk machine by typing following command
 ^o sudo reboot
- Step 5: ssh to RedHawk from "pslogin": "ssh psstuart"
- Step 6: Reload the driver by following steps:

```
° $ sudo /ul/aes-stream-drivers/data_dev/driver/load
 $ cat /proc/datadev_0
        ----- Axi Version -----
      Firmware Version : 0x3
           ScratchPad : 0x0
         Up Time Count : 3292321
            Device ID : 0x0
             DNA Value : 0x0000000430a12684ec7d10100008004
          Build String : DrpTDetSemi: Vivado v2018.2, pslab01.slac.stanford.edu (x86_64), Built Wed
 Aug 14 22:34:06 PDT 2019 by weaver
 $ cat /u1/aes-stream-drivers/data_dev/driver/load
  #!/bin/sh
  # set up the proper kernel build environment)
 cd /lib/modules/$(uname -r)/build; ./ccur-config -c -n
  # build kernel loadable module
 cd /ul/aes-stream-drivers/data_dev/driver; make clean; make
  # add new driver
  /sbin/insmod /u1/aes-stream-drivers/data_dev/driver/datadev.ko cfgSize=0x200000 cfgRxCount=256
 cfgTxCount=16 || exit 1
  # give appropriate group/permissions
```

```
chmod 666 /dev/datadev*
```

How to install the Rogue With Anaconda

- Step 1: Go to https://www.anaconda.com/download to get the latest version of anaconda. Example steps for installing anaconda are included below:
 - ° \$ wget https://repo.anaconda.com/archive/Anaconda3-2020.07-Linux-x86_64.sh \$ bash Anaconda3-2020.07-Linux-x86_64.sh
- Step 2: Use the following command to add anaconda to your environment. This can be added to your .bash_profile.
 \$ source /path/to/my/anaconda3/etc/profile.d/conda.sh

Creating A Rogue Environment

- Step 3: The next step is to create an anaconda environment which includes the Rogue package.
 - \$ conda create -n rogue_5.15.3 -c tidair-packages -c conda-forge -c pydm-tag -c tidair-tag rogue=v5.15.3 h5py
 If you already have an anaconda environment that you would like to install Rogue into:
 - \$ conda install -c tidair-tag -c tidair-packages -c conda-forge rogue
 - The above commands will install the latest version of Rogue from the master branch.

The detailed instruction on installing Rogue with Anaconda can be found here.

How to run the software

- Step 1: Go to software folder by typing following command. ° cd user_project/software
- Step 2: To setup the rogue environment please follow the following command.

```
° pskube-minion03 ~$ bash
  bash-4.2$ source ~ruckman/anaconda3/etc/profile.d/conda.sh
  bash-4.2$ conda activate rogue_v5.15.3
(rogue_v5.15.3) bash-4.2$
```

- $^{\circ}~$ This will activate the conda rogue enviroment
- Step 3: open the scripts/devGui.py file and comment the default port number.

65	parser.add_argument(
66		"serverPort",				
67		type	=	int,		
68		required	=	False,		
69		#default	-	= 9099 ,		
70		help	=	"Zeromq	server	port",
71)					

- Step 4: Go back to the software folder and run the devGui script file. ° \$ python scripts/devGui.py --guiType None --h5Path notebooks/TestDataCheck.h5

1

Related articles

How to build firmware for the project: https://github.com/slaclab/snl-bes-dev

The detailed instruction on installing Rogue with Anaconda: https://slaclab.github.io/rogue/installing/anaconda.html

• Instructions to log into the RedHawk machine