ALS-U EPICS Environment

Jeong Han Lee
jeonglee@lbl.gov

This document covers only the Rocky Linux distribution, however, it may work with Redhat variants as well. If one would like to install them to Debian, one can install packages correctly (i.e., package names are slightly different), then all other Makefile procedures remain the same.

Packages

The following repository configuration I did before installing any packages and asking the unix team.

```
Unset
dnf -y install dnf-plugins-core
dnf -y config-manager --set-enabled powertools
dnf -y install "epel-release"
```

The following packages are not mandatory at ALS, since they already were installed as default. However, if you use your own Rocky Linux, please install them first before installing others. In the last chapter, we will show you more convenient packages for the development environment.

```
Unset
dnf -y install git sudo
```

Here are some packages are required for the ALS-U EPICS environment

```
Unset
dnf -y install \\
tree which autoconf libtool automake re2c graphviz flex-devel patch
readline-devel libXt-devel libXp-devel libXmu-devel libXpm-devel motif-devel
gcc-c++ ncurses-devel perl-devel net-snmp net-snmp-utils net-snmp-devel
libzip-devel libusb-devel platform-python-devel boost-devel pcre-devel
libcurl-devel libxml2-devel hdf5-devel netcdf-devel libtiff-devel
libjpeg-turbo-devel libevent-devel libpng-devel libusbx-devel systemd-devel
```

```
libtirpc-devel libtirpc rpcgen re2c libusb-devel libusb python3-devel cmake
libssh2-devel libssh2
```

Some packages I cannot install due to proxy configuration. I asked the Unix team to set up. For example,

```
Unset
net-snmp net-snmp-utils net-snmp-devel boost-devel hdf5-devel
```

ULDAQ Requirements for EPICS measComp module

For the ALS-U Environment, we reserve the vendor specific library within the `INSTALL_LOCATION/vendor` Under `vendor` path, `include` and `lib` will be used for all not-system libraries and headers. We use the `INSTALL_LOCATION` as `/usr/local/epics/alsu.`

```
Unset
$ git clone https://github.com/jeonghanlee/uldaq-env
$ cd uldaq-env
uldaq-env (master)$ echo "INSTALL_LOCATION=/usr/local/epics/alsu/vendor" >
configure/CONFIG_SITE.local
uldaq-env (master)$ make init
uldaq-env (master)$ make conf
uldaq-env (master)$ make build
uldaq-env (master)$ make install
uldaq-env (master)$ make exist
tree -aL 1 /usr/local/epics/alsu/vendor
/usr/local/epics/alsu/vendor
├── include
└── lib
```

Python packages

Rocky 8

```
Unset
$ sudo python3 -m pip install numpy nose2
```

Rocky 9

```
Unset
$ sudo python3 -m pip install numpy==1.19.5 nose2
```

EPICS Environment

There are three variables we should define.

- `EPICS_TS_NTP_INET` : I think we don't need to set up this if we use any Linux host, however, it doesn't hurt. If you are not in the ALS network, you can use `time.google.com`, or any other NTP server around your location instead of tic.lbl.gov
- `VENDOR_ULDAQ_PATH` : This is what one see in the above
- `INSTALL_LOCATION` : This is where the EPICS will be. One must have the write permission on that path. We use the `INSTALL_LOCATION` as `/usr/local/epics/alsu.` If you don't define it, it will use `${HOME}/epics` as a default location. And only if your path does not contain `epics`, the `epics` path will be added to your path as well.

We are still in the release candidate phase, so please use the master branch with the latest commit.

```
Unset
$ git clone https://github.com/jeonghanlee/EPICS-env.git
$ cd EPICS-env
EPICS-env (master)$ echo "EPICS_TS_NTP_INET=tic.lbl.gov" >
configure/RELEASE.local
```

```
EPICS-env (master)$ echo "VENDOR_ULDAQ_PATH=/usr/local/epics/alsu/vendor" >>
configure/RELEASE.local
EPICS-env (master)$ echo "INSTALL_LOCATION=/usr/local/epics/alsu" >
configure/CONFIG_SITE.local
```

```
Unset
EPICS-env (master)$ make init
EPICS-env (master)$ make patch
EPICS-env (master)$ make conf
EPICS-env (master)$ make build
EPICS-env (master)$ make install
EPICS-env (master)$ make symlinks
```

```
Unset
$ source /usr/local/epics/alsu/1.1.0/rocky-8.8/7.0.7/setEpicsEnv.bash

$ softIoc
$ softIocPVA
$ softIocPVX
$ pvget -h
$ pvxget -h
$ caget -h
```

The SNMP mib files for Rocky 8/9

To troubleshoot any SNMP connection easily, I asked Kuldeep to review the following external files for the SNMP MIBs files. He approved the installation, so I installed it on `appdev`, which is the development server at ALS.

```
Unset
$ git clone https://github.com/jeonghanlee/snmp-mibs-downloader-env
$ cd snmp-mibs-downloader-env
$ snmp-mibs-downloader-env (master)$ make init
$ snmp-mibs-downloader-env (master)$ make install
$ snmp-mibs-downloader-env (master)$ make get
```

Additional Packages

Here are the list of additional packages which we believe could help developers to navigate their IOC development and deployment process.

```
Unset
dnf -y install vim bash-completion
```