LRZ oneAPI Workshop, June 4th

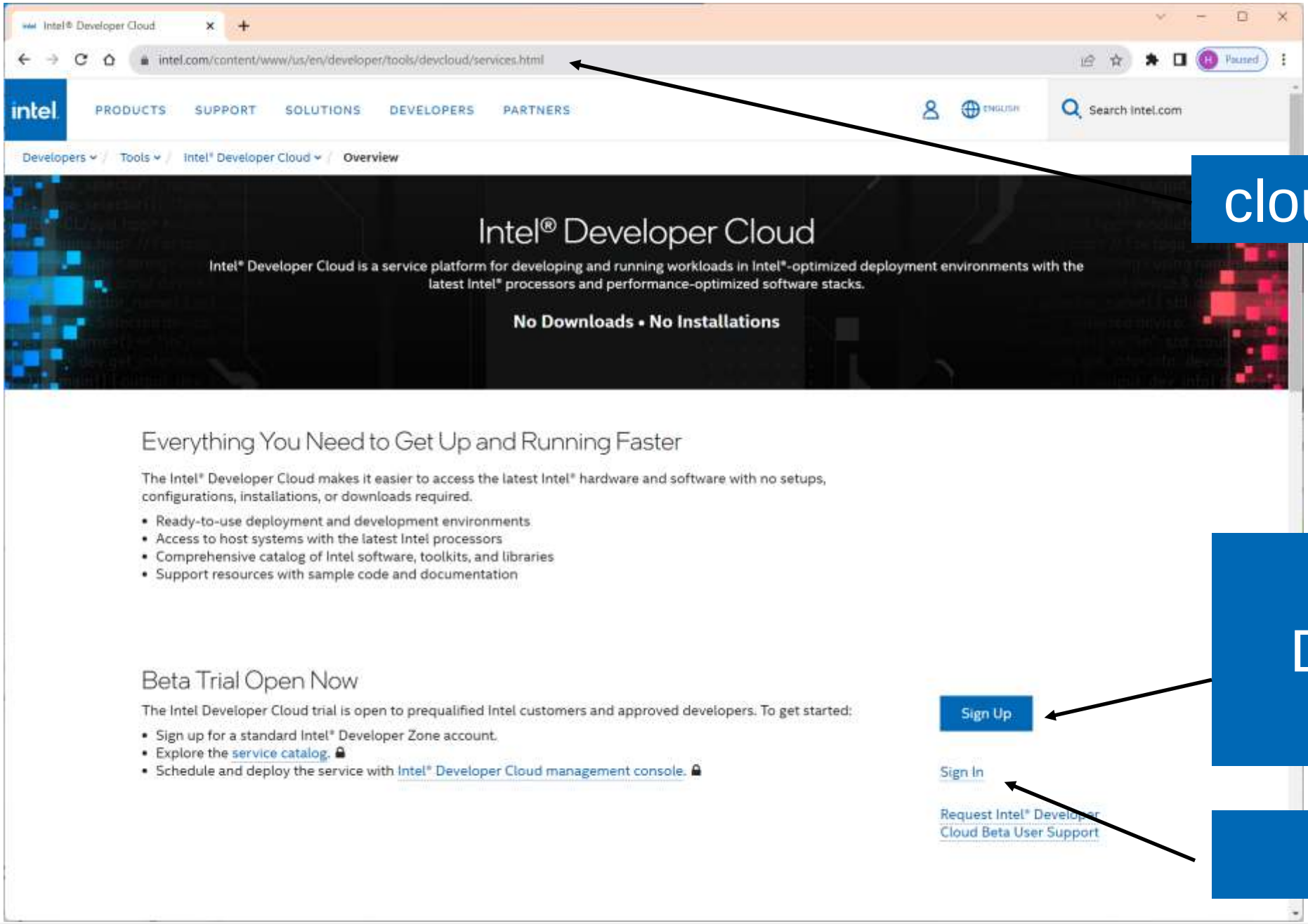
Intel® Developer Cloud (IDC)

Overview

Dr. Heinrich Bockhorst
Dr. Tobias Klöffel

Agenda

- Public Intel Developer Cloud (IDC) slurm usage
- IDC instances for HLRS workshop
- Backup: Old Intel Devcloud
- Backup: Using putty to access IDC instances

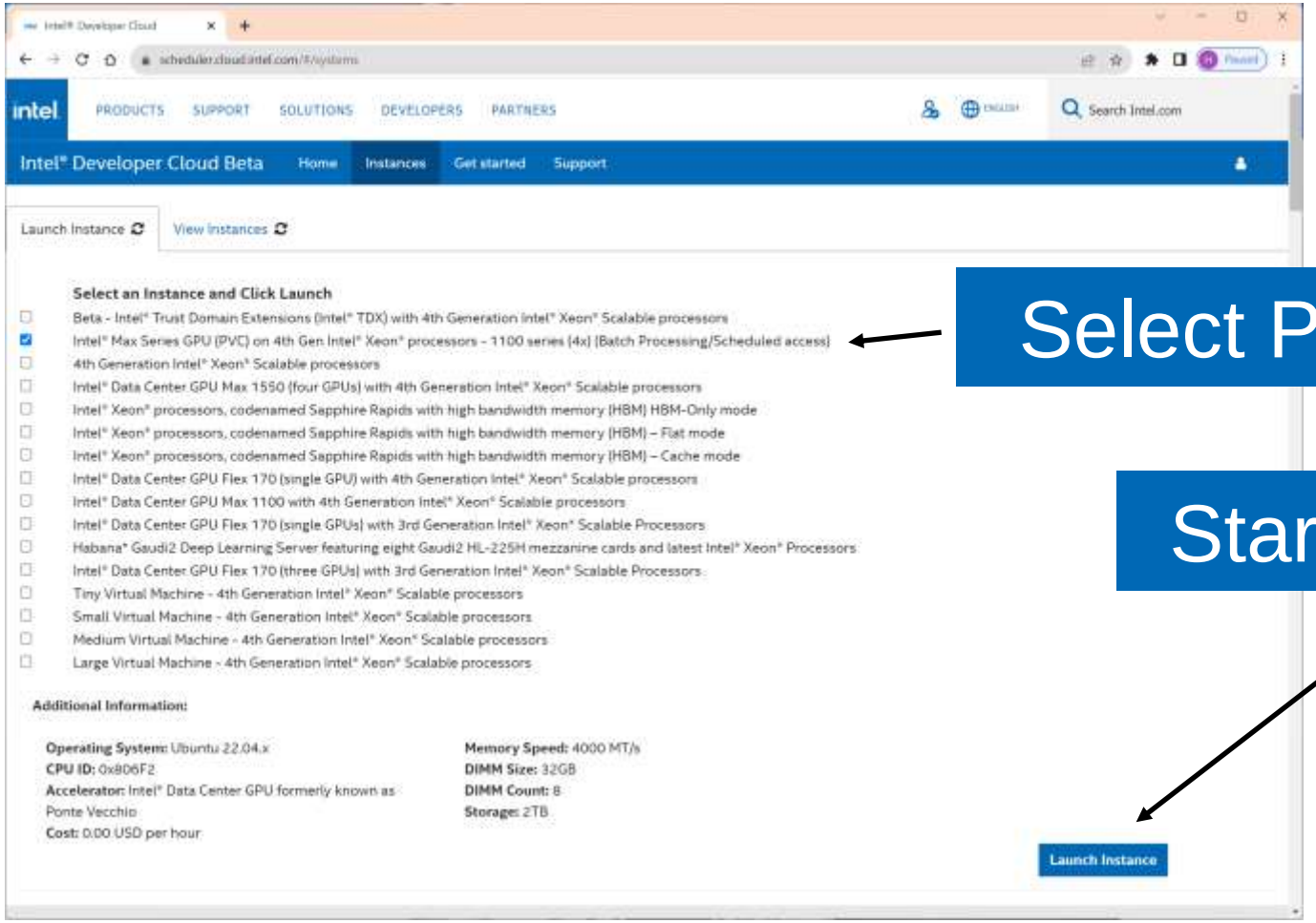


cloud.intel.com

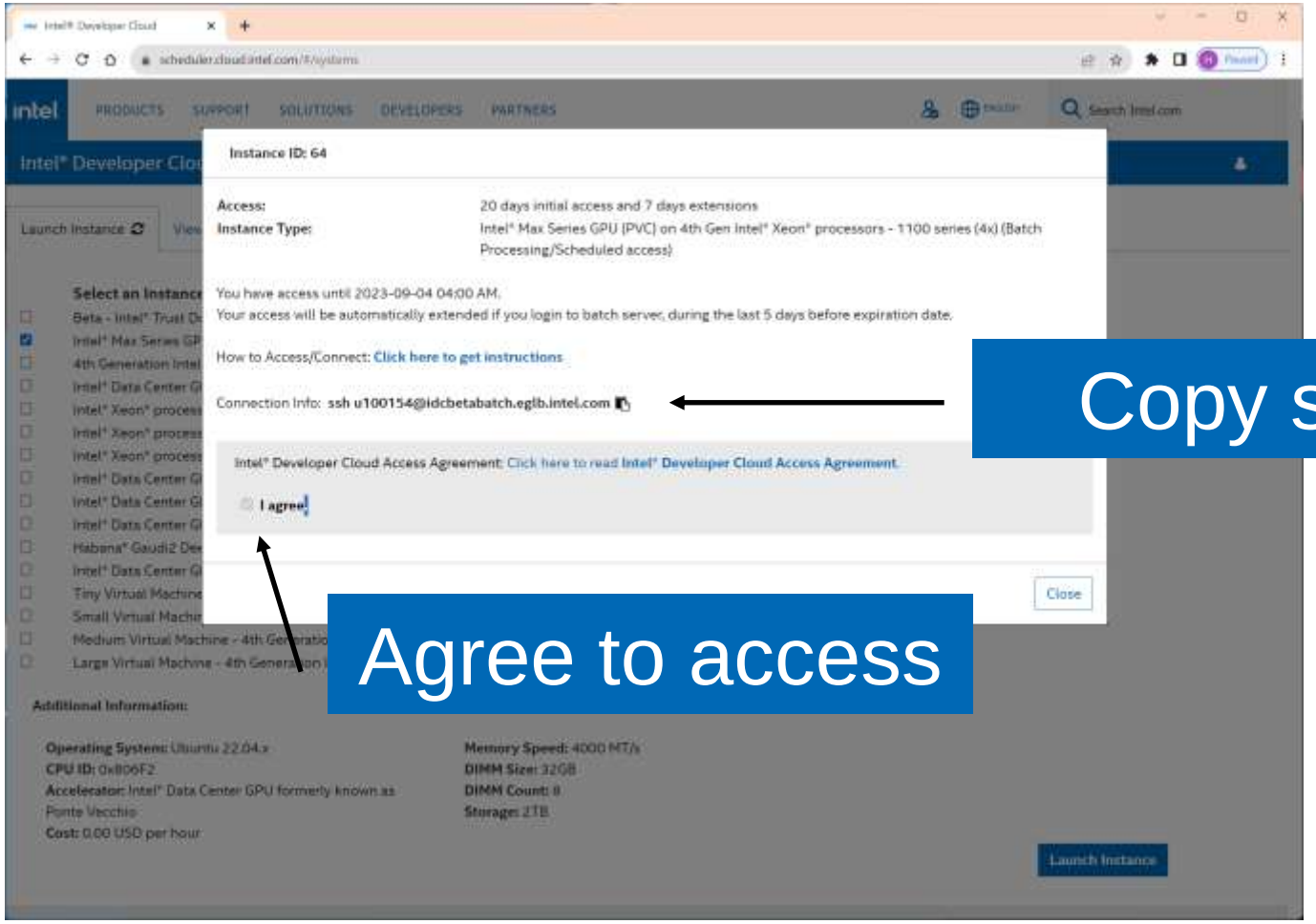
Create Developer account

Sign in

Select your Hardware – second line is PVC



Copy ssh line



Wsl example

Run interactively

```
hbockhor@hbockhor-mobl: ~
Last login: Tue Sep 12 02:12:48 2023 from 134.191.221.84

============================================================
Quick Start Guide: https://slurm.schedmd.com/quickstart.html
============================================================

=========================
Frequently used commands:
=========================

sinfo -al
squeue -al
srun --pty bash
sbatch -p {PARTITION-NAME} {SCRIPT-NAME}
scancel {JOB-ID}

============================================================
Available Intel Pre-Production Platforms to schedule jobs
============================================================
Tue Sep 12 02:21:25 2023
PARTITION   AVAIL  TIMELIMIT   JOB_SIZE ROOT OVERSUBS     GROUPS  NODES       STATE NODELIST
pvc-shared*    up    4:00:00          1   no FORCE:50        all     12   allocated idc-beta-batch-pvc-node-[01-02,04-10,13,15-16]
pvc-shared*    up    4:00:00          1   no FORCE:50        all      9        idle idc-beta-batch-pvc-node-[03,11-12,14,17-21]

By default, only one GPU device is selected, run unset $ONEAPI_DEVICE_SELECTOR to use specific device(s).

================================================================================================================
Support: https://www.intel.com/content/www/us/en/support/contact-intel.html#support-intel-products_67709:59441:231482
================================================================================================================

Your User Account expires on                                    : Oct 02, 2023

WARNING: Before your account expires, please download your data. To auto extend, login here in the last 5 days.
         Avoid storing any files in the /tmp folder to prevent other users from accessing them or removed on reboots.

============================================================================================================
If you DO NOT agree with terms and conditions at
https://scheduler.cloud.intel.com/public/intel_developer_cloud_access_agreement.html DISCONNECT IMMEDIATELY!
```

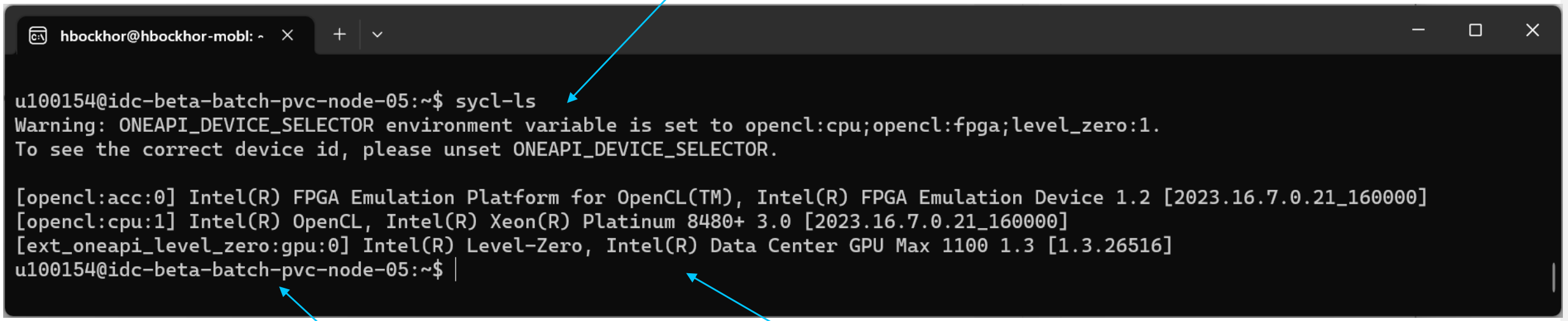
IDC oneapi commands

- Interactive usage:
  `$ srun --pty bash`
- Show jobs
  `$ squeue -l`
  identify my job by looking at node name and timestamp
- Source oneAPI
  `$` source /opt/intel/oneapi/setvars.sh

