



**Hewlett Packard
Enterprise**

ARCHER2 User Forum Update Software Environment

Harvey Richardson and Douglas Shanks
June 7, 2023

Introduction

- ARCHER2 upgrade in spring 2023
- This updates the following software
 - System Management software (CSM)
 - The OS on UANs and compute nodes (COS)
 - The Slingshot network software
 - The Cray Programming Environment



System Software updates

Hardware support (not relevant to ARCHER2 today):

- Support for new network hardware (Cassini NICs) and many updates for GPU support across system software and Slingshot.
- Low Noise Mode (relevant to GPU driver activity)

System software (CSM, COS & Slingshot)

- Updated versions of various system components
- Security fixes
- COS is now based on SLES 15 SP4, note that Python2 is no longer shipped as part of SLES and only Python3 is supported by the PE
- Many improvement to the scalability of system activities (node boot/reconfiguration etc.)
- Monitoring improvements for Power/CDUs

Slingshot

- New monitoring infrastructure (for administrators)
- Improved error detection, better resiliency to multi point faults and improved detection of marginal hardware issues.



Cray Programming Environment

- The CPE provides the compilers, MPI implementation, other libraries, modules system, compiler wrappers etc.
- Before the Upgrade we had the following versions:
 - CPE 21.04 (default)
 - CPE 21.09
 - CPE 22.04
 - Latest compilers: CCE 13.0.2; GNU 11.2.0; AOCC 3.2.0;
- After the upgrade:
 - CPE 22.12
 - Latest compilers: CCE 15.0.0; GNU 11.2.0 (12.2.0); AOCC 3.2.0;

New Website

- There is a new website with programming environment information: <https://cpe.ext.hpe.com>



Updates 21.04 to 22.04 (defaults)

- There were of course many fixes and updates between 21.04 and 22.04 previously
- You needed to load cpe/22.04 to get the latest environment we had
- **We will just mention one change that might have been missed...**
- Mixed compiler modules



Mixed Compiler Modules

- There are special ‘mixed’ compiler modules that can be used to load an *outside-of* PrgEnv compiler
 - This is compatible with the Lmod hierarchy and allows for example PrgEnv-cray along with GCC
 - Available for AOCC, AMD, CCE, GCC (check via `module av mixed`)
- As an example

```
% which gcc ; gcc --version  
/usr/bin/gcc  
gcc (SUSE Linux) 7.5.0
```

OS gcc

```
% module list PrgEnv
```

```
Currently Loaded Modules Matching: PrgEnv
```

```
 1) PrgEnv-cray/8.3.3
```

```
% module load gcc-mixed
```

```
% gcc --version  
gcc (GCC) 12.2.0 20220819 (Cray Inc.)
```

CPE default gcc

```
% cc --version  
Cray clang version 15.0.0 (324a8e7de6a18594c06a0ee5d8c0eda2109c6ac6)
```

Added Features and Fixes

- CCE 13.0.2 to 15.0.0
 - LLVM 15 base
 - General performance improvements for CPU and GPU
 - Complete support for AddressSanitizer & ThreadSanitizer for Fortran programs (CPU only)
 - archer OpenMP runtime data-race detection (link with `-larcher`, put library in `LD_LIBRARY_PATH`) (see <https://github.com/llvm/llvm-project/tree/main/openmp/tools/archer>)
 - OpenMP simd support on by default
 - OpenACC 2.6 and 3.0 Features
 - The Fortran option `-htcmalloc` is deprecated and planned to be removed in CCE 16.0.0
 - C++ applications built using CCE 13 or earlier may need to be recompiled
- Cray MPICH 8.1.15 To 8.1.23
 - Cray-MPICH compiler wrappers are now in the `PATH` when the module is loaded
 - MPI-IO Upgrade to support stripe sizes > 2GB
 - Fix for `MPI_Igather`, `MPI_Iscatter` and `MPI_Scatter` algorithms
 - Support Lustre group-lock when collective buffering is disabled
 - Updated Dynamic Process Management information in `mpi` man page

Added Features and Fixes

- MRNET
 - MRNet (Multicast/Reduction Network library) is a customizable, high-performance software infrastructure for building scalable tools and applications. It supports efficient multicast and data aggregation functionality using a tree of processes between the tool's front-end and back-ends.
- PMIx
 - OpenPMIx 4.2.2 supported through Slurm (--mpi=pmix)
An implementation of the PMIx standard, which enabling third-party MPI and OpenSHMEM implementations to learn about their parallel environment from the application launcher.
- MPIxlate (cray-mpixlate)
 - MPIxlate provides integration of OpenMPI apps with cray-mpich



