

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

The screenshot shows the Intel Developer Cloud website. The browser's address bar contains the URL `intel.com/content/www/us/en/developer/tools/devcloud/services.html`. The navigation menu includes `PRODUCTS`, `SUPPORT`, `SOLUTIONS`, `DEVELOPERS`, and `PARTNERS`. The main heading is `Intel® Developer Cloud`, followed by the text: `Intel® Developer Cloud is a service platform for developing and running workloads in Intel®-optimized deployment environments with the latest Intel® processors and performance-optimized software stacks.` Below this is the text `No Downloads • No Installations`.

The section `Everything You Need to Get Up and Running Faster` includes the text: `The Intel® Developer Cloud makes it easier to access the latest Intel® hardware and software with no setups, configurations, installations, or downloads required.` and a list of features:

- Ready-to-use deployment and development environments
- Access to host systems with the latest Intel processors
- Comprehensive catalog of Intel software, toolkits, and libraries
- Support resources with sample code and documentation

The `Beta Trial Open Now` section includes the text: `The Intel Developer Cloud trial is open to prequalified Intel customers and approved developers. To get started:` and a list of steps:

- Sign up for a standard Intel® Developer Zone account.
- Explore the [service catalog](#).
- Schedule and deploy the service with [Intel® Developer Cloud management console](#).

At the bottom right, there are three links: `Sign Up` (a blue button), `Sign In`, and `Request Intel® Developer Cloud Beta User Support`.

Annotations include a blue box with `cloud.intel.com` pointing to the address bar, a blue box with `Create Developer account` pointing to the `Sign Up` button, and a blue box with `Sign in` pointing to the `Sign In` link.

cloud.intel.com

Create Developer account

Sign in

Select your Hardware – second line is PVC

Intel Developer Cloud Beta Home Instances Get started Support

Launch Instance View Instances

Select an Instance and Click Launch

- Beta - Intel® Trust Domain Extensions (Intel® TDX) with 4th Generation Intel® Xeon® Scalable processors
- Intel® Max Series GPU (PVC) on 4th Gen Intel® Xeon® processors - 1100 series [4x] [Batch Processing/Scheduled access]
- 4th Generation Intel® Xeon® Scalable processors
- Intel® Data Center GPU Max 1550 (four GPUs) with 4th Generation Intel® Xeon® Scalable processors
- Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) HBM-Only mode
- Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) – Flat mode
- Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) – Cache mode
- Intel® Data Center GPU Flex 170 (single GPU) with 4th Generation Intel® Xeon® Scalable processors
- Intel® Data Center GPU Max 1100 with 4th Generation Intel® Xeon® Scalable processors
- Intel® Data Center GPU Flex 170 (single GPUs) with 3rd Generation Intel® Xeon® Scalable Processors
- Habana® Gaudi2 Deep Learning Server featuring eight Gaudi2 HL-225H mezzanine cards and latest Intel® Xeon® Processors
- Intel® Data Center GPU Flex 170 (three GPUs) with 3rd Generation Intel® Xeon® Scalable Processors
- Tiny Virtual Machine - 4th Generation Intel® Xeon® Scalable processors
- Small Virtual Machine - 4th Generation Intel® Xeon® Scalable processors
- Medium Virtual Machine - 4th Generation Intel® Xeon® Scalable processors
- Large Virtual Machine - 4th Generation Intel® Xeon® Scalable processors

Additional Information:

Operating System: Ubuntu 22.04.x	Memory Speed: 4000 MT/s
CPU ID: 0x806F2	DIMM Size: 32GB
Accelerator: Intel® Data Center GPU formerly known as Ponte Vecchio	DIMM Count: 8
Cost: 0.00 USD per hour	Storage: 2TB

Launch Instance

Select PVC batch

Start instance

Copy ssh line

The screenshot shows the Intel Developer Cloud interface. A modal window titled "Instance ID: 64" is open, displaying the following information:

- Access:** 20 days initial access and 7 days extensions
- Instance Type:** Intel® Max Series GPU [PVC] on 4th Gen Intel® Xeon® processors - 1100 series (4x) (Batch Processing/Scheduled access)
- Expiration:** You have access until 2023-09-04 04:00 AM. Your access will be automatically extended if you login to batch server, during the last 5 days before expiration date.
- How to Access/Connect:** [Click here to get instructions](#)
- Connection Info:** `ssh u100154@idcbetabatch.eglb.intel.com` (An arrow points from a blue callout box to this line.)
- Agreement:** Intel® Developer Cloud Access Agreement: [Click here to read Intel® Developer Cloud Access Agreement](#). Below this is a radio button labeled "I agree" (An arrow points from a blue callout box to this button.)
- Close** button

Below the modal, the "Additional Information" section is visible:

Operating System: Ubuntu 22.04.x	Memory Speed: 4000 MT/s
CPU ID: 0x806F2	DIMM Size: 32GB
Accelerator: Intel® Data Center GPU formerly known as Ponte Vecchio	DIMM Count: 8
Cost: 0.00 USD per hour	Storage: 2TB

A "Launch Instance" button is located at the bottom right of the page.

Copy ssh line

Agree to access

```
hbockhor@hbockhor-mobl: - x + v
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!
```

Wsl example

Run interactively

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

```
hbockhor@hbockhor-mobl: ~  
u100154@idc-beta-batch-pvc-node-05:~$ sycl-ls  
Warning: ONEAPI_DEVICE_SELECTOR environment variable is set to opencl:cpu;opencl:fpga;level_zero:1.  
To see the correct device id, please unset ONEAPI_DEVICE_SELECTOR.  
  
[opencl:acc:0] Intel(R) FPGA Emulation Platform for OpenCL(TM), Intel(R) FPGA Emulation Device 1.2 [2023.16.7.0.21_160000]  
[opencl:cpu:1] Intel(R) OpenCL, Intel(R) Xeon(R) Platinum 8480+ 3.0 [2023.16.7.0.21_160000]  
[ext_oneapi_level_zero:gpu:0] Intel(R) Level-Zero, Intel(R) Data Center GPU Max 1100 1.3 [1.3.26516]  
u100154@idc-beta-batch-pvc-node-05:~$
```

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®

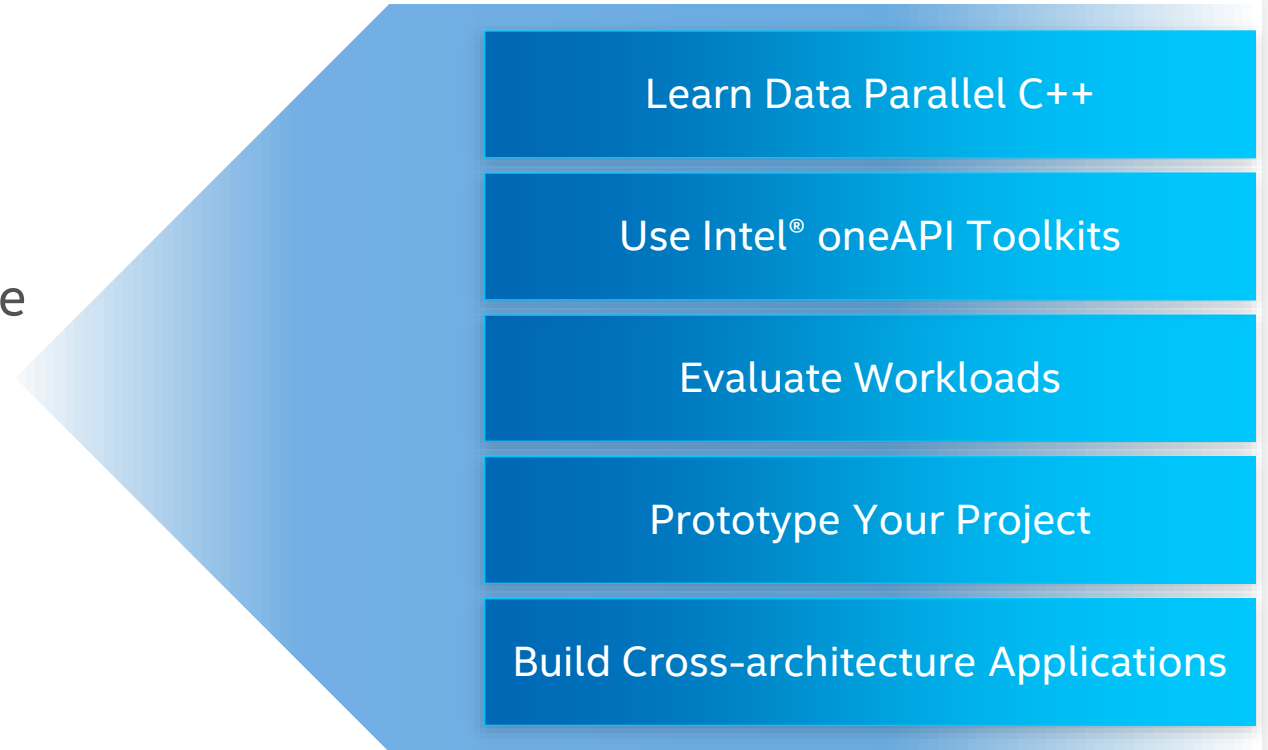
OLD DevCloud

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/>

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 multichitecture 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 X 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

