

# Notes for the Test Beam Setup at FermiLab

Martin

March 1, 2018

## 1 Computing Setup

### 1.1 Network Topology

The non-routable network 192.168.100.0/24 spans the control room and the enclosures. It is masqueraded (NAT'ed) via the 192.168.100.99 gateway machine, so machines on the network are able to get to the general internet. For a manual configuration, set up the network as follows:

IP address	192.168.100.X	
netmask	255.255.255.0	
broadcast	192.168.100.255	
gateway	192.168.100.99	
nameserver	131.225.0.254	FermiLab's Name server
ntp server	131.225.8.126/127	FermiLab's time server
ntp server	131.225.17.126/127	more time servers

These are the current devices connected to that network:

Name	IP address	Location	Role
sphenxdaq	192.168.100.1	rack room	Main Data Acquisition Machine
ts0	192.168.100.2	rack room	Terminal server
iocondev13	192.168.100.6	rack room	GTM controller
Wiener MPOD	192.168.100.12	enclosure	Wiener power supply
spmdaq	192.168.100.18	enclosure	Craig's dual SiPM test DAQ machine
va095	192.168.100.57	rack room	sPHENIX DAQ machine
hcalcam	192.168.100.80	enclosure	data logging camera pi (hcal)
emcalcam	192.168.100.81	enclosure	data logging camera pi (emcal)
eicdaq.fnal	192.168.100.91	MT2.6A	EIC DAQ machine
ftbfnl01local	192.168.100.99	on table	FTBFBNL01 machine as seen from our network
SiPM Control	192.168.100.110	enclosure	Control board emcal
SiPM Control	192.168.100.120	enclosure	Control board hcal

## 1.2 Accessing the Network from Outside

We made it so that we can log in from “outside” (that includes the general FermiLab network) to our DAQ machines. While Brookhaven uses ssh keys to authenticate users, FermiLab uses Kerberos to do the same. The downside for us is that everyone needs to use the kerberos software on his or her computer, but most of the time this is already installed.

Use the `kinit <username>` command to obtain a kerberos token. Remember that in most cases, especially on Windows, your username will not be the same as your login name at FermiLab. If successful, the `klist` command should show a *ticket granting ticket*:

```
$ klist
Ticket cache: KCM:501
Default principal: purschke@FNAL.GOV

Valid starting          Expires                Service principal
01/18/2017 13:36:14    01/19/2017 15:36:14  krbtgt/FNAL.GOV@FNAL.GOV
    renew until 01/25/2017 13:36:14
01/18/2017 13:36:36    01/19/2017 15:36:14  host/ftbfnl01.fnal.gov@FNAL.GOV
    renew until 01/25/2017 13:36:14
```

Then you can login to our gateway machine (which is seen as 192.168.100.99 from the internal network) as user “ftbf\_user” to ftbfnl01.fnal.gov, as in

```
ssh -l ftbf_user ftbfnl01.fnal.gov
```

Your FNAL account name must be enabled in order for you to login, which we can take care of ourselves. That ftbfbnl01 machine is one of ours, which got a new system disk where the computing division installed the standard Fermi SL6.

### 1.3 Making a reasonable alias for logging in

The ftbfbnl01 gateway acts, in many aspects, like the rssh gateways at BNL. Consider setting up a script or an alias to log in consistently with the same tunnels (so you can bookmark some tunneled pages).

The services you may want to establish tunnels to are the Elog and maybe the cameras.

Try, as a suggestion

```
ssh -l ftbf_user ftbfbnl01.fnal.gov \  
-L 17815:localhost:7815 \  
-L 10080:192.168.100.80:8081 \  
-L 10081:192.168.100.81:8081 \  
-L 10082:192.168.100.91:8081
```

This allows you to see, in your local web browser

- the logbook as `http://localhost:17815`
- the hcal camera as `http://localhost:10080`.
- the emcal camera as `http://localhost:10081`.
- the webcam at the EIC setup in 2.6A `http://localhost:10082`.

To make my life easier, I added to my `$HOME/.ssh/config`

```
Host va095  
  User phnxrc  
  ProxyCommand ssh ftbf_user@ftbfbnl01.fnal.gov nc -w7200 %h %p  
  
Host eicdaq_fnal  
  User eic  
  ProxyCommand ssh ftbf_user@ftbfbnl01.fnal.gov nc -w7200 %h %p
```

so I can just type “ssh va095” and am logged in as user phnxrc on our DAQ machine va095. Similar for eicdaq\_fnal.

## 1.4 Copying data from the daq machines to RCF

The obvious gateway into RCF is to scp files to the proper destination on rftpexp.rhic.bnl.gov. Keep in mind that you need an ssh agent running (typically at the machine where the connection originates from, usually your laptop), which has *your* RCF key.

Here is how I copy:

```
rsync -av /data0/phnxrc/fnal/ purschke@rftpexp.rhic.bnl.gov:  
/sphenix/data/data03/phnxreco/sphenix/t1044/fnal/
```

(that's all in one line.)

rsync is all about placing the right trailing slashes (or not), so the trailing slashes you see here are important.

Some of us experienced a lot of trouble with login in, which is not really different from our day-to-day interaction with the RCF.

Some pointers:

- you *must* run an ssh-agent on your laptop. Macs all do, Linux machines all do, on Windows machines you need to manually start one, either ssh-agent if you use cygwin, or “pagent” (comes as a putty add-on) if you are using putty.
- `ssh-add -L` is your friend. This command lists all “identities” that your agent has, and is the prime debugging tool if you think you should be able to login but cannot.
- verify that you can log in to RCF directly from your laptop. Once logged in, list the keys with `ssh-add -L`. Log in to the ftbfbnl01 machine and issue the `ssh-add -L` command there. It should show the same keys. If not, if it lists no identities, or displays a message that it cannot see an agent, your chain of ssh-agent forwards is broken. The agent connection gets forwarded through all the ssh hops from one machine to the next, which is set up on all our machines. If this is the case, the problem is virtually guaranteed to be with your laptop.
- have a look at [https://www.phenix.bnl.gov/WWW/offline/wikioff/index.php/Ssh\\_access\\_to\\_internal\\_machines\\_the\\_easy\\_way](https://www.phenix.bnl.gov/WWW/offline/wikioff/index.php/Ssh_access_to_internal_machines_the_easy_way), which explains most of the stuff that you should put into your `~/.ssh/config` file.