LinuxDNA ICC Kernel Compile HowTo

By: C. Tyler McAdams of LinuxDNA.com
April 12th, 2010

Configuration, make and installation of the LinuxDNA kernel patch has become much easier for the latest patch sets. This guide should step you through all of the necessary requirements needed to successfully compile the new 2.6.32 (and up) 32bit or 64bit patches available on the linuxdna.com mirror.

Step one - ICC Installation and Configuration:

The first thing needed is to install the Intel C/C++ Compiler (ICC). The compiler is completely free for individuals to download and use for non-commercial type use. You will need to register for the serial number and also the license file that is put in: /opt/intel/licenses *Please note if you are using Gentoo your compiler will not function until either a non-commercial or commercial license file has been added to this directory and you have run: source /etc/profile*. Distributions such as Ubuntu and Gentoo Linux have the means to install ICC directly from the repositories. Using the distribution's built-in method is the easiest way to install the compiler but sometimes the compiler is slightly out of date and will not work with the patch that you want to apply. Please check the linuxdna.com mirror to see which compilers have been tested with the patch that you want to use. The tested versions will be stated in the text above the link to download the given patch. If you are using Gentoo Linux and want to use the latest 64bit kernel you can use the following ebuild which should give you the latest available ICC (11.1.069):

http://www.linuxdna.com/dev-lang_icc-11.1.069.tar.bz2

The following link can be used to get the latest version ICC in a stand-alone installer directly from Intel:

http://www.intel.com/cd/software/products/asmo-na/eng/download/eval/219771.htm

The ICC installer comes with a shell script: installer.sh that walks you through installing the compiler manually. Please note that you will need to install both the 32bit and 64bit components of the ICC compiler in order to compile our latest 64bit patches. Red Hat type Linux distros that use SElinux will complain during install that ICC can not be installed due to a SELinux violation. It is ok to disable SElinux temporarily and then re-enable it after you finish installing ICC. You can use the following command to do so:

setenforce 0

After you have finished installation you can simply re-enable SElinux with the following command and ICC will work with no errors:

setenforce 1

Once you are ready to use ICC you must first source the compiler for the type of compiling you wish to do. There are three choices depending on the architecture:

For x86_64 or x64 (64bit) architectures:

source /opt/intel/Compiler/11.1/069/bin/iccvars.sh intel64

For x86 (32bit) architectures:

source /opt/intel/Compiler/11.1/069/bin/iccvars.sh ia32

For Itanium IA64 (64bit) architectures:

source /opt/intel/Compiler/11.1/069/bin/iccvars.sh ia64

Please note that you will need to source the intel64 environment in order to successfully compile our latest patches. To compile for the Itanium architecture you will need to make sure you downloaded the IA64 version of ICC from Intel. Also note that the version ICC you source is chosen by it's directory structure. If you have multiple versions installed or an older version of the compiler the directory structure will differ from above depending on that compiler. For example if you have version 11.0.083 installed and wanted to use that particular version for a 64bit patch you would need to use the following command:

source /opt/intel/Compiler/11.0/083/bin/iccvars.sh intel64

If you wanted to use the compiler to build a 32bit kernel the command differs slightly:

source /opt/intel/Compiler/11.0/083/bin/iccvars.sh ia32

If you do not want to have to type out this command every time you use ICC you can add this line to your bash shell's .bashrc file. (Remember to change this line after installing a newer version to avoid problems however)

Once ICC has been sourced you can test to make sure that ICC is working with this command:

icc -v

You should get similar output:

[root@compaq ctylerm]# icc -v
Version 11.1

Step two – patching, configuring and making your kernel:

After ICC has been installed and sourced most of the work has been done. From here making your kernel is as simple as applying the linuxdna patch, configuring the kernel and executing make. To apply the linuxdna patch move the patch to your kernel source's top directory and use the patch command with the -p1 option:

patch -p1 < dna-2.6.33-intel64.patch

If you would like to revert the patch and apply a newer patch you can use these commands to do so:

patch -p1 -R < *old-patch*

patch -p1 < *new-patch*

In the past there was an intermediate script needed, known as the "wrapper", that needed to be used in order for ICC to work correctly with the kernel source. The latest patches have been designed to work without this wrapper to help cut down on confusion regarding the build process. Once you have configured your kernel (be it with *make old config* or manually through *make menuconfig*) building the kernel is as simple as issuing the make command:

make

Using the make command normally with any normal additions (such as *&& make modules_install*) will make a fully gpl 2.0 compatible kernel that can them be copied and moved to any system you wish – granted it's the same distro linux. There is also the option to use ICC's built in LD and AR linking however:

make AR=xiar LD=xild

This can improve performance but if you choose to reuse the kernel on other systems please note you will need to install either ICC or the ICC redistributable libraries first in order for the kernel to function.

Conclusion

These basic steps work for any 64bit LinuxDNA patch 2.6.32.9 and above. If you are still having issues getting the kernel or ICC to work you can post your problem to the LinuxDNA Google Group for more technical support:

http://groups.google.com/group/linuxdna