Click here

Overview | Intel® DevCloud

devcloud.intel.com/oneapi/home/

Wetter | Arduino - Home | Minecraft | Intel | Service | MPI | Intel MPI | Gromacs | IPCC | flash | Kaufen | PC Bauen | OneAPI | Programming | Science | Music | Tavel | Other bookmarks

intel PRODUCTS SUPPORT SOLUTIONS DEVELOPERS PARTNERS

Software / Tools / DevCloud / oneAPI

Sign out (u184278)
Expiration Date : 05/23/2024
Request Extension

Intel® DevCloud for oneAPI

Overview Get Started Documentation Forum

Announcements

VIEW ALL ANNOUNCEMENTS >

- > Feb 3, 2023 [New! Intel® AI Analytics Toolkit 2023.1 is now publicly available](#) — We're happy to announce availability of Intel® AI Analytics Toolkit 2023.1 in the Intel® Developer Cloud for oneAPI Projects. This release now br...
- > Jan 11, 2023 [4th Gen Intel® Xeon® Scalable Processors are now publicly available](#) — We're happy to announce availability of 4th Gen Intel® Xeon® Scalable Processors in the Intel® Developer Cloud for oneAPI Projects. This la...
- > Dec 16, 2023 [Intel® oneAPI 2023 Tools are now available in the Intel® Developer Cloud for oneAPI Projects](#) — We're happy to announce Intel® oneAPI 2023 Tools availability in the Intel® Developer Cloud for oneAPI Projects. T...

Test Performance on CPU, GPU, and FPGA Architectures

CPU:

- Intel® Xeon® Scalable 6128 processors
- Intel® Xeon® Scalable 8256 processors
- Intel® Xeon® E-2176 P630 processors (with Intel® Graphics Technology)

GPU:

- Intel® Xeon® E-2176 P630 processors (with Intel® Graphics Technology)
- Intel® Iris® Xe MAX

FPGA:

- Intel® Arria® 10 FPGAs
- Intel® Stratix® 10 FPGAs

What You Get

- Free access to Intel® oneAPI toolkits and components and the latest Intel® hardware
- 220 GB of file storage
- 192 GB RAM
- 120 days of access (extensions available)
- Terminal Interface (Linux®)
- Microsoft Visual Studio® Code integration
- Remote Desktop for Intel® oneAPI Rendering Toolkit

Why oneAPI?

- Freedom of choice for accelerated computing across multiple architectures: CPU, GPU, and FPGA
- An open alternative to proprietary lock-in
- SYCL®, a standards-based abstraction layer for programming heterogeneous and offload processors using modern ISO C++
- Optimized libraries for API-based programming
- Advanced analysis and debug tools
- CUDA® source code migration
- Additional support for OpenCL and RTL development on FPGA nodes

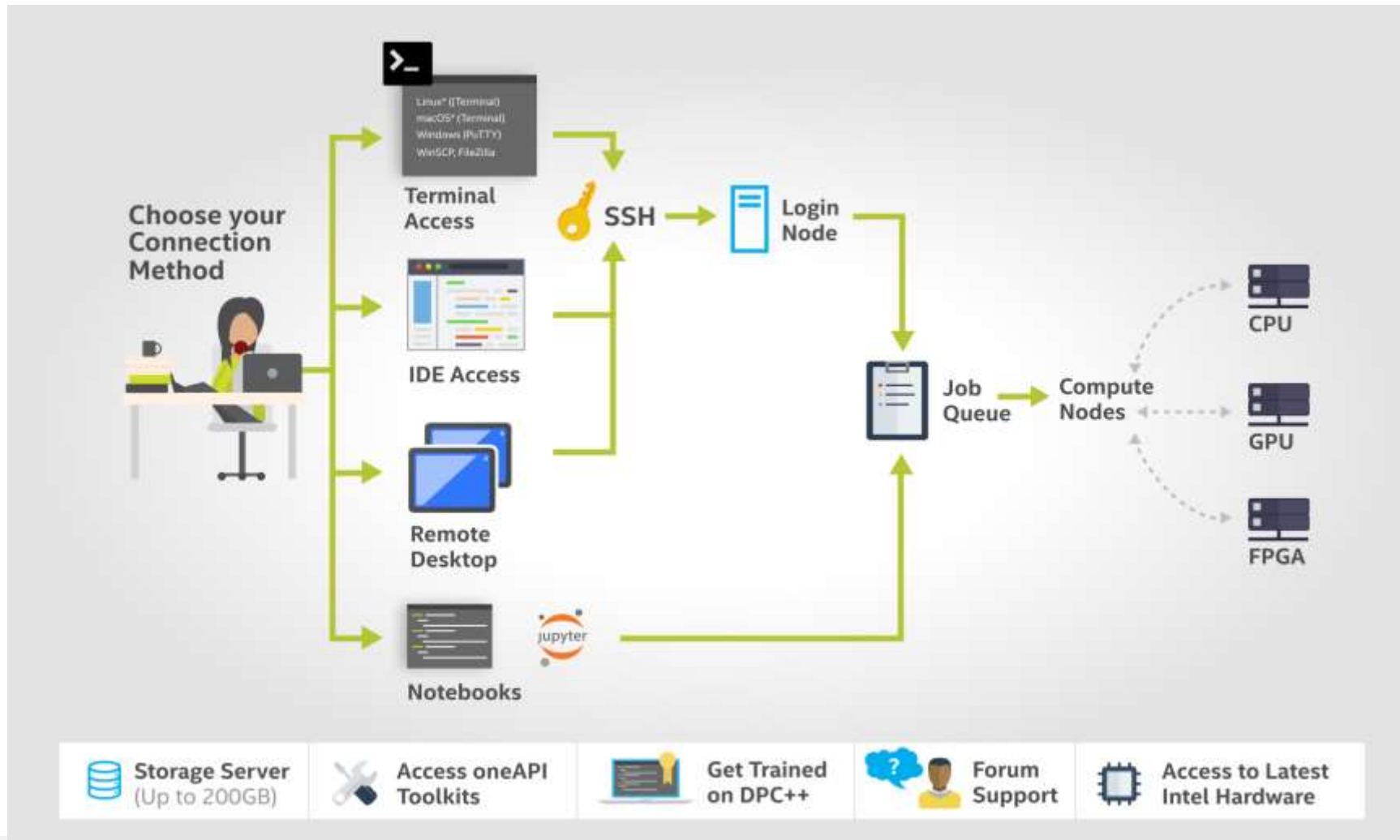
User name
and
Expiration
Date

View
Documentation

View: Connect to DevCloud

The screenshot shows a web browser window displaying the Intel DevCloud documentation page. The URL is `devcloud.intel.com/oneapi/documentation/connect-with-ssh-linux-macos/`. The page features a blue header with the Intel logo and navigation links for PRODUCTS, SUPPORT, SOLUTIONS, DEVELOPERS, and PARTNERS. Below the header, there are breadcrumb links: Software / Tools / DevCloud / oneAPI. The main content area is titled "Intel® DevCloud for oneAPI" and includes sub-links for Overview, Get Started, Documentation, and Forum. A search bar is located on the left side. The main content area is divided into two columns. The left column contains a search bar and a list of links: "Connect to the DevCloud", "Download & Configure Third Party Dependencies", "Connect from Linux/macOS", "Connect from Windows with DpenSSH (recommended)", "Connect from Windows with Cygwin (deprecated)", "Using VSCode", and "How to use the DevCloud". The right column contains the main text of the page, which starts with "Overview" and "Documentation" breadcrumbs, followed by the title "Connect from Linux/macOS using an SSH Client". The text explains that users can access the cluster using SSH and provides two options: "Option 1: Automated Configuration" and "Option 2: Manual Configuration". The footer of the page includes links for Company Information, Our Commitment, Communities, Investor Relations, Contact Us, Newsroom, and Jobs.