IDC commands II

- Test SYCL:
  $ sycl-ls

Just single card visible



Node name

Level zero device

IDC reserved nodes

- 4 nodes similar to public IDC

- Personal accounts for HLRS Users who provided a public key
- Account names will be provided on slack
- Follow the recipe in the exercise-support channel


intel

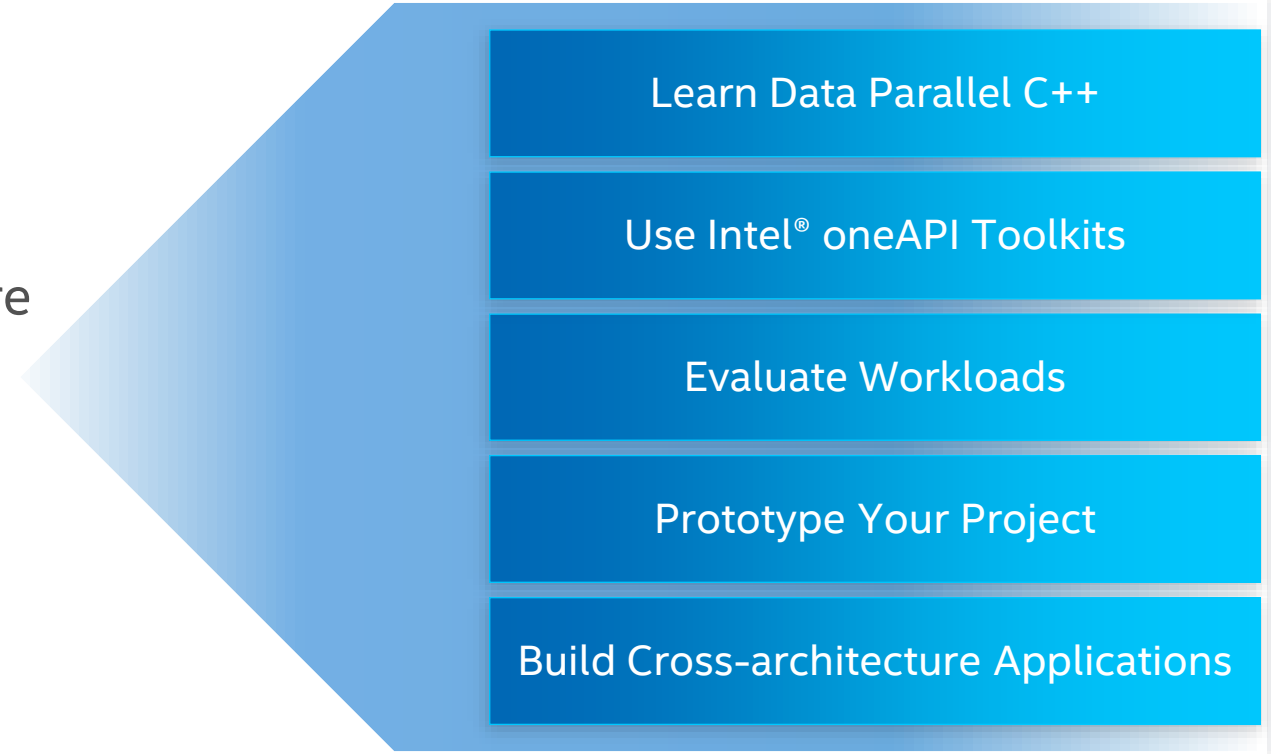
OLD DevCloud

Intel® DevCloud for oneAPI

Free Access, A Fast Way to Start Coding

A development sandbox to develop, test and run workloads across a range of Intel® CPUs, GPUs, and FPGAs using Intel's oneAPI software

For customers focused on data-centric workloads on a variety of Intel® architecture

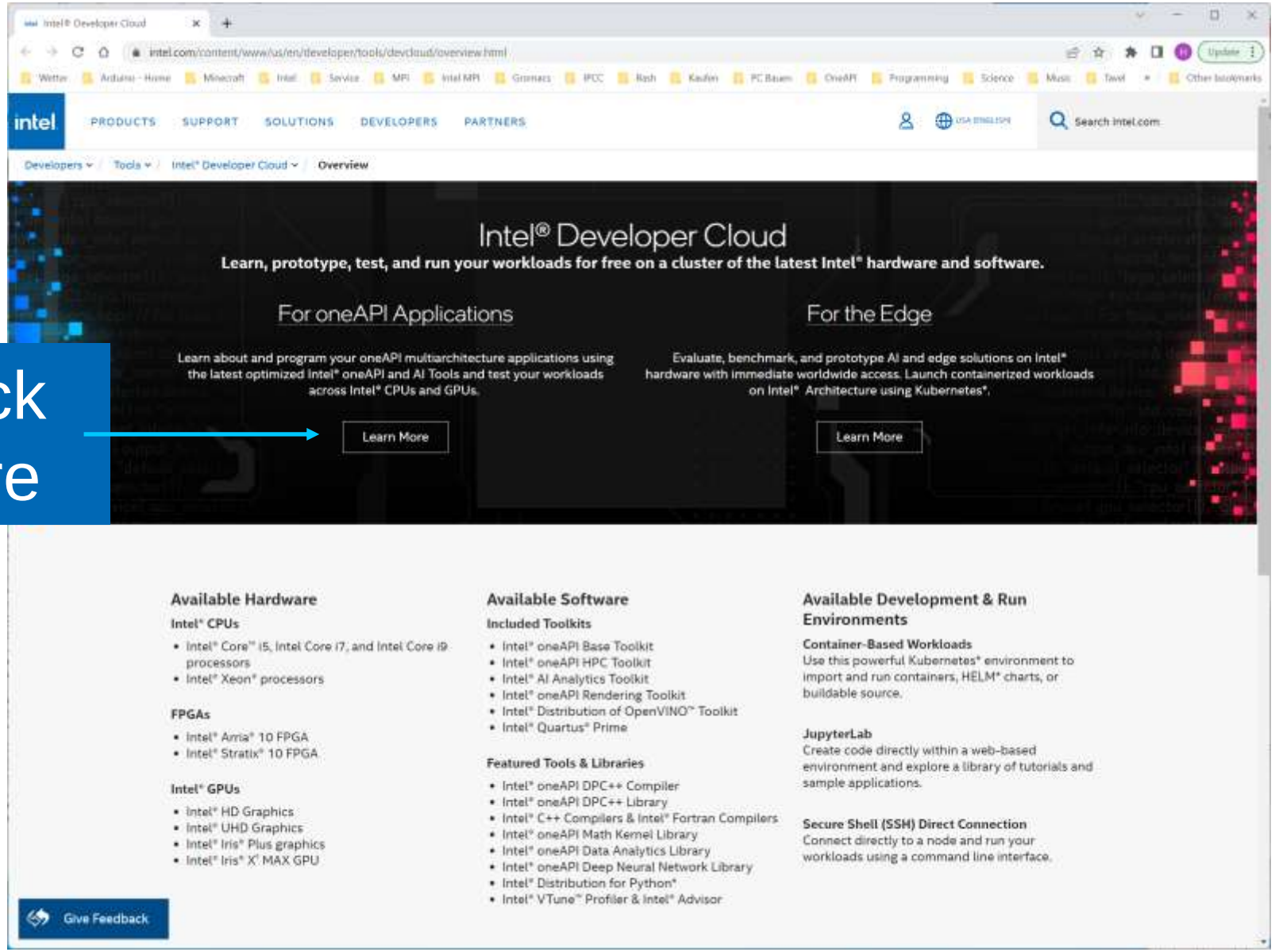


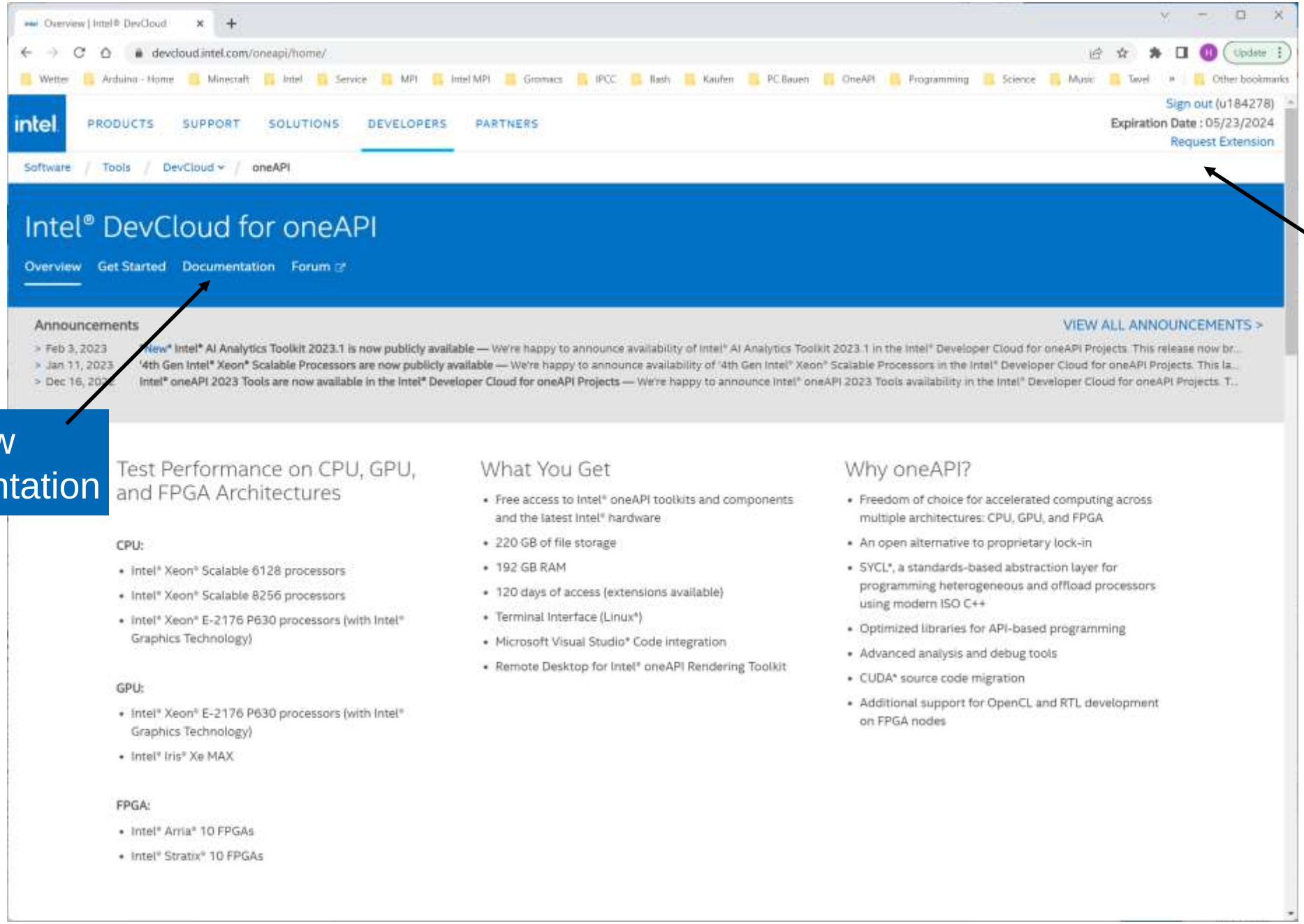
No Downloads | No Hardware Acquisition | No Installation | No Set-up & Configuration

Get Up & Running in Seconds! -- click on "Get Free Access"

https://devcloud.intel.com/oneapi/home/

Click here

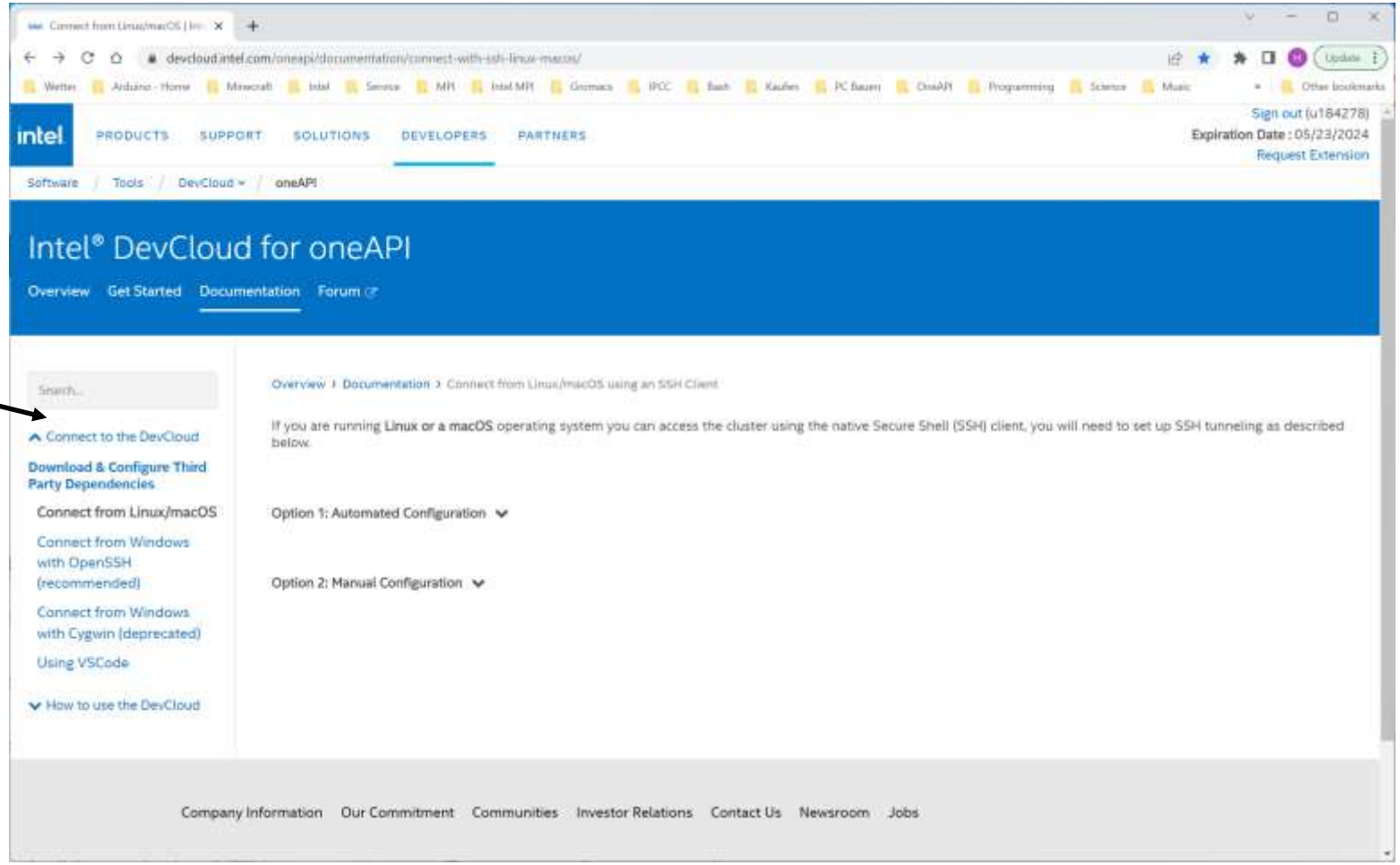

intel
PRODUCTS
SUPPORT
SOLUTIONS
DEVELOPERS
PARTNERS
Search Intel.com
Developers
Tools
Intel® Developer Cloud
Overview
Intel® Developer Cloud
Learn, prototype, test, and run your workloads for free on a cluster of the latest Intel® hardware and software.
For oneAPI Applications
Learn about and program your oneAPI multiarchitecture applications using the latest optimized Intel® oneAPI and AI Tools and test your workloads across Intel® CPUs and GPUs.
Learn More
For the Edge
Evaluate, benchmark, and prototype AI and edge solutions on Intel® hardware with immediate worldwide access. Launch containerized workloads on Intel® Architecture using Kubernetes®.
Learn More
Available Hardware
Intel® CPUs
Intel® Core™ i5, Intel Core i7, and Intel Core i9 processors
Intel® Xeon® processors
FPGAs
Intel® Arria® 10 FPGA
Intel® Stratix® 10 FPGA
Intel® GPUs
Intel® HD Graphics
Intel® UHD Graphics
Intel® Iris® Plus graphics
Intel® Iris® Xᵉ MAX GPU
Available Software
Included Toolkits
Intel® oneAPI Base Toolkit
Intel® oneAPI HPC Toolkit
Intel® AI Analytics Toolkit
Intel® oneAPI Rendering Toolkit
Intel® Distribution of OpenVINO™ Toolkit
Intel® Quartus® Prime
Featured Tools & Libraries
Intel® oneAPI DPC++ Compiler
Intel® oneAPI DPC++ Library
Intel® C++ Compilers & Intel® Fortran Compilers
Intel® oneAPI Math Kernel Library
Intel® oneAPI Data Analytics Library
Intel® oneAPI Deep Neural Network Library
Intel® Distribution for Python*
Intel® VTune™ Profiler & Intel® Advisor
Available Development & Run Environments
Container-Based Workloads
Use this powerful Kubernetes* environment to import and run containers, HELM* charts, or buildable source.
JupyterLab
Create code directly within a web-based environment and explore a library of tutorials and sample applications.
Secure Shell (SSH) Direct Connection
Connect directly to a node and run your workloads using a command line interface.
Give Feedback



