

Preparation for the Hackathon: Access to Cray KNL System

Hans-Hermann Frese, Thomas Steinke

ZIB

Outline

- On-line documentation
- Access to HLRN resources
- Ideas for the Hackathon

On-line Documentation Workshop, Cray TDS + HLRN

- <http://bit.ly/zib-knl-2017>
- <http://tinyurl.com/ZIBCrayTDSDoc>
 - ❖ Login, uploading a ssh public key
 - ❖ Access to programs managed by `modules` (environment)
 - ❖ (interactive) Batch jobs, KNL mode provisioning
 - ❖ Using Intel VTune, how to use VNC
- <http://www.hlrn.de>
 - ❖ HLRN on-line documentation including general TDS topics

Access to HLRN: Upload a Public SSH Key

- Step 1: Generate SSH key pair with passphrase on your local host (laptop)
localhost\$ ssh-keygen
- Step 2: Login to HLRN Service Portal
<https://zulassung.hlrn.de>
- Step 3: Select
Verwalten Ihrer Keys zum Einloggen auf den HLRN-Rechnern
- Step 4: Login using HLRN username and PIN (see letters)
- Step 5: Select
Upload eines neuen Schlüssels anfordern
- Step 6: Wait for an email message from HLRN Service Portal with the upload URL
- Step 7: Open URL included in email message
- Step 8: Use „**Datei auswählen**“ and select the file with the public SSH key for upload
- Step 9: Logout from HLRN Service Portal
- Step 10: Use a SSH client to login in to HLRN

Login to the Cray TDS

- Step 1: login to Berlin HLRN complex

```
localhost$ ssh username@blogin.hlrn.de
```

```
Last login: Mon Sep 26 08:00:00 2016 from ...
```

```
blogin1:~ $
```

- Step 2: login to bxcmomtds1 (MOM and interactive working node)

```
blogin1:~ $ ssh bxcmomtds1
```

```
Last login: Mon Sep 26 08:02:00 2016 from ...
```

```
bxcmomtds1:~>
```

How to Access the Intel Parallel Studio XE

- Switch to Intel Programming Environment

```
bxcmomtds1:~> module switch PrgEnv-cray PrgEnv-intel
```

```
bxcmomtds1:~> ftn -V
```

```
Intel(R) Fortran Intel(R) 64 Compiler for applications running on  
Intel(R) 64, Version 17.0.0.098 Build 20160721
```

```
Copyright (C) 1985-2016 Intel Corporation. All rights reserved.
```

Loading a Cray Developer Toolkit

- Load a Cray Developer Toolkit (CDT)

```
bxcmomtds1:~> module load cdt  
Switching to atp/2.0.2.  
Switching to cce/8.5.3.  
Switching to cray-libsci/16.09.1.  
Switching to cray-mpich/7.4.3.  
Switching to craype/2.5.7.  
Switching to modules/3.2.10.5.  
Switching to pmi/5.0.10-1.0000.11050.0.0.ari.  
bxcmomtds1:~>
```

Prepare for Using the Cray Performance Analysis Tools

- Step 1: load the Cray Developer Toolkit

```
bxcmomtds1:~> module load cdt
```

- Step 1: load the Cray Performance Analysis Tools

```
bxcmomtds1:~> module load perftools-base
```

```
bxcmomtds1:~> module load perftools[-lite]
```

```
bxcmomtds1:~> pat_build -v
```

```
CrayPat/X: Version 6.4.1 Revision 6a6694f 06/27/16 17:24:11
```


Ideas for the Hackathon

“Lab Book”: Keep the Records of Your Work

- Make a note of the initial performance for future reference.
 - ❖ May use different compilers (Intel, Cray, GCC)
- Make notes both of your optimization steps and the achieved performance.
 - ❖ This is valuable information for you and for us!
- Prepare a summary of the achieved results
 - ❖ Present it on Day 3 morning, and optionally send to steinke@zib.de
 - ❖ We like to learn about best-practice solutions and what failed

Organization of Work

- Define your environment ...
 - ❖ ... to build your application
 - ❖ ... to run your application
 - ❖ Make lab book notes
- Hackathon teams ... (same code, same research group)