Connection Methods



Get Started (ssh)

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

1 Connect to DevCloud

Connect to the DevCloud using SSH Clients.

2 Hello World! Get Started by running a simple sample on DevCloud.

Use this simple sample to confirm that you are connected to oneAPI DevCloud.

2.1. CPU/GPU Vector-Add sample walkthrough

1. Connect to the DevCloud.

```
[myname@myhomecomputer] $ | ssh devcloud
```

2. Download the samples.

```
[u115975@login-2] $ | git clone https://github.com/oneapi-src/oneAPI-samples.git
```

3. Go to the vector-add sample.

```
[u115975@login-2] $ | cd oneAPI-samples/DirectProgramming/DPC++/DenseLinearAlgebra/vector-add/
```

Build and run the sample in batch mode

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]

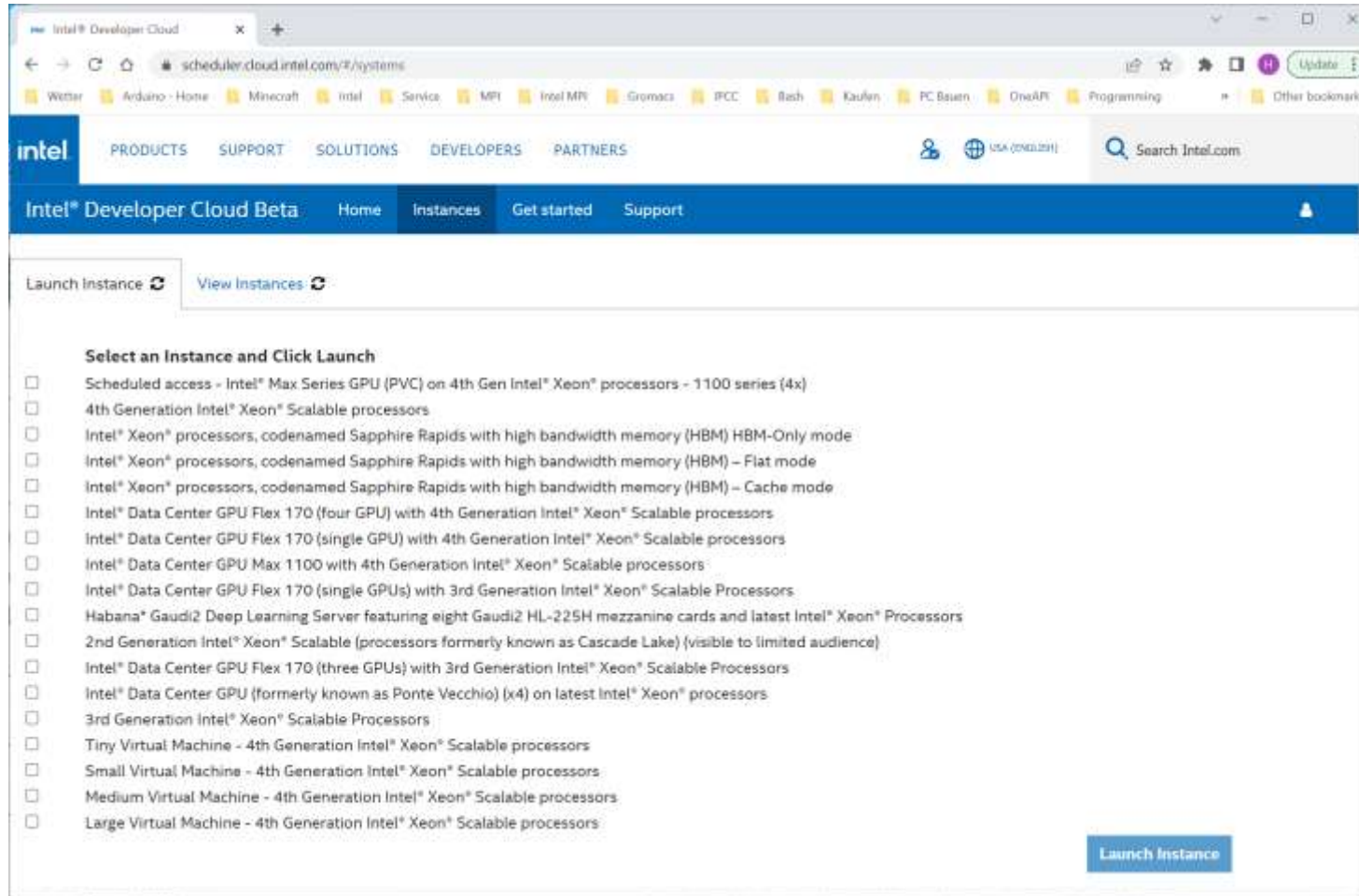
`custom_selector(gpu)` : gpu, Intel(R) Level-Zero, Intel(R) UHD Graphics [0x9a60] 1.3 [1.3.24595]

`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



The screenshot shows the Intel Developer Cloud Beta website interface. The browser address bar displays 'scheduler.cloud.intel.com/#/systems'. The navigation menu includes 'PRODUCTS', 'SUPPORT', 'SOLUTIONS', 'DEVELOPERS', and 'PARTNERS'. The main navigation bar features 'Intel® Developer Cloud Beta', 'Home', 'Instances', 'Get started', and 'Support'. Below the navigation bar, there are two buttons: 'Launch Instance' and 'View Instances'. The main content area is titled 'Select an Instance and Click Launch' and contains a list of instance options, each with a checkbox. The options include various processor and GPU configurations, such as 'Scheduled access - Intel® Max Series GPU (PVC) on 4th Gen Intel® Xeon® processors - 1100 series (4x)', '4th Generation Intel® Xeon® Scalable processors', 'Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) HBM-Only mode', 'Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) – Flat mode', 'Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) – Cache mode', 'Intel® Data Center GPU Flex 170 (four GPU) with 4th Generation Intel® Xeon® Scalable processors', 'Intel® Data Center GPU Flex 170 (single GPU) with 4th Generation Intel® Xeon® Scalable processors', 'Intel® Data Center GPU Max 1100 with 4th Generation Intel® Xeon® Scalable processors', 'Intel® Data Center GPU Flex 170 (single GPUs) with 3rd Generation Intel® Xeon® Scalable Processors', 'Habana® Gaudi2 Deep Learning Server featuring eight Gaudi2 HL-225H mezzanine cards and latest Intel® Xeon® Processors', '2nd Generation Intel® Xeon® Scalable (processors formerly known as Cascade Lake) (visible to limited audience)', 'Intel® Data Center GPU Flex 170 (three GPUs) with 3rd Generation Intel® Xeon® Scalable Processors', 'Intel® Data Center GPU (formerly known as Ponte Vecchio) (x4) on latest Intel® Xeon® processors', '3rd Generation Intel® Xeon® Scalable Processors', 'Tiny Virtual Machine - 4th Generation Intel® Xeon® Scalable processors', 'Small Virtual Machine - 4th Generation Intel® Xeon® Scalable processors', 'Medium Virtual Machine - 4th Generation Intel® Xeon® Scalable processors', and 'Large Virtual Machine - 4th Generation Intel® Xeon® Scalable processors'. A 'Launch Instance' button is located at the bottom right of the list.