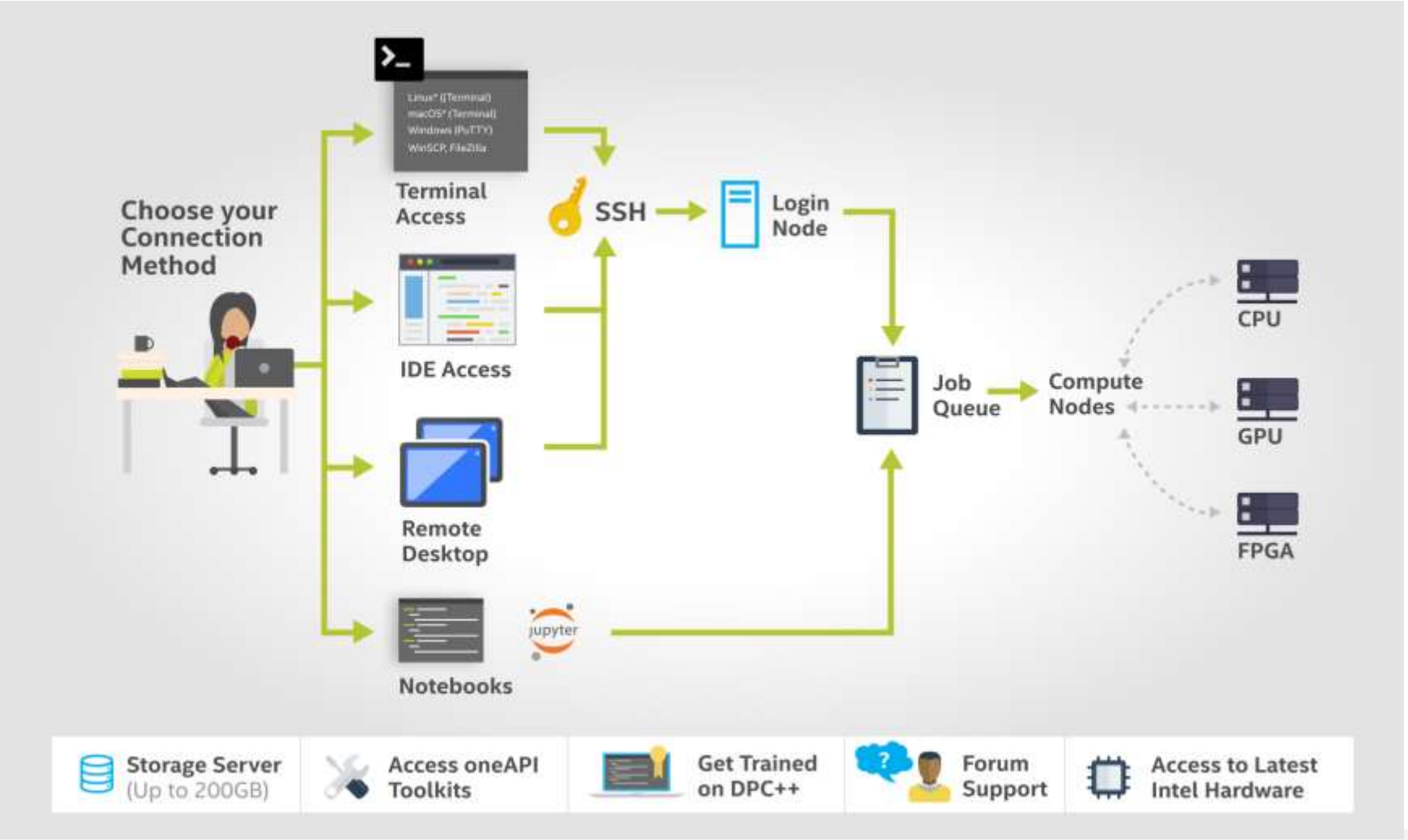
View Documentation

User name and Expiration Date

View: Connect to DevCloud

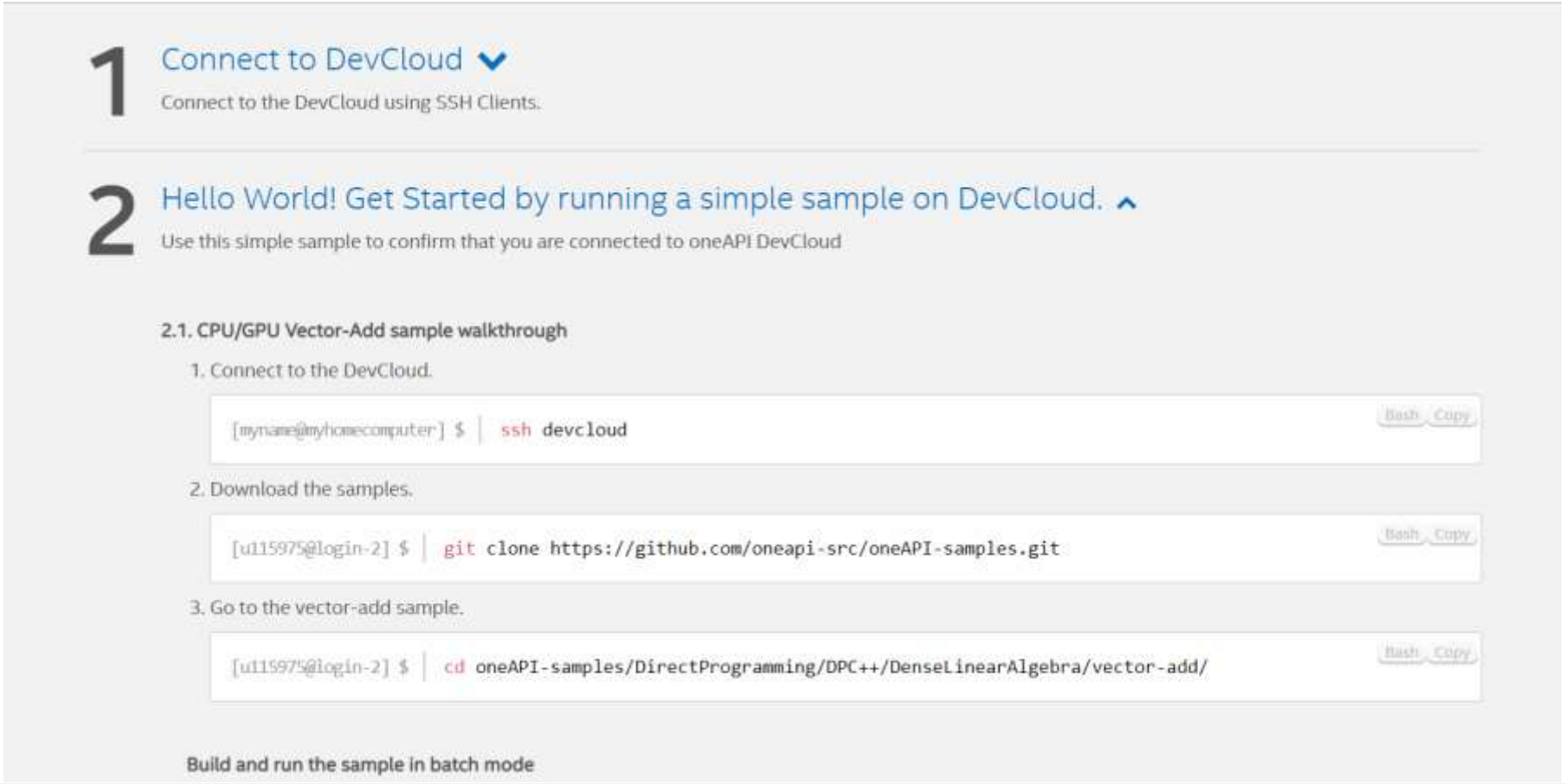


Connection Methods



Get Started (ssh)

https://devcloud.intel.com/oneapi/get_started/baseToolkitSamples



PBS Batch System

- DevCloud uses the PBS Batch System for node access
- Interactive jobs are possible (6 hours default)
- https://devcloud.intel.com/oneapi/documentation/job-submission

How to submit a batch job

```
[u115975@login-2] $ qsub -l nodes=1:gpu:ppn=2 -d . job.sh
```

Note: `-l nodes=1:gpu:ppn=2` (lower case L) is used to assign one full GPU node to the job.
Note: The `-d .` is used to configure the current folder as the working directory for the task.
Note: `job.sh` is the script that gets executed on the compute node.

How to request interactive mode

```
[u115975@login-2] $ qsub -I -l nodes=1:gpu:ppn=2 -d .
```

Note: `-I` (upper case i) is the argument used to request an interactive session.

Basic PBS Queries

- Query available nodes

```
> pbsnodes | grep '^s'
s001-n001
…
```

- Check node characteristics

```
> pbsnodes | grep properties | sort -u
     properties = core,tgl,i9-11900kb,ram32gb,netgbe,gpu,gen11
     properties = xeon,cfl,e-2176g,ram64gb,net1gbe,gpu,gen9
     properties = xeon,clx,ram192gb,net1gbe,batch,extended,fpga,stratix10,fpga_runtime
     properties = xeon,icx,gold6348,ramgb,netgbe,jupyter,batch
     properties = xeon,icx,plat8380,ram2tb,net1gbe,batch
     properties = xeon,skl,gold6128,ram192gb,net1gbe,fpga_runtime,fpga,agilex
     properties = xeon,skl,gold6128,ram192gb,net1gbe,fpga_runtime,fpga,arria10
     properties = xeon,skl,gold6128,ram192gb,net1gbe,jupyter,batch
     properties = xeon,skl,gold6128,ram192gb,net1gbe,jupyter,batch,fpga_compile
     properties = xeon,skl,ram384gb,net1gbe,renderkit
     properties = xeon,spr,max9480,ram256gb,netgbe,batch,hbm
     properties = xeon,spr,ram1024gb,netgbe,dnp50
```

Basic oneAPI Queries

- oneAPI environment on node
  - **source /opt/intel/oneapi/setvars.sh # or load module**
  - **which icpx**

/glob/development-tools/versions/oneapi/2023.1.2/oneapi/compiler/2023.1.0/linux/bin/icpx

- Check GPU characteristics

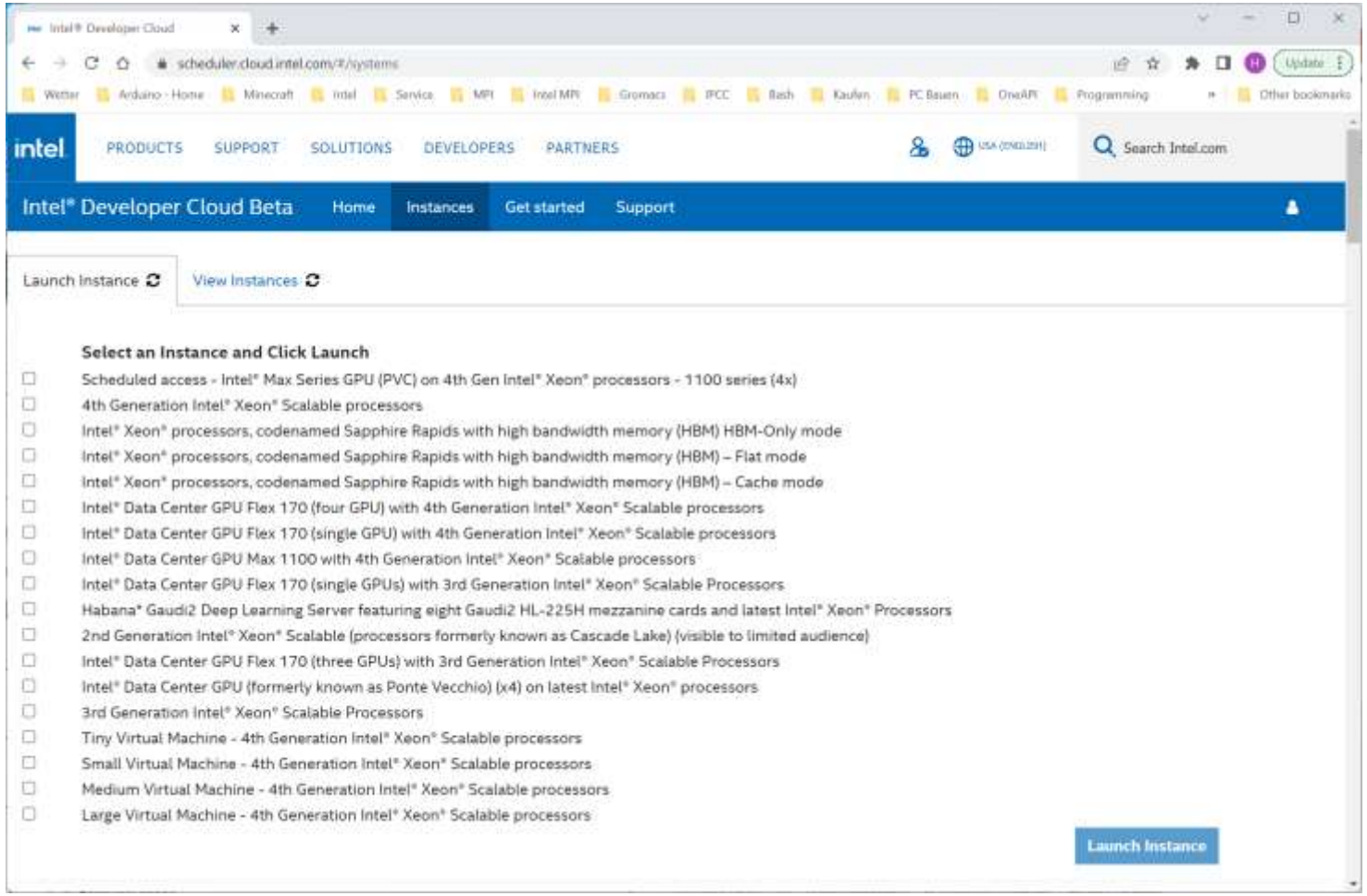
> sycl-ls --verbose

```
...
Platform [#3]:
    Version  : 1.3
    Name     : Intel(R) Level-Zero
    Vendor   : Intel(R) Corporation
    Devices  : 1
        Device [#0]:
        Type       : gpu
        Version    : 1.3
        Name       : Intel(R) UHD Graphics [0x9a60]
        Vendor     : Intel(R) Corporation
        Driver     : 1.3.24595
default_selector()      : gpu, Intel(R) Level-Zero, Intel(R) UHD Graphics [0x9a60] 1.3 [1.3.24595]
accelerator_selector()  : No device of requested type available. Please chec...
cpu_selector()          : cpu, Intel(R) OpenCL, 11th Gen Intel(R) Core(TM) i9-11900KB @ 3.30GHz 3.0 [2023.15.3.0.20_160000]
gpu_selector()          : gpu, Intel(R) Level-Zero, Intel(R) UHD Graphics [0x9a60] 1.3 [1.3.24595]
custom_selector(gpu)    : gpu, Intel(R) Level-Zero, Intel(R) UHD Graphics [0x9a60] 1.3 [1.3.24595]
custom_selector(cpu)    : cpu, Intel(R) OpenCL, 11th Gen Intel(R) Core(TM) i9-11900KB @ 3.30GHz 3.0 [2023.15.3.0.20_160000]
```

Notes:

- Login nodes have very low limits: please compile etc. on compute nodes!
- Please use tools only on compute nodes for same reason!
- Jupyter notebooks also offer a terminal – in case of trouble with ssh.
- Mark Expiration Date in your Calendar!

New Intel Developer Cloud – cloud.intel.com



Notices & Disclaimers

Texas Advanced Computing Center (TACC) Frontera references
Article: *HPCWire: Visualization & Filesystem Use Cases Show Value of Large Memory Fat Notes on Frontera*.
www.intel.com/content/dam/support/us/en/documents/memory-and-storage/data-center-persistent-mem/Intel-Optane-DC-Persistent-Memory-Quick-Start-Guide.pdf
software.intel.com/content/www/us/en/develop/articles/introduction-to-programming-with-persistent-memory-from-intel.html
wreda.github.io/papers/assise-osdi20.pdf

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Your costs and results may vary.

Intel technologies may require enabled hardware, software or service activation.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

Code Samples

- OpenMP offload

- DPC++

MandelbrotOMP sample
This sample demonstrates how to accelerate program performance with SIMD and parallelization using OpenMP*, in the context of calculating the Mandelbrot set.

View code on GitHub*

openMP Reduction Sample
The openmp_reduction code sample is a simple program that calculates pi. This program is implemented using C++ and openMP for Intel CPU and accelerators.

View code on GitHub*

ISO3DFD Open MP Offload Sample
The ISO3DFD sample refers to Three-Dimensional Finite-Difference Wave Propagation in Isotropic Media. It is a three-dimensional stencil to simulate a wave propagating in a 3D isotropic medium and shows some of the more common challenges and techniques when targeting OMP Offload devices (GPU) in more complex applications to achieve good performance.

View code on GitHub*

Direct Programming/DPC++

Vector-Add
This simple vector-add program in Data Parallel C++ (DPC++) supports FPGAs, GPUs, and CPUs.

View code on GitHub*

Mandelbrot Sample
Mandelbrot is an infinitely complex fractal patterning that is derived from a simple formula. It demonstrates using DPC++ for offloading computations to a GPU (or other devices) and shows how processing time can be optimized and improved with parallelism.