Added Features and Fixes

- HPC Tools ([CPE 22.04 to 22.12](#))
 - gdb4hpc
 - Lots of bug fixes
 - Fix memory leaks and hang on exit when a back-end gdb has a fatal error
 - Add launch option --sbatch for submitting Slurm and --qsub for PALS batch scripts
 - **New “run”** command to re-launch applications, preserving breakpoints between launches
 - Many more updates
 - Abnormal Termination Processing (ATP)
 - Use llvm-addr2line to resolve missing line numbers
 - Set ZMQNet as default network, switch to MRNet with ATP_PROTO
 - Build and enable ATP Slurm plugin at module load time
 - Add support for Lightweight Corefile generation and utility
 - Add ATP_CORES_HOSTS to specify hosts on which to dump cores
 - valgrind4hpc
 - Monitor valgrind process for stop signals and start it again if it encounters any
 - **sanitizers4hpc ...**

Added Features and Fixes

- Slurm [21.08.8-2 to 22.05.8](#)
 - Old-style Heterogeneous jobs working as expected
 - Add switch/hpe_slingshot plugin
 - Add support for PMIx v4 and v5
 - YAML and JSON output now supported from various Slurm commands.
 - Minor memory leak fixes
 - Many other changes



Sanitizers for HPC

Use several tools to check program correctness at run-time for parallel applications



Perform dynamic analysis of parallel programs with sanitizers4hpc

- Sanitizers4hpc is a debugging tool to aid in the detection of memory leaks and errors in parallel applications
 - Static instrumentation at compile time via `-fsanitize=<sanitizer>`
 - Sanitizers are Address, Leak, Thread (<https://github.com/google/sanitizers>)
 - It aggregates any duplicate messages across ranks to help provide an understandable picture of program behaviour
- Sanitizers4hpc supports the Sanitizer libraries included with both the Cray CCE and the GNU GCC compilers
 - Cray Fortran only supports Address and Thread sanitizers
 - Note: Address Sanitizer and Thread Sanitizer cannot be used simultaneously
- More info:
 - `man sanitizers4hpc`
 - <https://cpe.ext.hpe.com/docs/debugging-tools/sanitizers4hpc.1.html>



Sanitizers for HPC

- Compile the application with **-fsanitize=<sanitizer>**, e.g.

```
cc -g -fsanitize=leak leak.c -o leak
```

- Load the module to access the sanitizers4hpc executable
- Get an interactive session via salloc

```
> salloc ...  
> sanitizers4hpc -l "-n4" -- ./a.out arg1 arg2
```

- **-l** to pass arguments to the system launcher (e.g., Slurm)
- The target binary and its arguments are listed after the double dash -
- More info and examples: **man sanitizers4hpc**



Sanitizers for HPC Example: Address sanitizer

```
1: program address
2:   implicit none
3:
4:   integer, dimension(10) :: array
5:
6:   array = 2
7:
8:   array(12) = 3
9:
10:  print *, array
11:
12: end program address
```

```
RANKS: <0>
AddressSanitizer: global-buffer-overflow on address 0x000000551fec at pc 0x00000041f856 bp 0x7fffd76e8750 sp 0x7fffd76e8748
WRITE of size 4 at 0x000000551fec thread T0
  #0 0x41f855 in address_ /mnt/lustre/a2fs-tdswork1/work/y02/y02/dshanks/sanitizers4hpc_f90/address.f90:8
  #1 0x15343b8e529c in __libc_start_main (/lib64/libc.so.6+0x3529c) (BuildId: c8417d767baccfad39b474e484d46947915cd8f)
  #2 0x41f429 in _start /home/abuild/rpmbuild/BUILD/glibc-2.31/csu/./sysdeps/x86_64/start.S:120

0x000000551fec is located 4 bytes to the right of global variable 'array' defined in '/mnt/lustre/a2fs-tdswork1/work/y02/y02/dshanks/sanitizers4hpc_f90/address.f90:4' (0x551fc0) of size 40
SUMMARY: AddressSanitizer: global-buffer-overflow /mnt/lustre/a2fs-tdswork1/work/y02/y02/dshanks/sanitizers4hpc_f90/address.f90:8 in address_
```

Thank you