Launch Instance [View Instances](#)

Select an Instance and Click Launch

- Scheduled access - Intel® Max Series GPU (PVC) on 4th Gen Intel® Xeon® processors - 1100 series (4x)
- 4th Generation Intel® Xeon® Scalable processors
- Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) HBM-Only mode
- Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) – Flat mode
- Intel® Xeon® processors, codenamed Sapphire Rapids with high bandwidth memory (HBM) – Cache mode
- Intel® Data Center GPU Flex 170 (four GPU) with 4th Generation Intel® Xeon® Scalable processors
- Intel® Data Center GPU Flex 170 (single GPU) with 4th Generation Intel® Xeon® Scalable processors
- Intel® Data Center GPU Max 1100 with 4th Generation Intel® Xeon® Scalable processors
- Intel® Data Center GPU Flex 170 (single GPUs) with 3rd Generation Intel® Xeon® Scalable Processors
- Habana® Gaudi2 Deep Learning Server featuring eight Gaudi2 HL-225H mezzanine cards and latest Intel® Xeon® Processors
- 2nd Generation Intel® Xeon® Scalable (processors formerly known as Cascade Lake) (visible to limited audience)
- Intel® Data Center GPU Flex 170 (three GPUs) with 3rd Generation Intel® Xeon® Scalable Processors
- Intel® Data Center GPU (formerly known as Ponte Vecchio) (x4) on latest Intel® Xeon® processors
- 3rd Generation Intel® Xeon® Scalable Processors
- Tiny Virtual Machine - 4th Generation Intel® Xeon® Scalable processors
- Small Virtual Machine - 4th Generation Intel® Xeon® Scalable processors
- Medium Virtual Machine - 4th Generation Intel® Xeon® Scalable processors
- Large Virtual Machine - 4th Generation Intel® Xeon® Scalable processors

Launch Instance

Notices & Disclaimers

Texas Advanced Computing Center (TACC) Frontera references

Article: [HPCWire: Visualization & Filesystem Use Cases Show Value of Large Memory Fat Nodes 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

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*](#)

■ DPC++

Connection with Jupyter* Notebook

- [JupyterLab*](#)

Connect with Jupyter* Lab



Connect with Jupyter* Notebook

Use Jupyter Notebook to learn about how oneAPI can solve the challenges of programming in a heterogeneous world and understand the Data Parallel C++ (DPC++) language and programming model.

[Launch JupyterLab*](#)

- [JupyterLabs*](#) for AI



AI Sample Applications

Find sample applications for your specific market needs with examples of how to optimize, tune, and accelerate your applications.

[Learn More](#)



Connect and Create

Develop your own machine learning solutions using Jupyter* Notebooks or a containerized launch environment. Benchmark your code and optimize it for Intel* hardware.

[Connect to JupyterLab](#)
[Connect to Container Playground](#)

Basic Training Modules in JupyterLab*

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

Learn the Essentials of Data Parallel C++



Module 0 Introduction to JupyterLab* and Notebooks.

Learn to use Jupyter notebooks to modify and run code as part of learning exercises.

[Try it in Jupyter](#)



Module 1 Introduction to DPC++

- Articulate how oneAPI can help to solve the challenges of programming in a heterogeneous world.
- Use oneAPI solutions to enable your workflows.
- Understand the DPC++ language and programming model.
- Become familiar with using Jupyter notebooks for training throughout the course.

[Try it in Jupyter](#)



Module 2 DPC++ Program Structure

- Articulate the SYCL* fundamental classes.



Module 3 DPC++ Unified Shared Memory

- Use new DPC++ features like Unified Shared Memory (USM) to

oneAPI Essentials in JupyterLab*

The screenshot displays the JupyterLab environment. On the left, a file browser shows the directory structure for 'oneAPI_Essentials', including folders for various topics and files like 'Makefile', 'README.md', 'sample.json', and 'Welcome.ipynb'. The main area contains a code cell with the following C++ code:

```
[ ]: %writefile lab/simple.cpp
// =====
// Copyright © 2020 Intel Corporation
// =====
// SPOX-License-Identifier: MIT
// =====
#include <CL/sycl.hpp>
using namespace sycl;
static const int N = 16;
int main(){
    // define queue which has default device associated for offload
    queue q;
    std::cout << "Device: " << q.get_device().get_info<info::device::name>() << "\n";

    // Unified Shared Memory Allocation enables data access on host and device
    int *data = malloc_shared<int>(N, q);

    // Initialization
    for(int i=0; i<N; i++) data[i] = i;

    // Offload parallel computation to device
    q.parallel_for<range<1>(N), [a] {id<1> i}{
        data[i] *= 2;
    }).wait();

    // Print Output
    for(int i=0; i<N; i++) std::cout << data[i] << "\n";

    free(data, q);
    return 0;
}
```

Below the code cell, there is a section titled 'Build and Run' with the instruction: 'Select the cell below and click Run to compile and execute the code above:'. The following shell code is provided:

```
[ ]: ! chmod 755 q; chmod 755 run_simple.sh;if [ -x "$(command -v qsub)" ]; then ./q run_simple.sh; else ./run_si
```

The bottom status bar shows 'Simple' mode, 'No Kernel | Idle', and 'oneAPI_Intro.ipynb'.

Connection with Visual Studio Code*

▲ Connect to the DevCloud

Download & Configure Third Party Dependencies

Connect with Cygwin

Connect with VSCode

Using the Code Sample
Browser for Intel®
oneAPI Toolkit
Extension on DevCloud

Connect with
Linux/macOS SSH

▼ How to use the DevCloud

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.

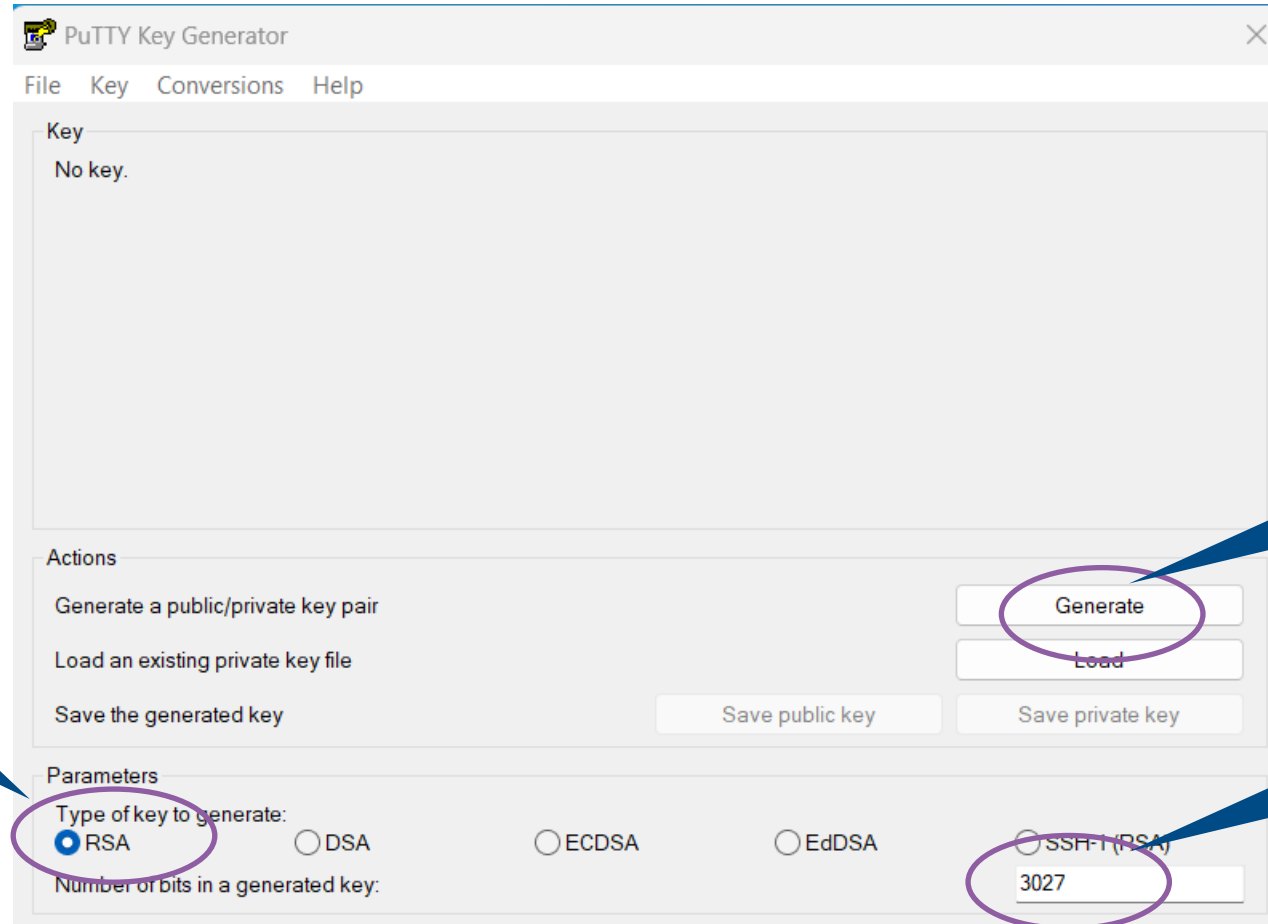
Several Cygwin packages are downloaded during the installation. The script is configured to use mirrors.kernel.org as the default download site. A full list of Cygwin mirror sites can be found on the Cygwin homepage <https://www.cygwin.com/>.