View code on GitHub*

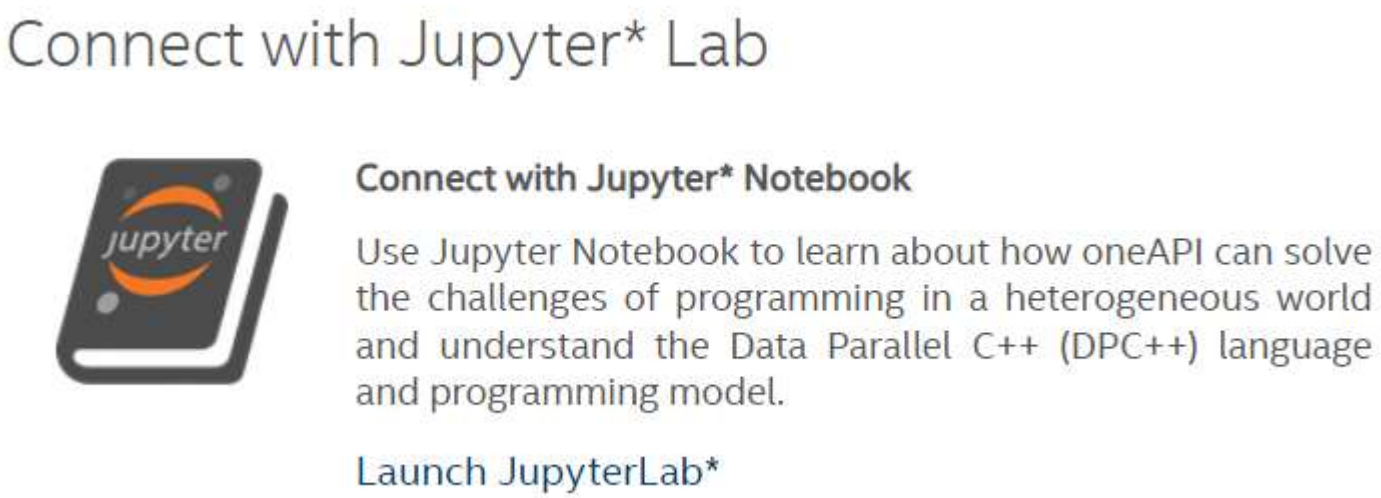
Complex Multiplication Sample
Complex multiplication is a program that multiplies two large vectors of Complex numbers in parallel and verifies the results. It also implements a custom device selector to target a specific vendor device. This program is implemented using C++ and DPC++ language for Intel CPU and accelerators. The Complex class is a custom class, and this program shows how we can use custom types of classes in a DPC++ program.

Sepia Filter
A program that converts an image to sepia tone.

View code on GitHub*

Connection with Jupyter* Notebook

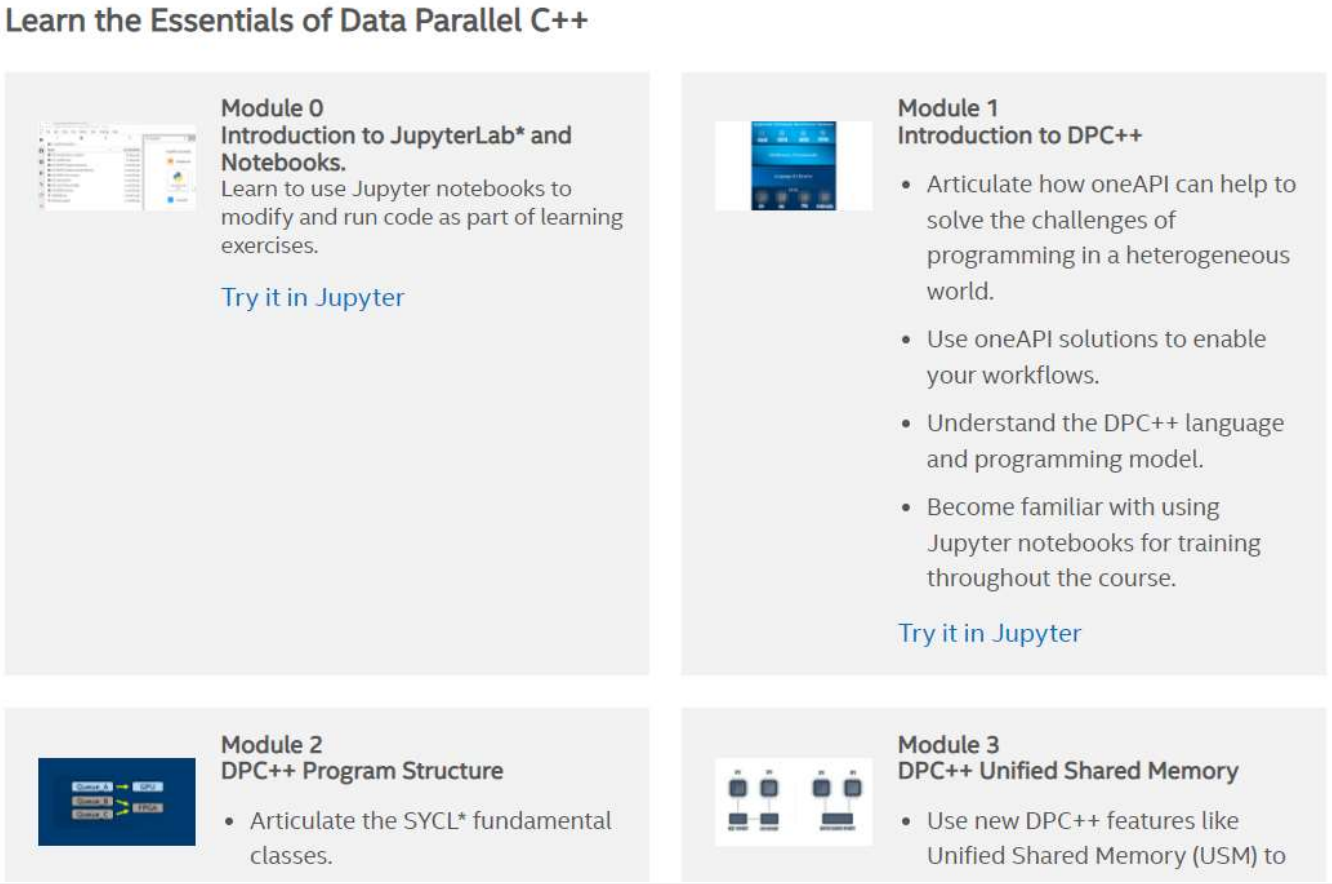
- JupyterLab*



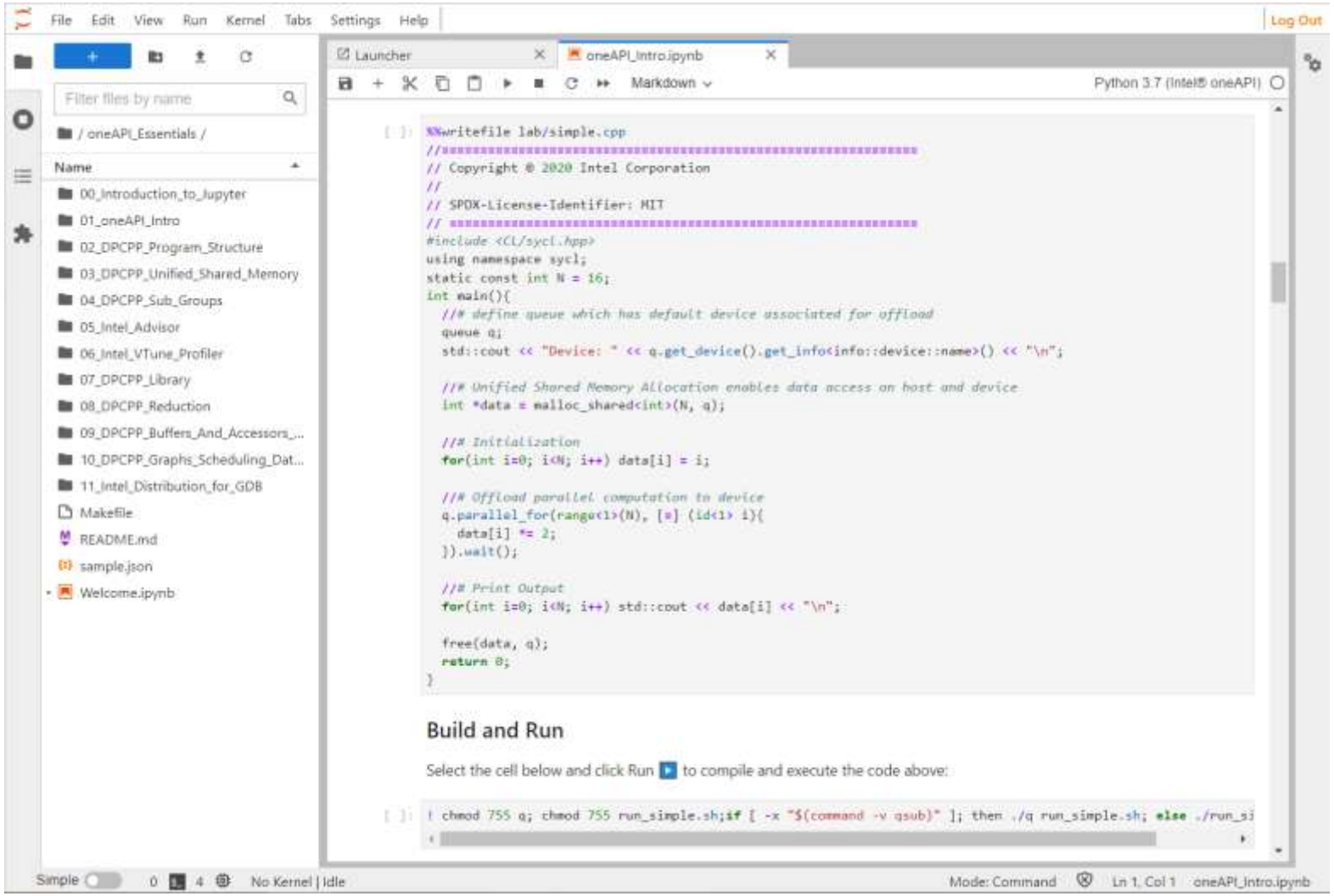
- JupyterLabs* for AI

Basic Training Modules in JupyterLab*

- https://devcloud.intel.com/oneapi/get_started/baseTrainingModules



oneAPI Essentials in JupyterLab*



Connection with Visual Studio Code*

Connect to DevCloud with Visual Studio Code

NOTE: Windows users must first download and install Cygwin before proceeding. Once it has been installed, return to this page to configure your connection.

Requirements:

- Windows users install Cygwin from the installation page
- VS Code
- VS Code SSH extension
- VS Code DevCloud Connector extension

Cygwin Installation

The Cygwin* environment offers a convenient way of connecting to the Intel® DevCloud from a local machine running Windows*, whether you have a direct connection or find yourself behind a proxy. If you already have Cygwin installed, please skip to the SSH connection instructions.

NOTE: Your Cygwin installation requires the openssh (ssh), nc and nano packages.

The following instructions will help you install a minimal version of Cygwin for accessing Intel DevCloud. For your convenience we're providing a simple script that automates the installation of Cygwin.

Download install_cygwin.bat from the installation page. It can be run from anywhere on your disk, either by executing it from the terminal or by double clicking on it.

The script uses curl to download the Cygwin setup file. When asked to provide proxy details, you can do so by entering proxy:port when asked, or by simply hitting enter to continue without a proxy.

The default installation path is c:\cygwin64. The script will prompt you to change this if you wish to install elsewhere.

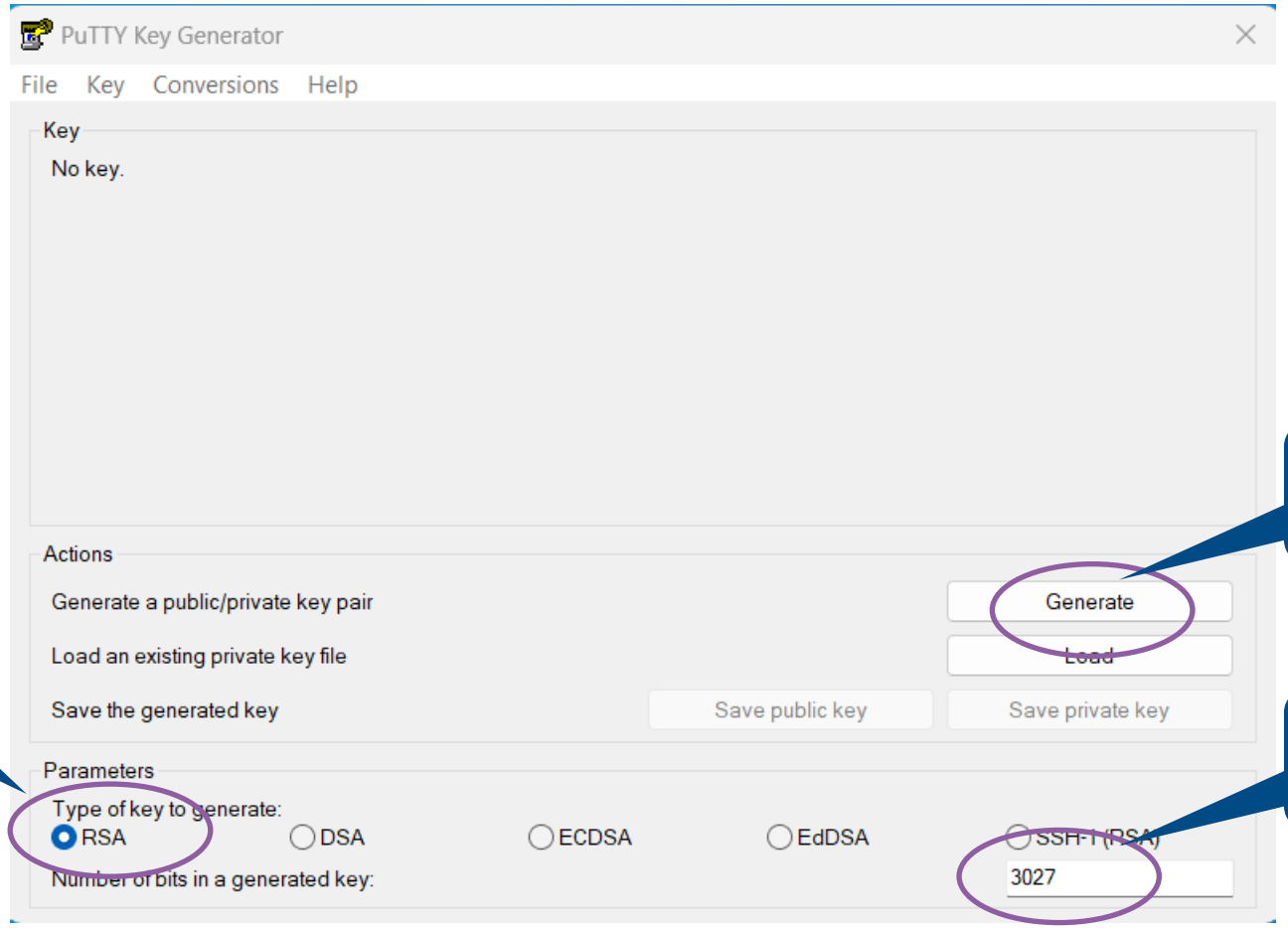
Several Cygwin packages are downloaded during the installation. The script is configured to use mirrors.kernel.org as the default download site. A full list of Cygwin mirror sites can be found on the Cygwin homepage https://www.cygwin.com/.

