**Cross compile auto conf and auto make**

**And using example of Joe editor**

When we have application in Linux where the make should be generate using configure

We can recognize it by doing ls and see files :

configure.ac

makefile.am

Then we know that the project was intend to be configure as the configure first to generate the makefiles

If we run the ./configure without parameters, it will build a the make file to the current architecture.

We want to configure and build for cross compile.

We want to provide the compiler and the host and produce the executable to run on our target.

This can be done as following.

The best is to create source script with three variables

1. CC

2. CXX

3. PATH

As before, the basic for the entire cross compilation is to have a cross compiler tool chain.

In my cross compiling with crosstool-ng I have created a compiler for arm with hardware floating points:

# this is our activate script

export PATH=$PATH:/home/adsteam/x-tools/arm-unknown-linux-gnueabihf/bin

export CC=arm-unknown-linux-gnueabihf-gcc

export CXX=arm-unknown-linux-gnueabihf-g++

We want to cross compile joe editor , my favorite editor.

So download the sources from joe own editor web site

<http://joe-editor.sourceforge.net/>

<https://sourceforge.net/projects/joe-editor/files/JOE%20sources/joe-4.4/>

Once download , we create the above script , let’s call it ACTIVATE.sh

To run configure for cross compilation we do:

./configure --host=arm-unknown-linux-gnueabihf --target=arm-linux -prefix=/home/username/joe\_editor/joe-4.4

Some notes:

The host is actually the compiler name **without** the – and the end and of course without the compiler name

The target is arm-linux

And the prefix is where the make install will install our executables.

When the configure ended , we run:

make

And when finished we run:

make install

we can do file on the target:

file joe/joe:

joe/joe: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked (uses shared libs), for GNU/Linux 4.10.8, not stripped

now we just need to copy it to our target and run it.