Using putty to access IDC instances

for Windows users

Generate ssh-key with putty-gen - 1

Open "puttygen"

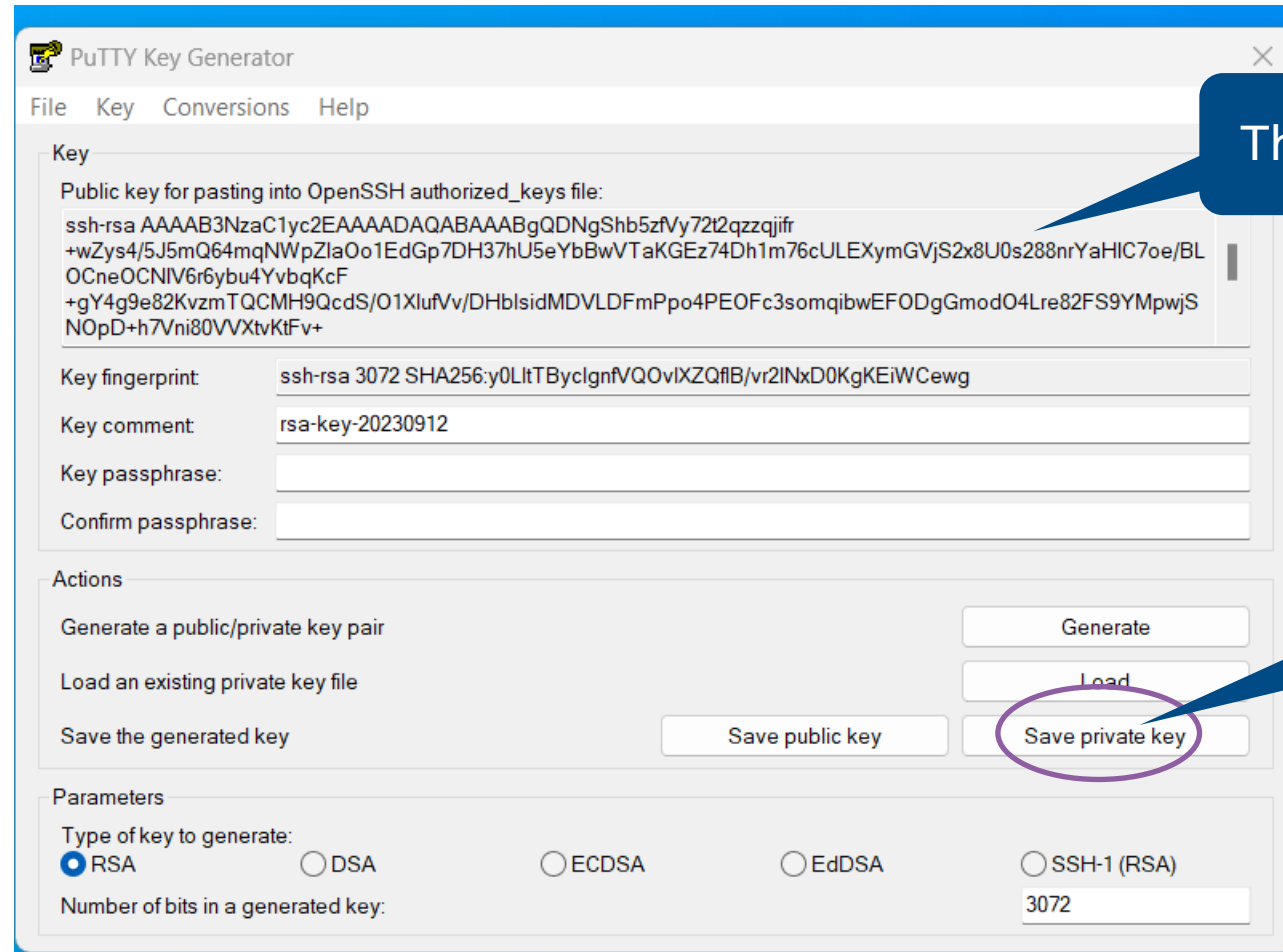


Select RSA

Click 'Generate' and move your mouse

3072

Generate ssh-key with putty-gen - 2

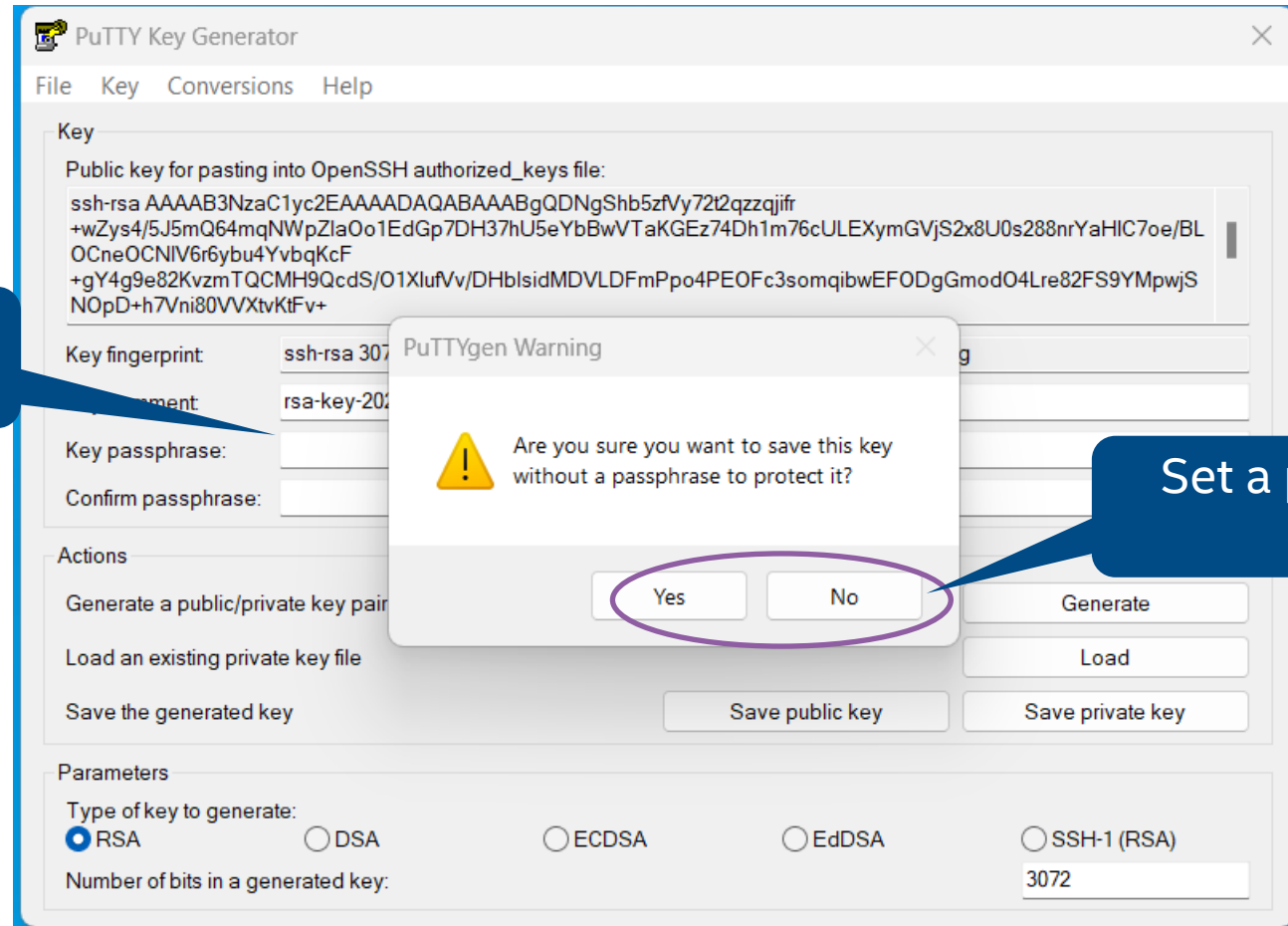


This is your public key

Save private key