Using putty to access IDC instances

for Windows users

Generate ssh-key with putty-gen - 1

Open “puttygen”

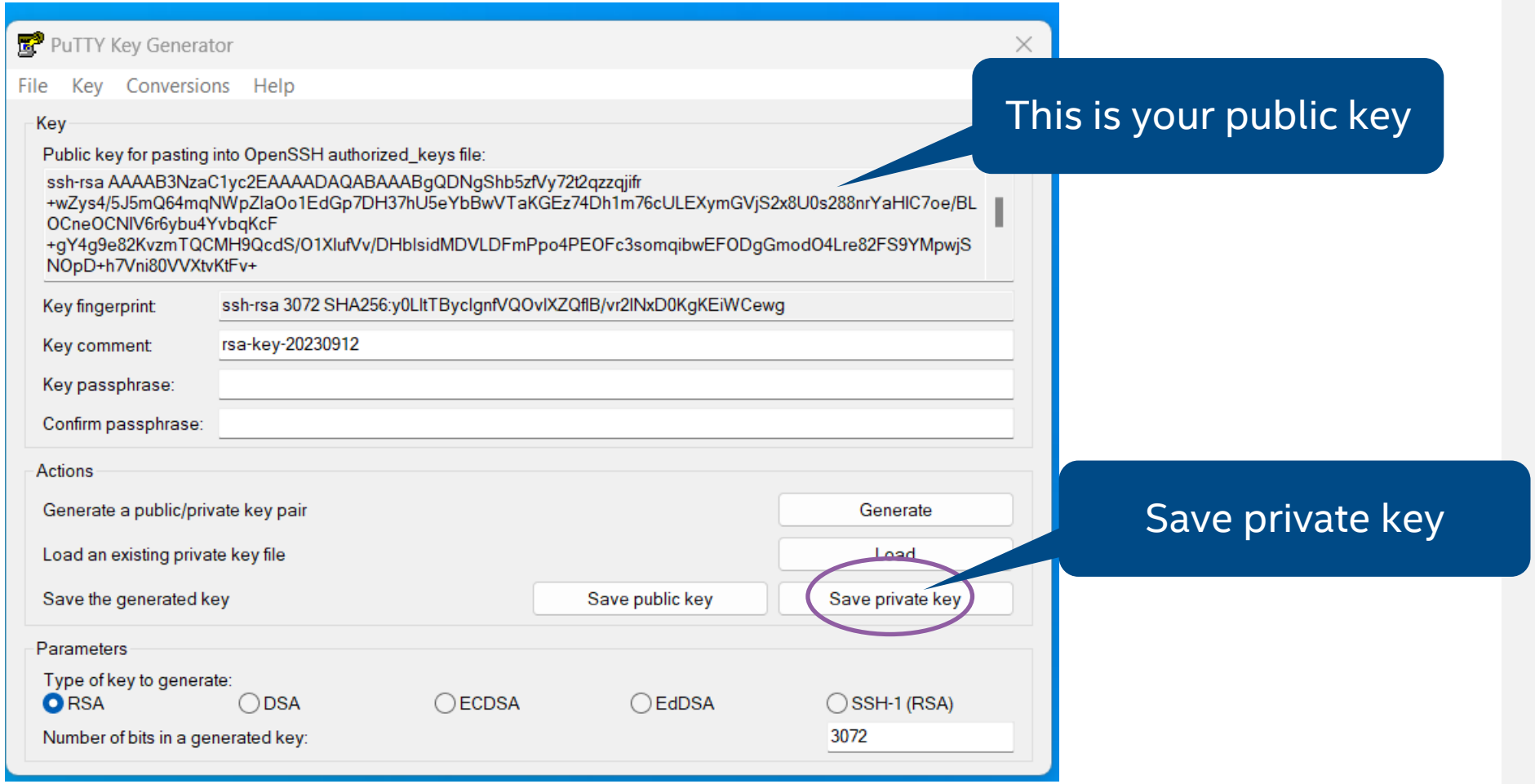


Select RSA

Click ‘Generate’ and move your mouse

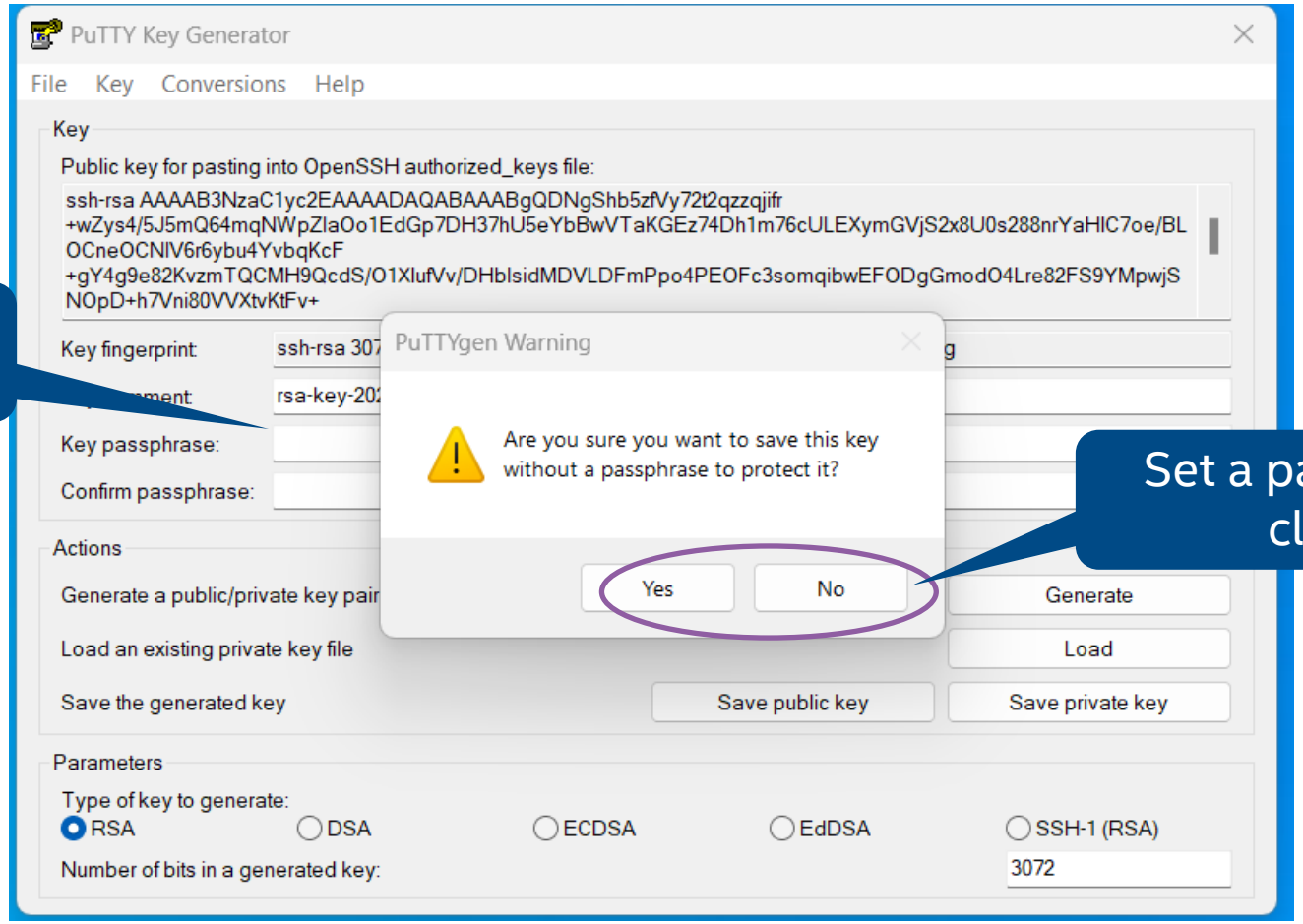
3072

Generate ssh-key with putty-gen - 2



Generate ssh-key with putty-gen - 3

If you did not set a passphrase

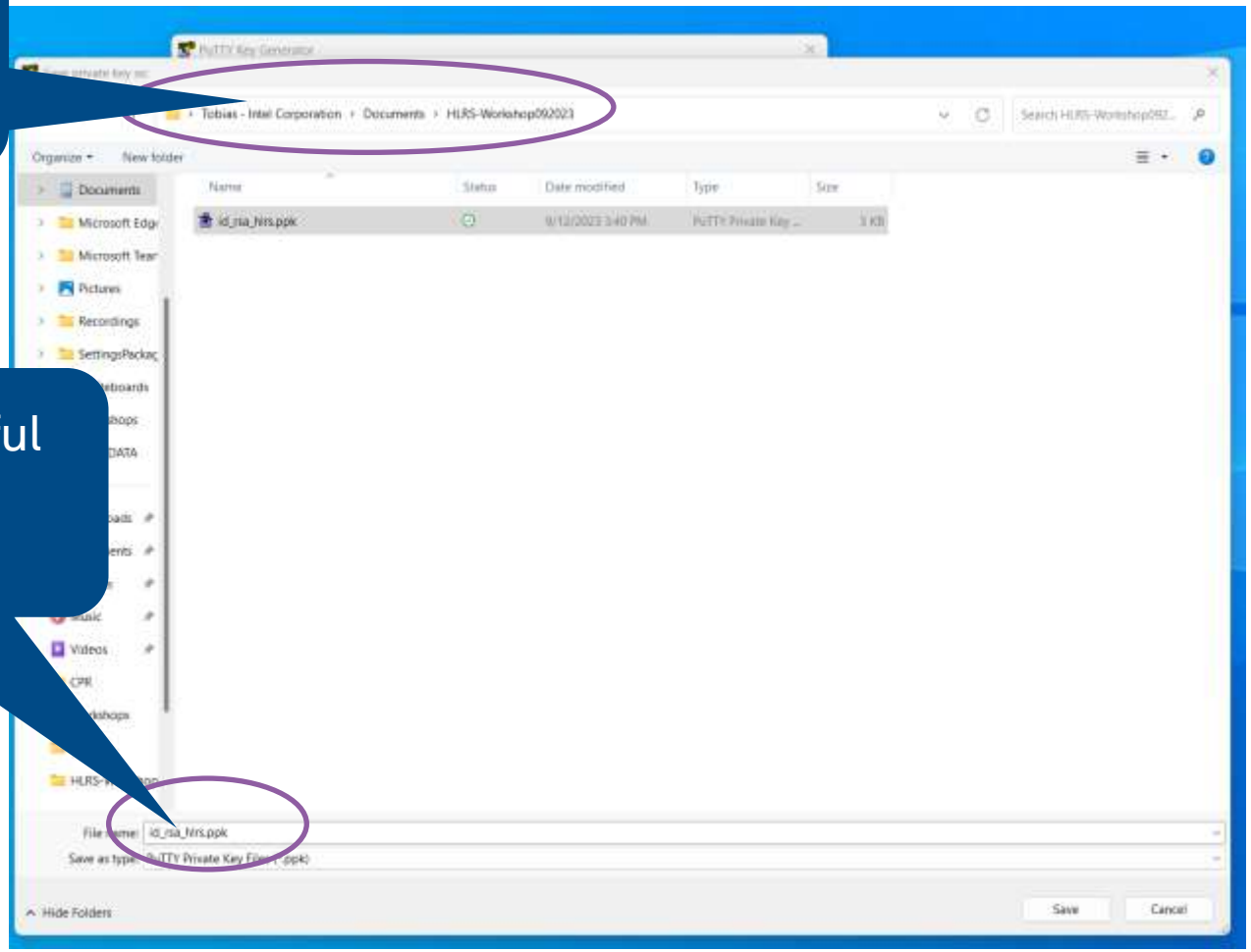


Set a passphrase or click yes

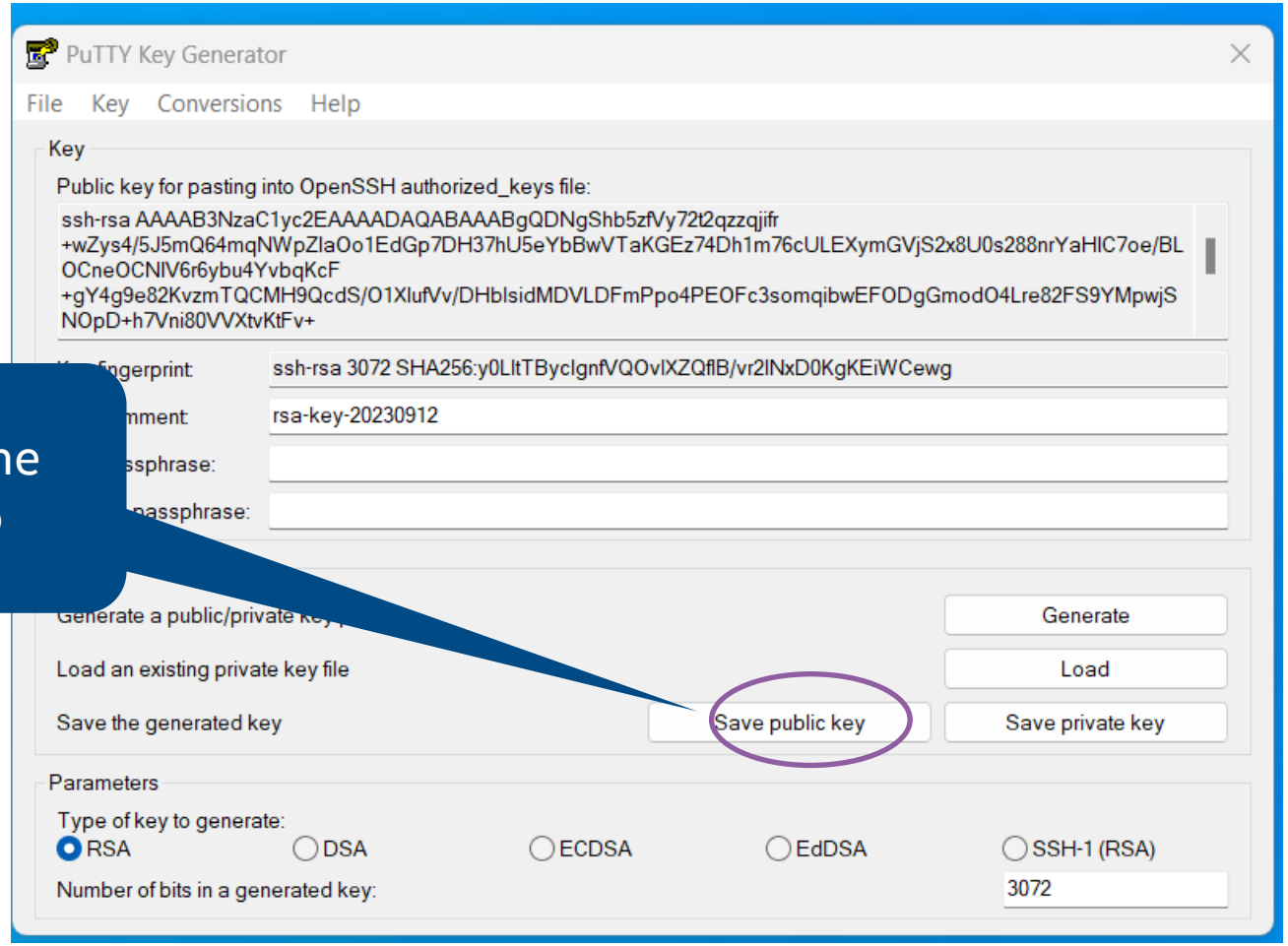
Generate ssh-key with putty-gen - 4

Remember the path you choose

Select a meaningful name: id_rsa_hlrs.ppk



Generate ssh-key with putty-gen - 5

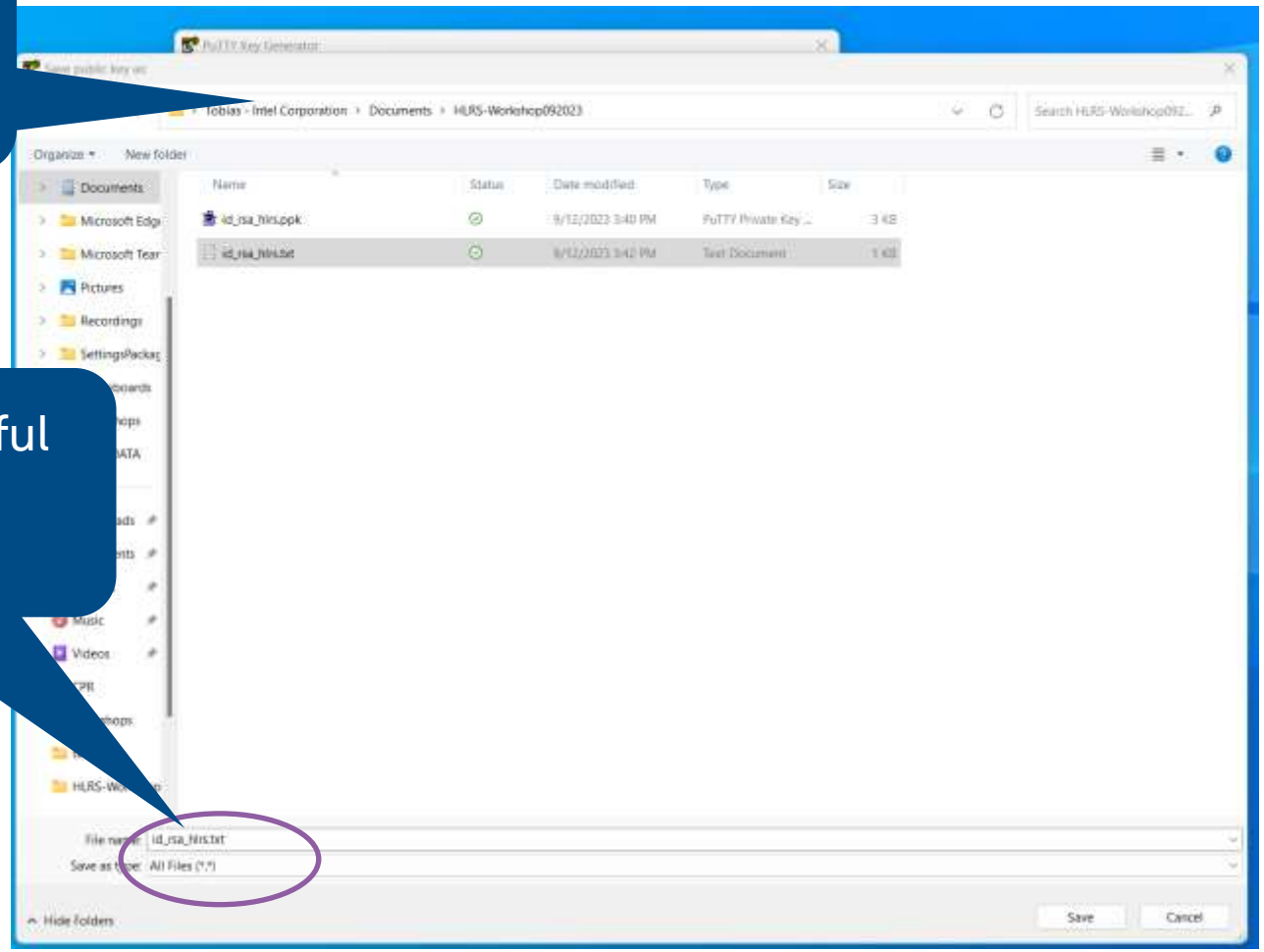