Generate ssh-key with putty-gen - 3

If you did not set a passphrase

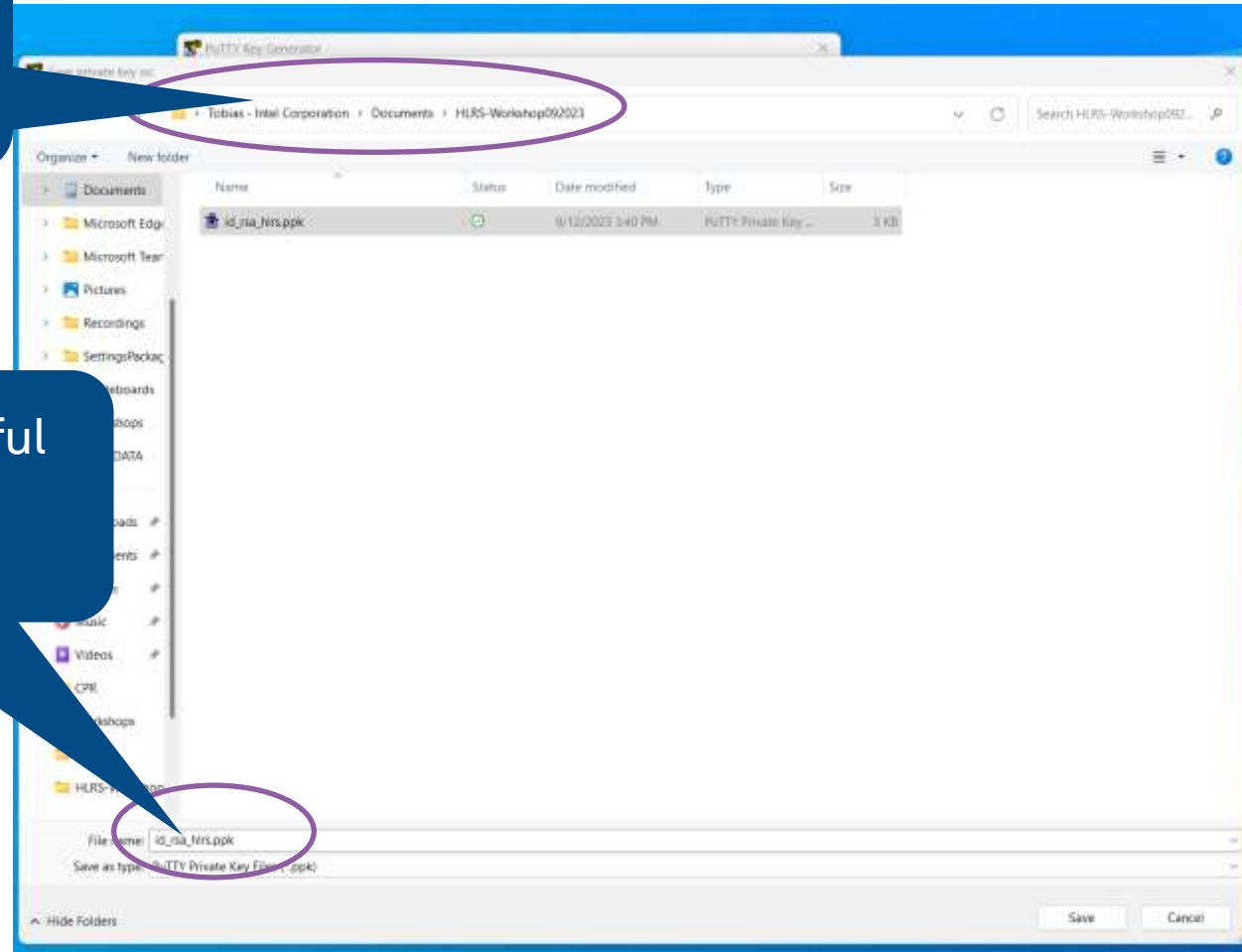


Set a passphrase or click yes

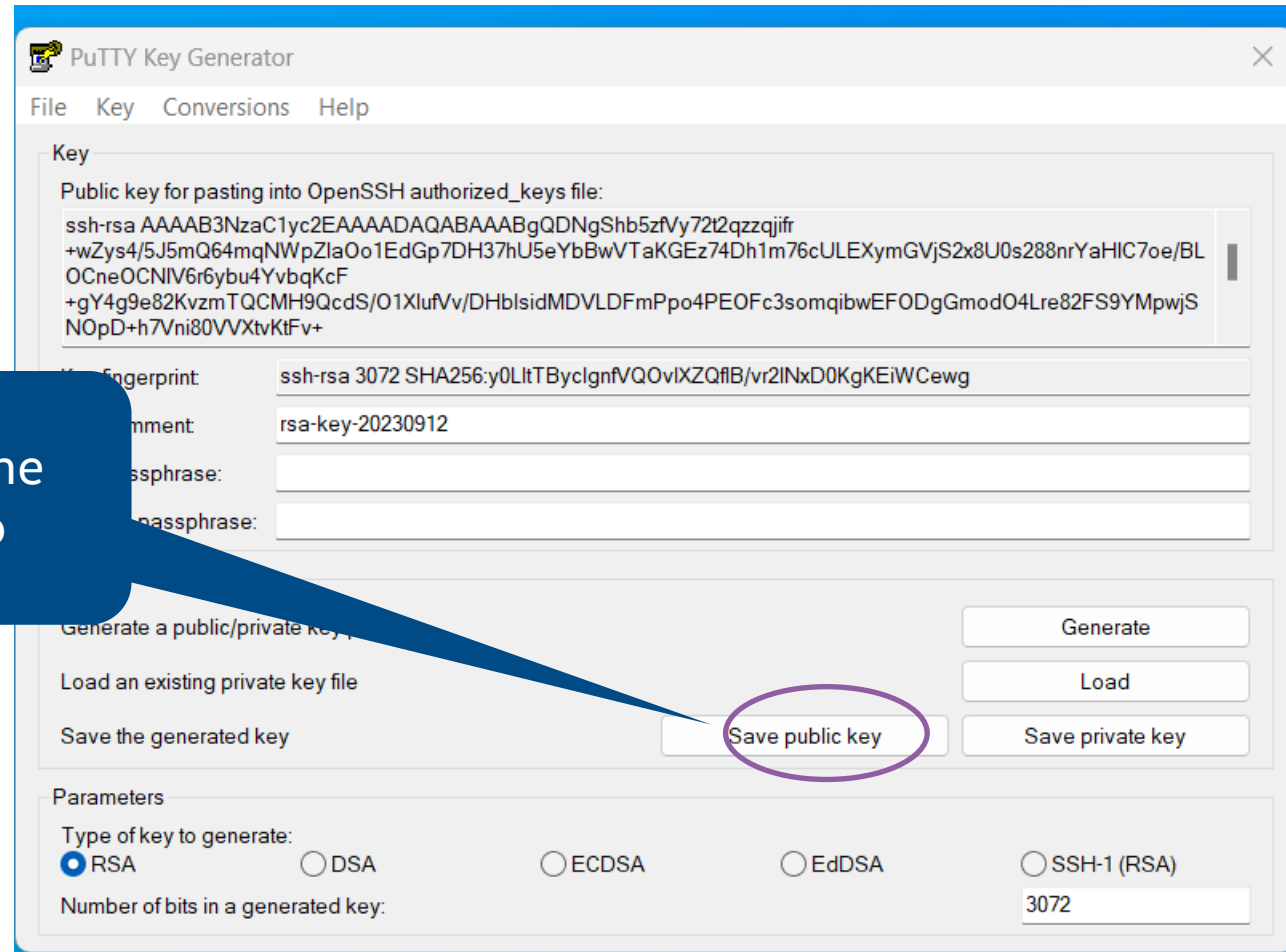
Generate ssh-key with putty-gen - 4

Remember the path
you choose

Select a meaningful
name:
id_rsa_hlrs.ppk



Generate ssh-key with putty-gen - 5