Useful to save the public key, too

Generate ssh-key with putty-gen - 6

Remember the path you choose

Select a meaningful name:
id_rsa_hlrs.txt

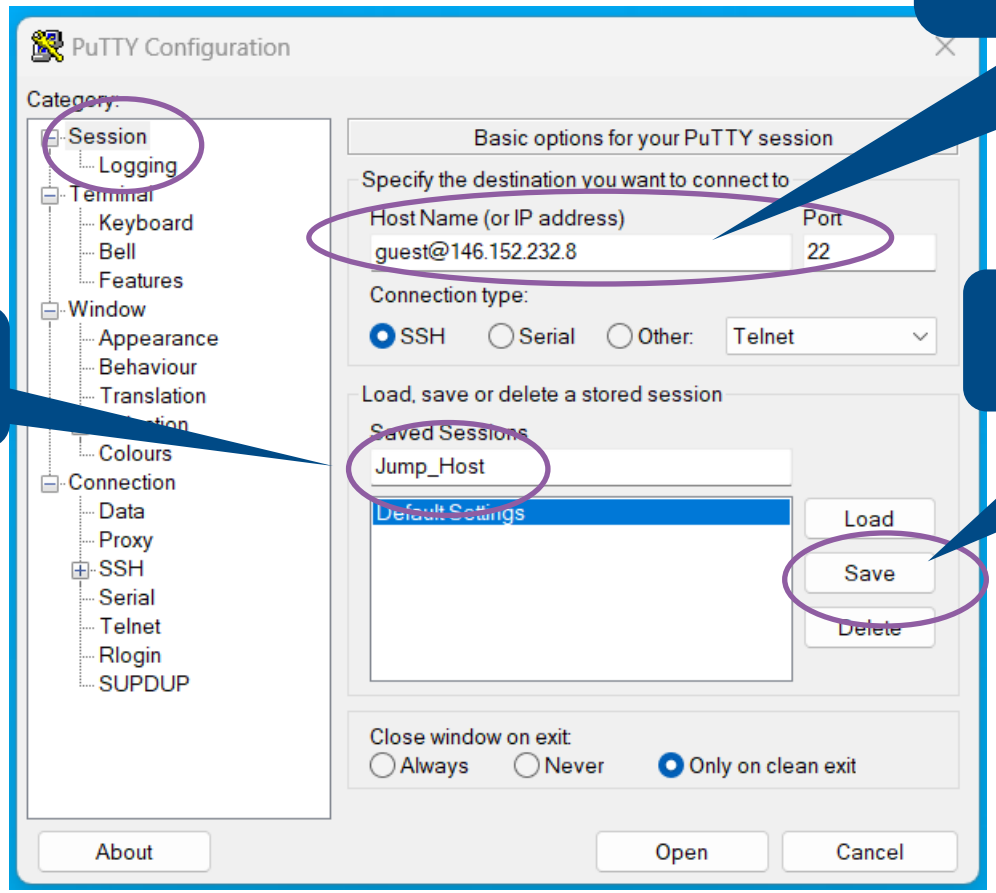


Connection with Putty – JumpHost configuration - 1

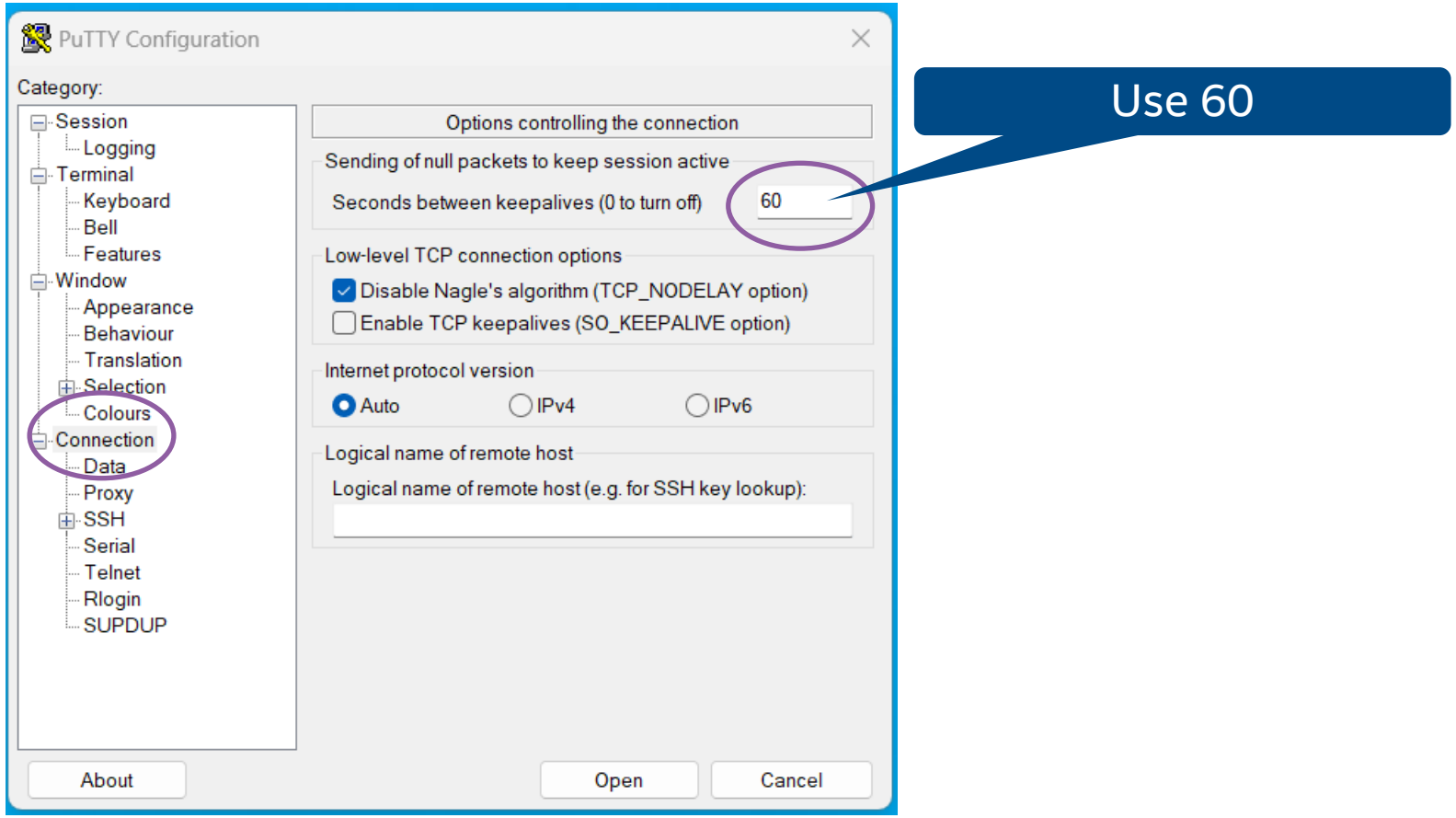
HostName (or IP address)
guest@146.152.232.8
Port: 22

Jump_Host

Do not forget to click save!

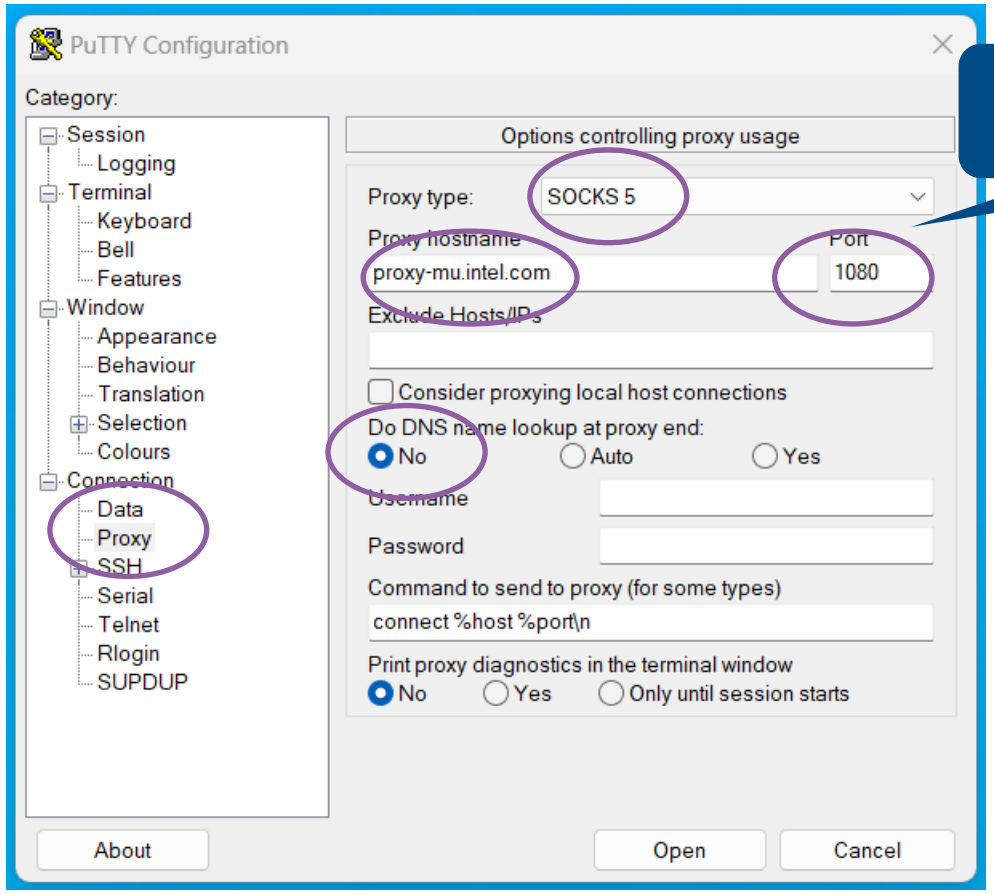


Connection with Putty – JumpHost configuration - 2

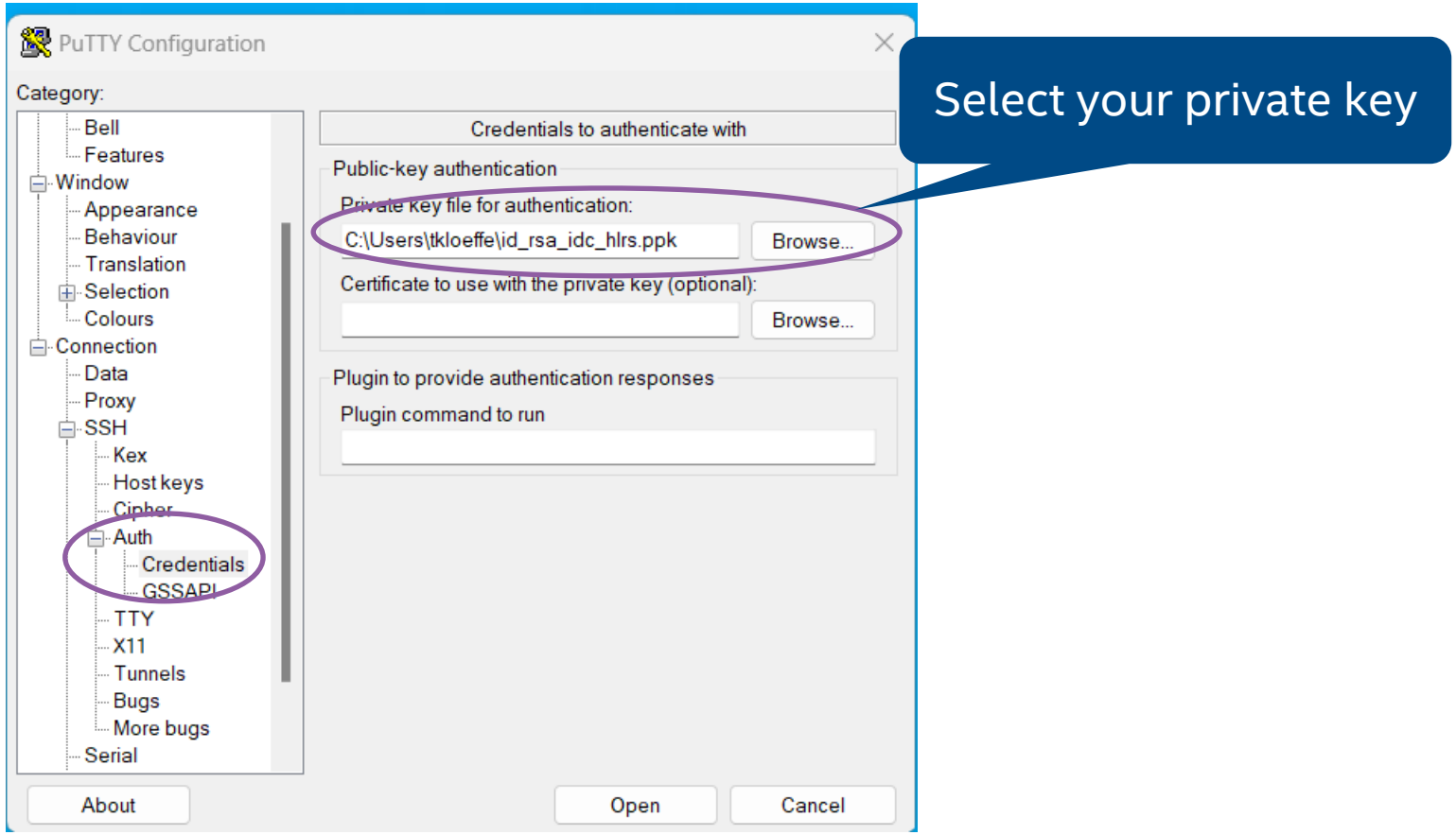


Connection with Putty – JumpHost configuration – 3

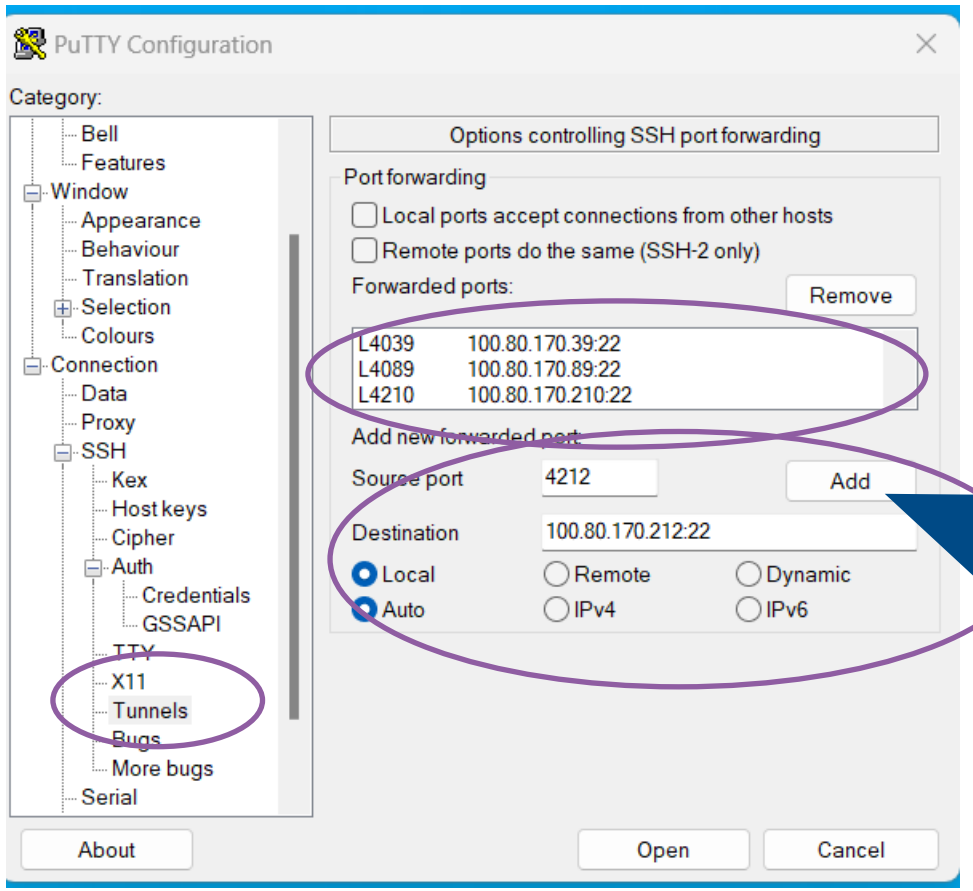
Skip if you do not need a proxy to connect to the internet!



Connection with Putty – JumpHost configuration – 4



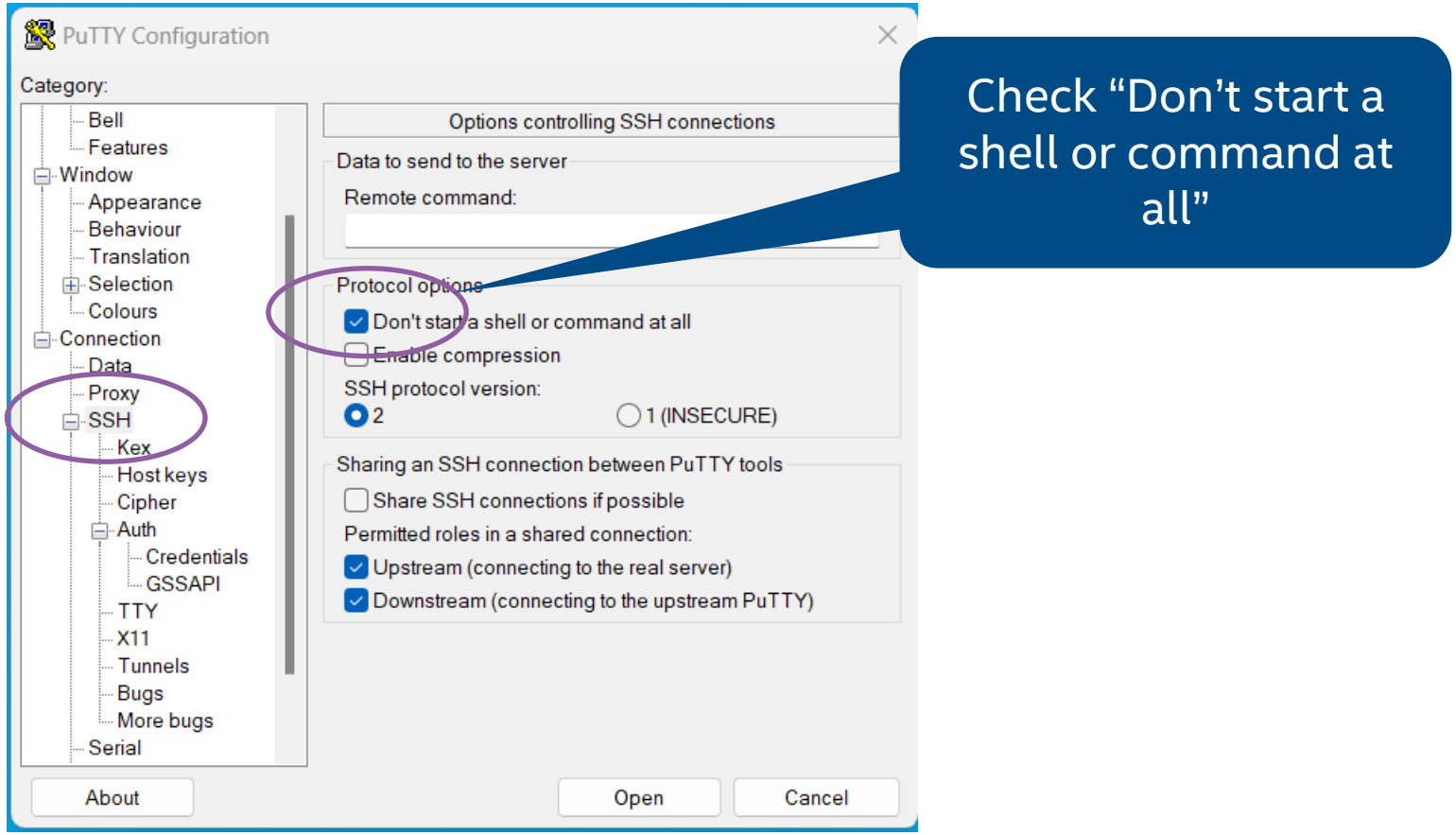
Connection with Putty – JumpHost configuration – 4



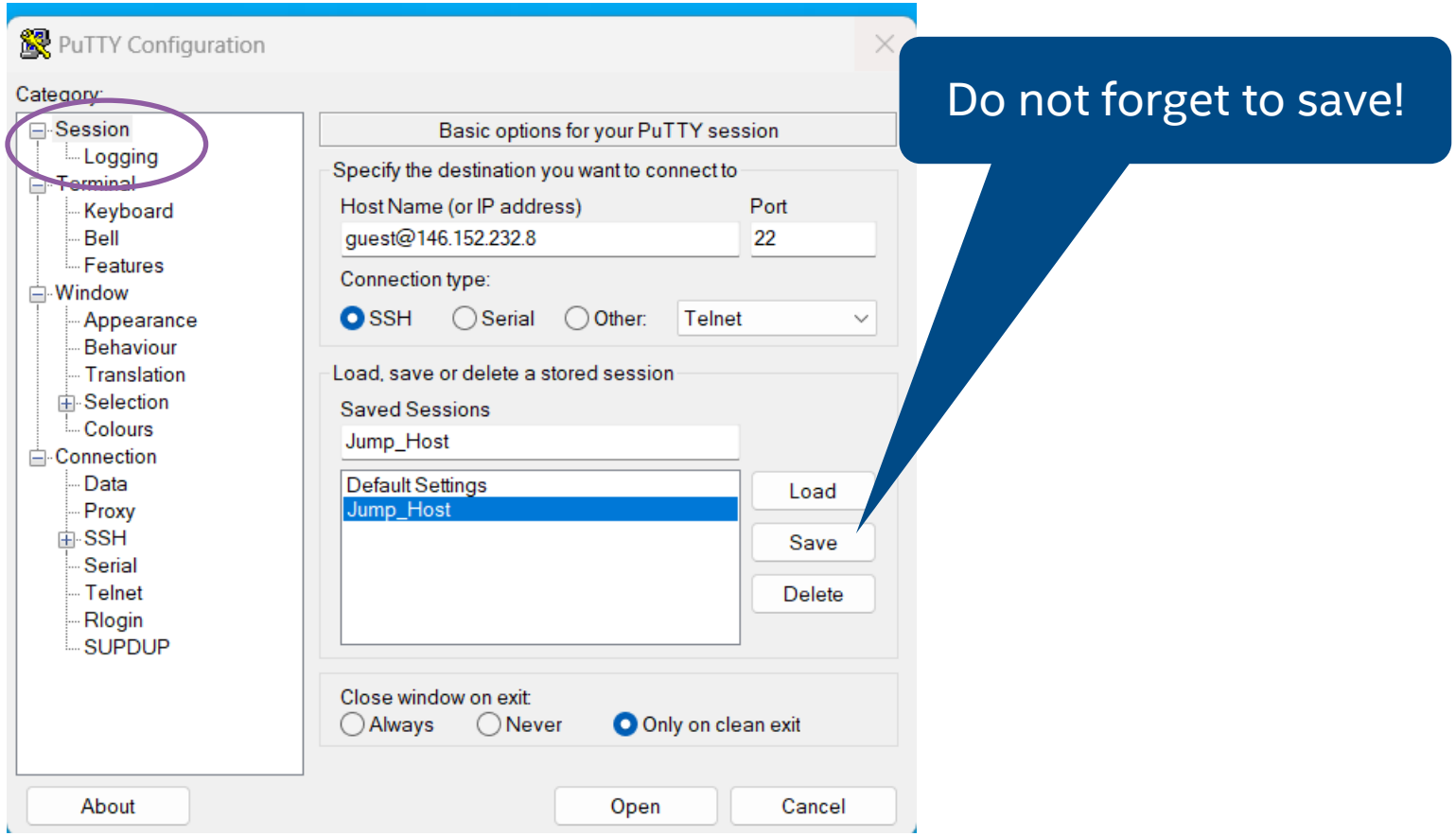
Configure all four destinations:
100.80.170.212:22
Source port: 4212
100.80.170.210:22
Source port: 4210
100.80.170.89:22
Source port: 4089
100.80.180.39:22
Source port: 4039

Don't forget to click 'add'

Connection with Putty – JumpHost configuration – 5



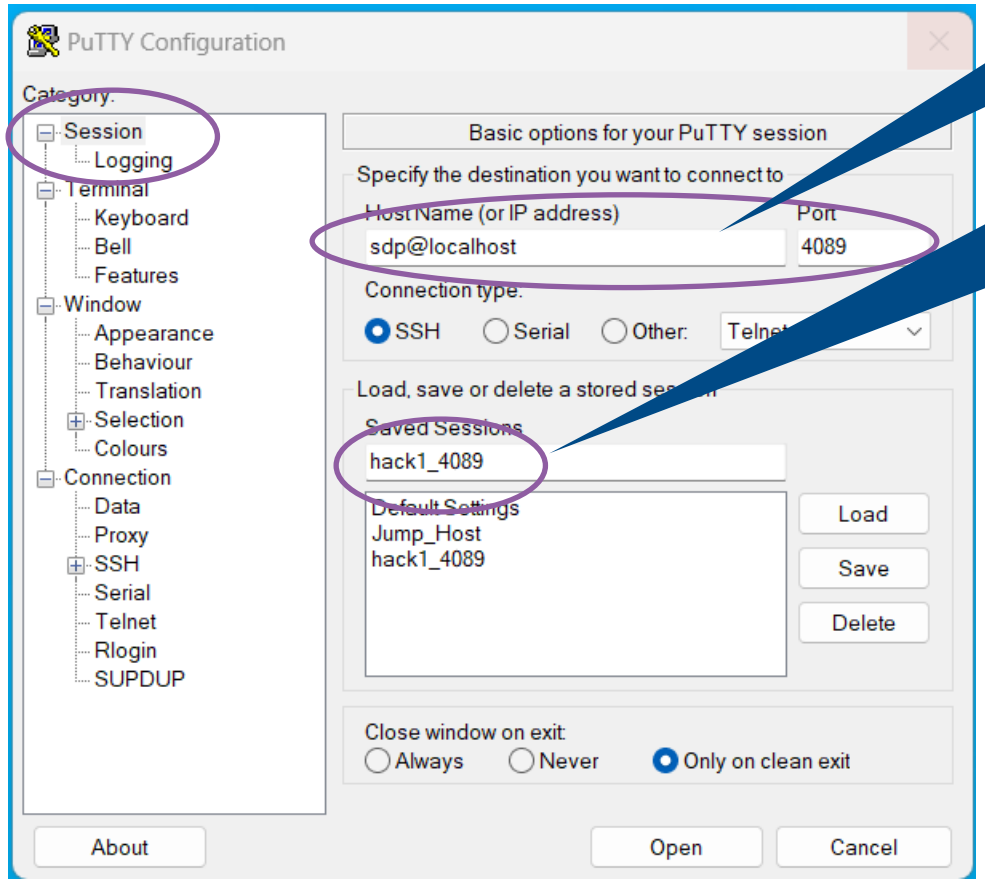
Connection with Putty – JumpHost configuration – 6



Connection with Putty – Node configuration - 1

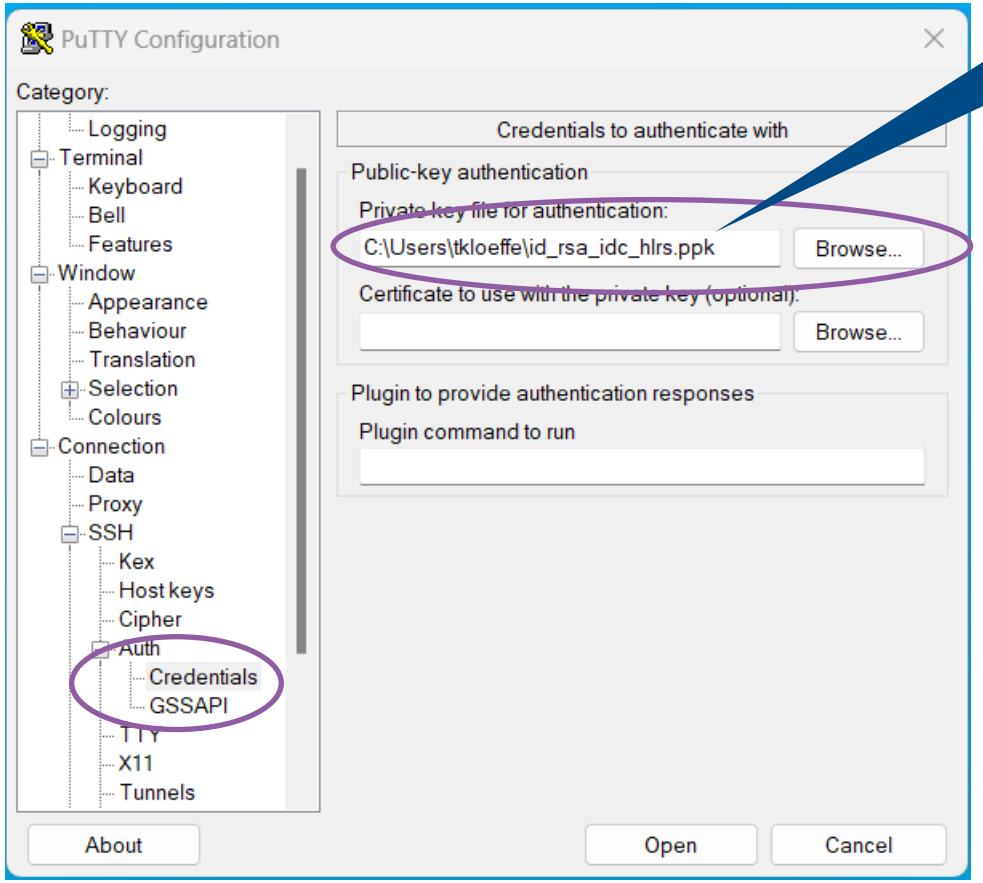
sdp@localhost
Port:4089

Click “Default Settings”
“Load”
Enter “hack1_4089”
“Save”