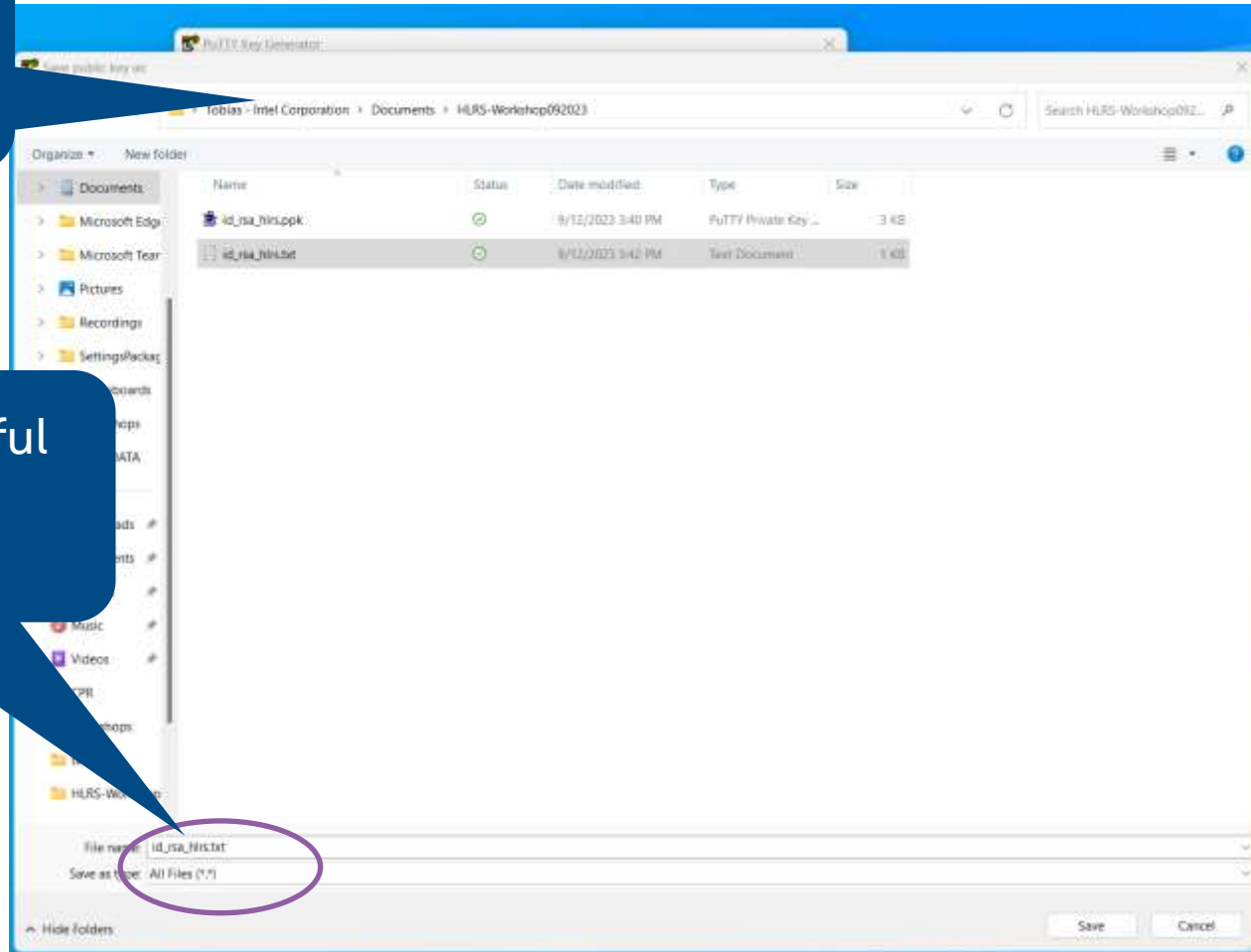


Useful to save the public key, too

Generate ssh-key with putty-gen - 6

Remember the path
you choose

Select a meaningful
name:
id_rsa_hlrs.txt

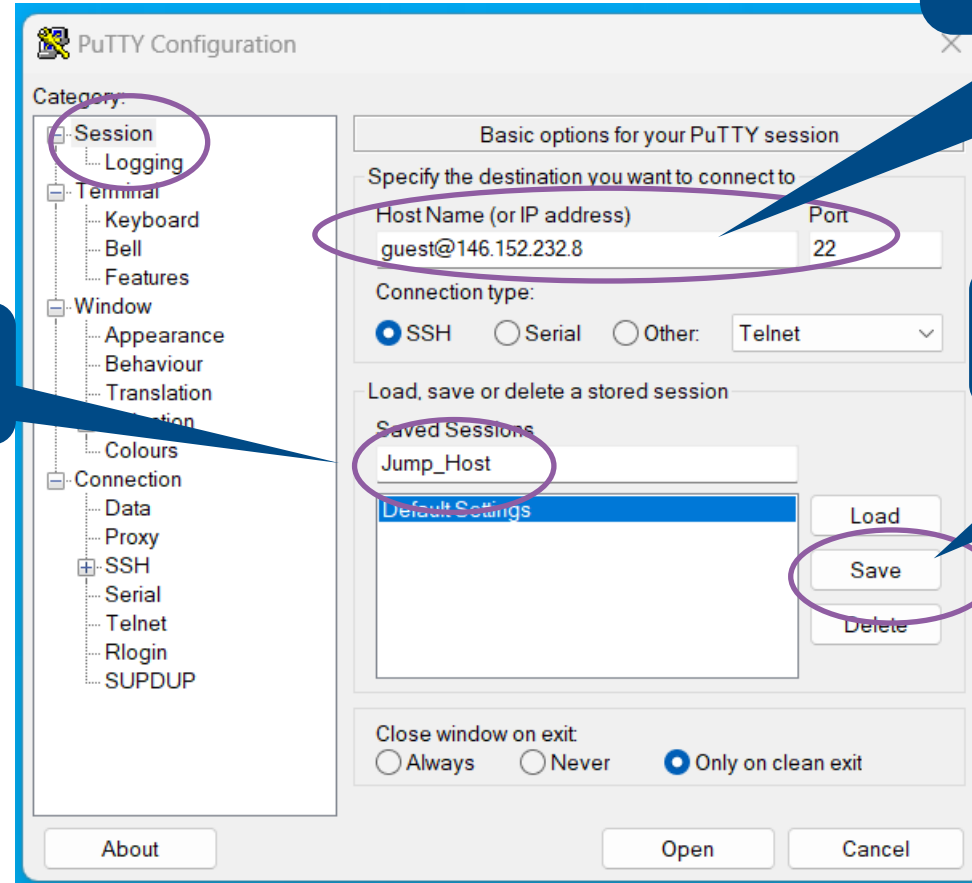


Connection with Putty – JumpHost configuration - 1

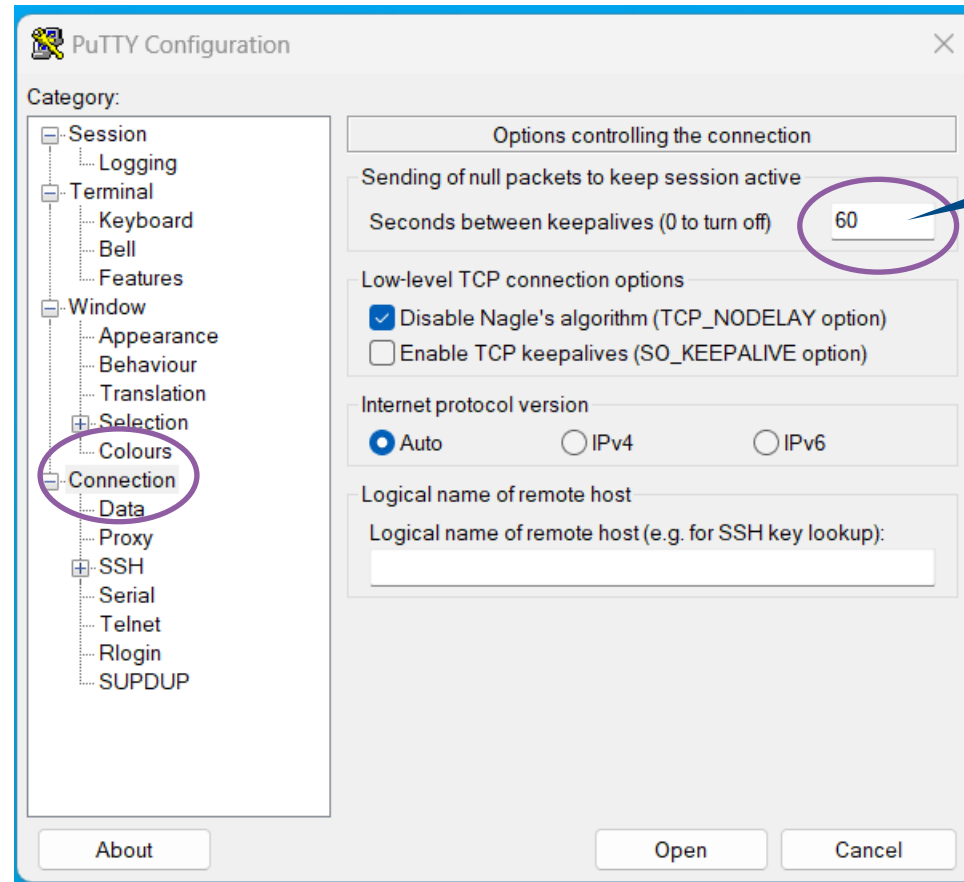
HostName (or IP address)