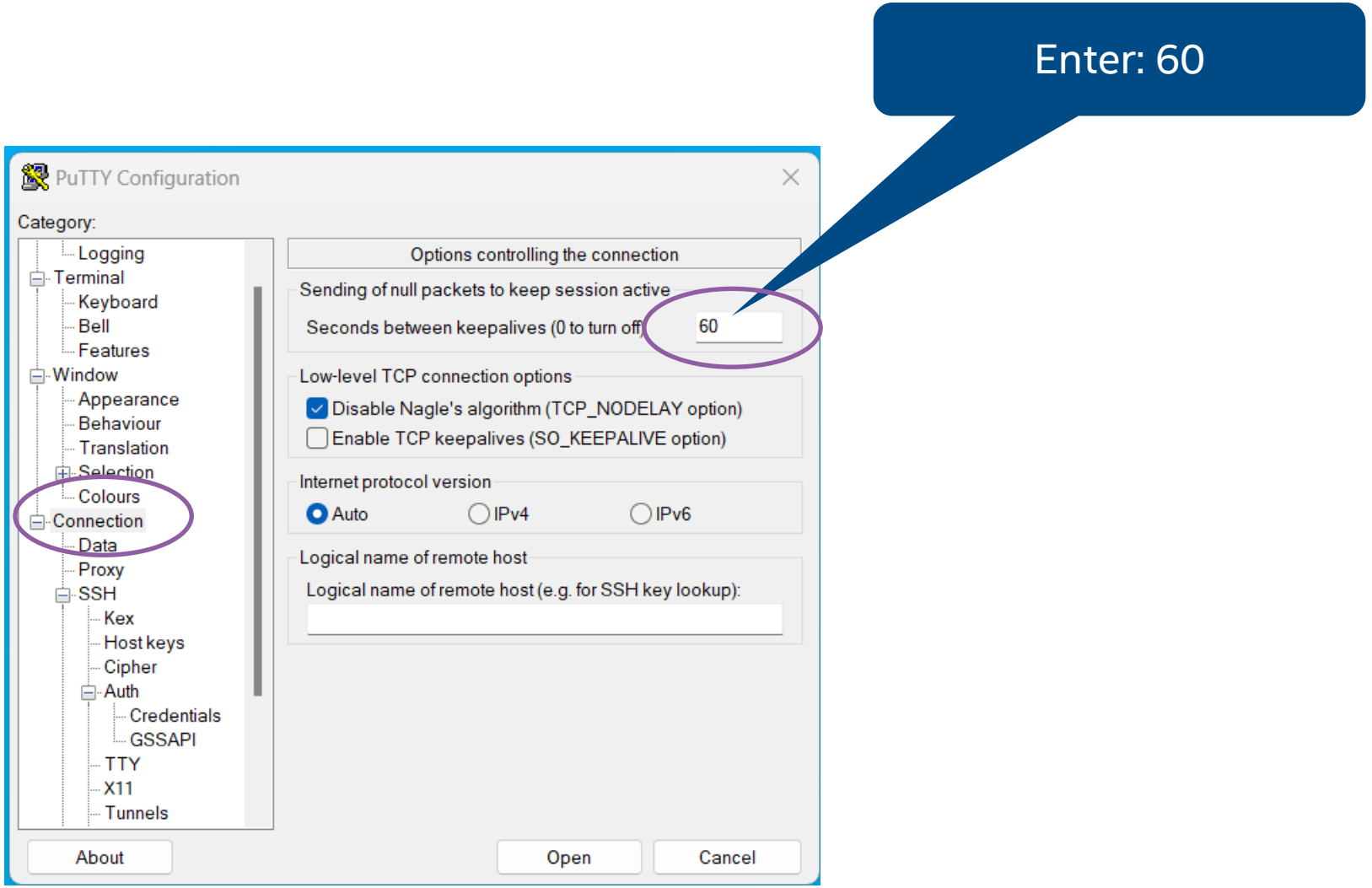


Connection with Putty – Node configuration - 2

Enter your private key again

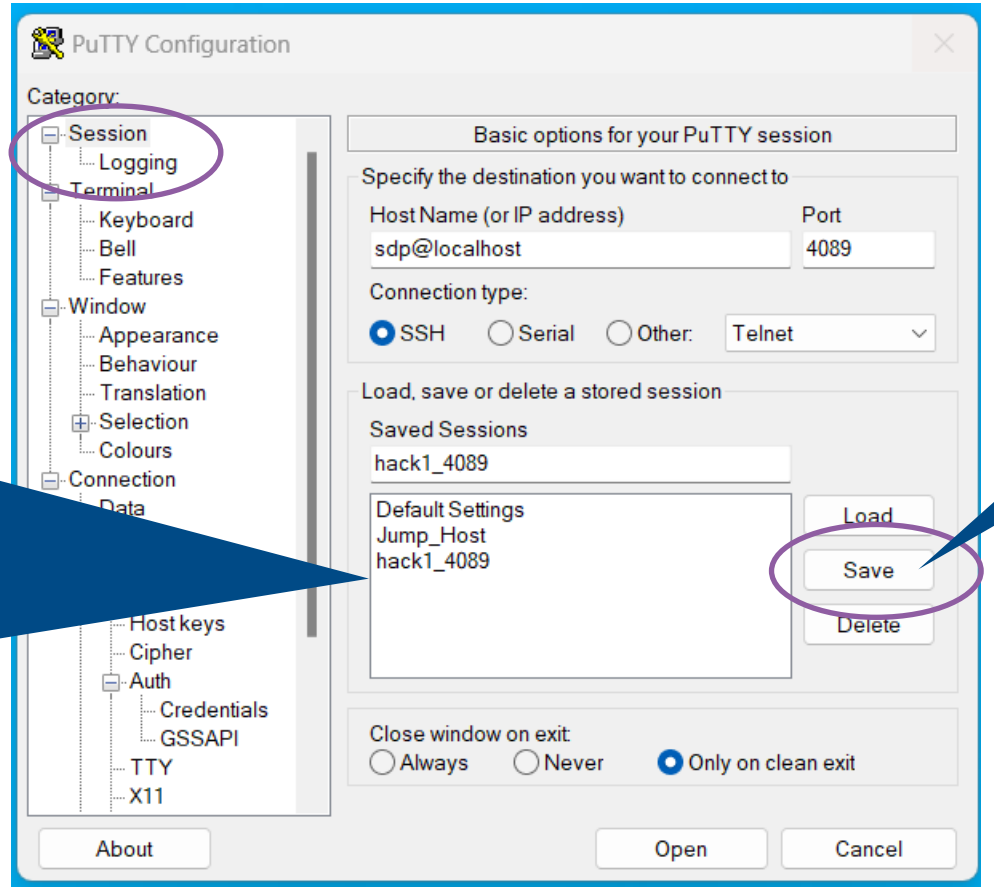


Connection with Putty – Node configuration - 3



Connection with Putty – Node configuration - 4

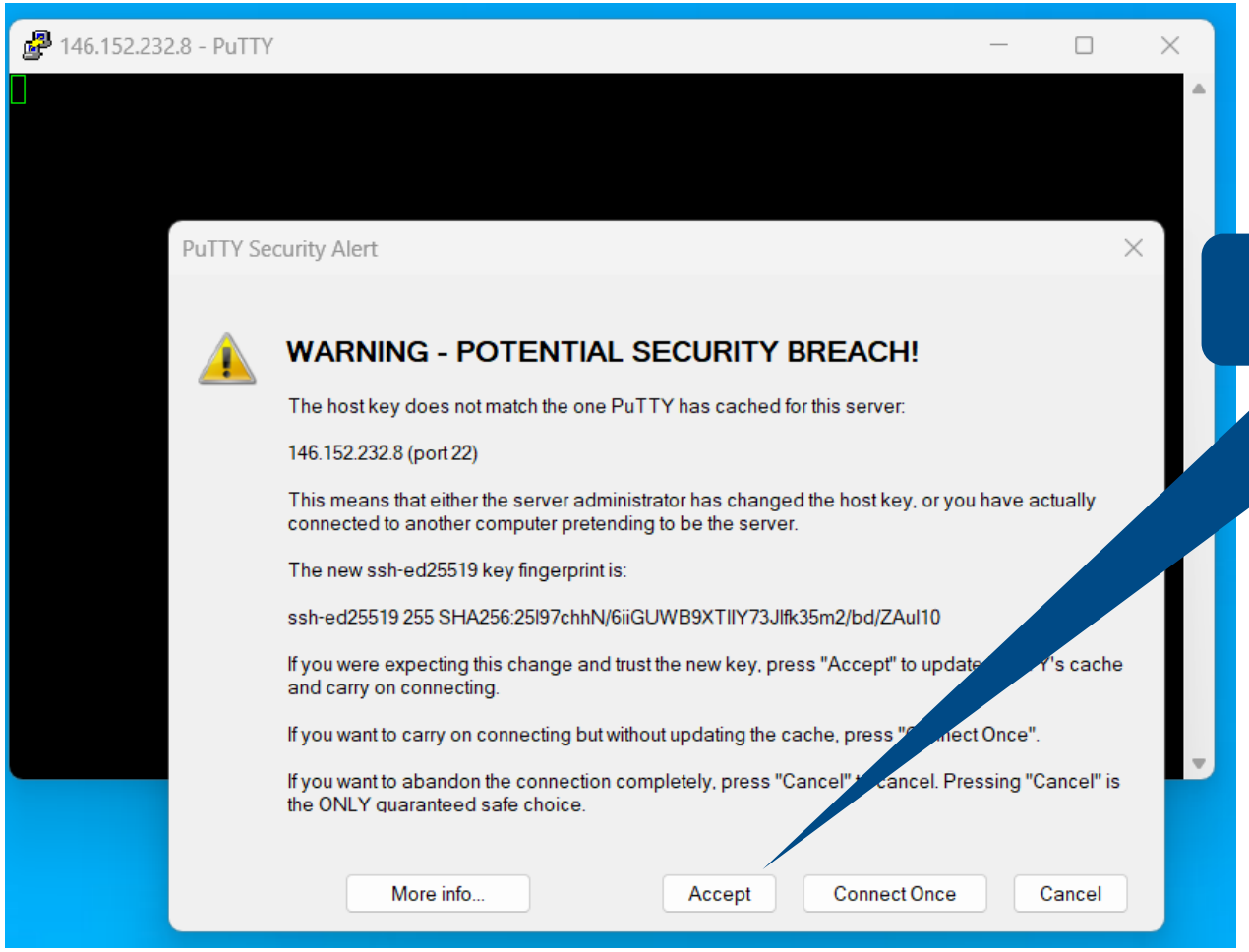
Repeat this for
hack4_4039
Port: 4039
hack3_4212
Port: 4212
hack2_4210
Port: 4210
(Load hack1_4089, change port, change name, click save)



Do not forget to save!

Connection with Putty – Connect to Jump_Host -1

Load Jump_Host
click ‘Open’

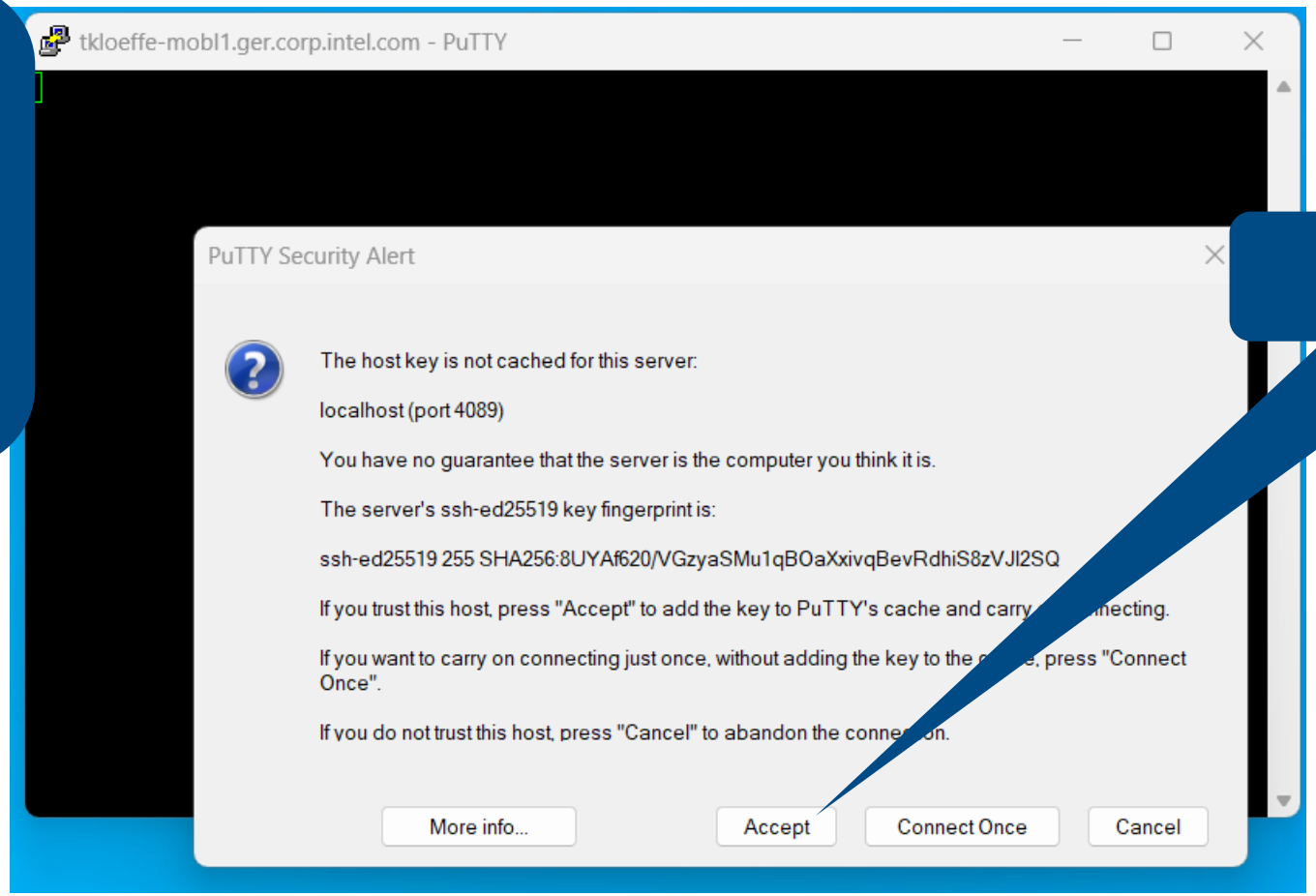


Accept

Connection with Putty – Connect to Jump_Host - 2

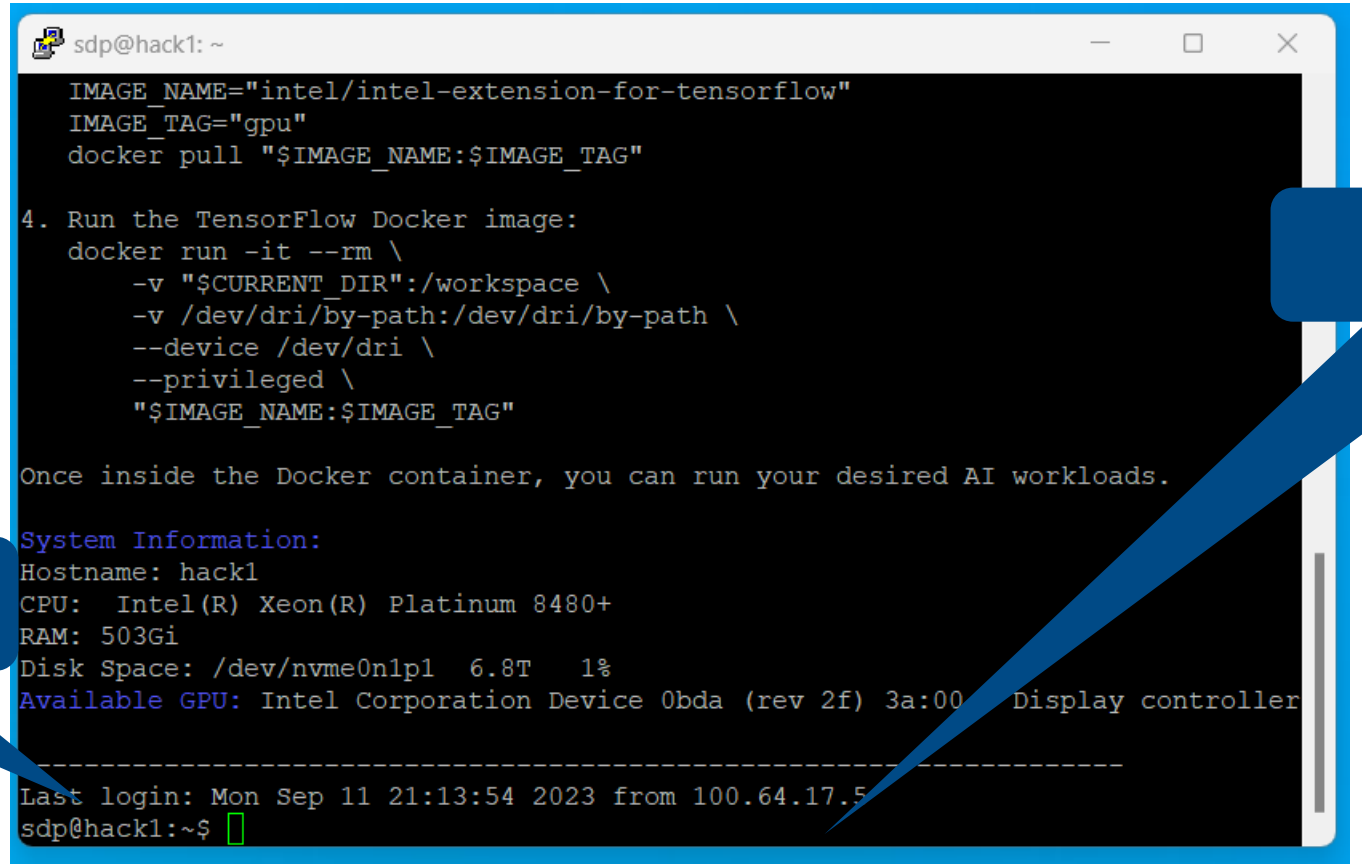
Connection with Putty – Connect to node - 1

Load the node configuration you are assigned to, e.g., hack1_4089



Accept

Connection with Putty – Connect to node - 1



hack1-hack4

Hurray!


intel