guest@146.152.232.8
Port: 22

Jump_Host

Do not forget to click save!



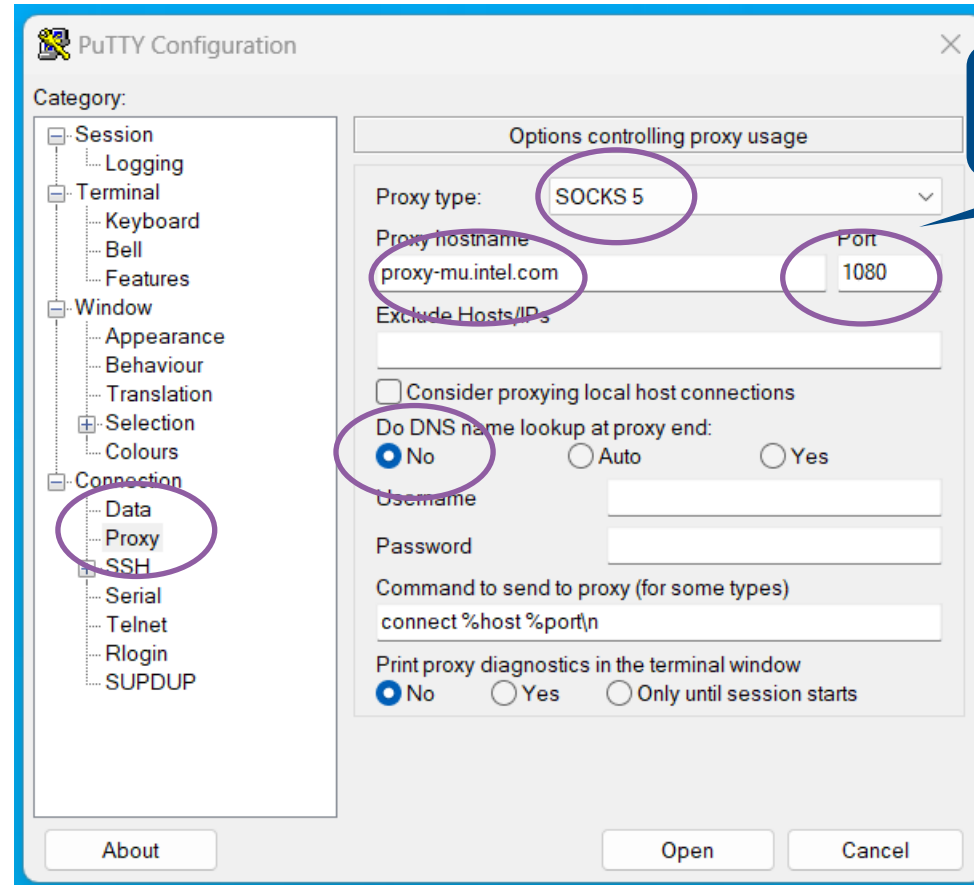
Connection with Putty – JumpHost configuration - 2



Use 60

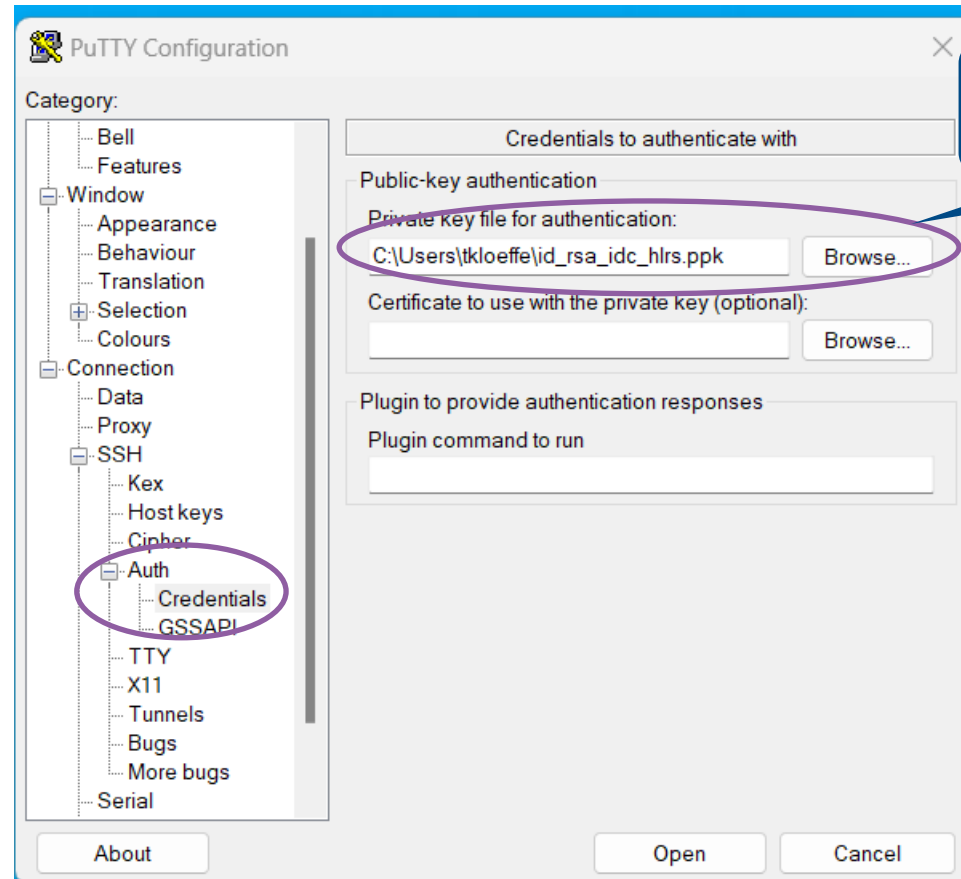
Connection with Putty – JumpHost configuration – 3

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



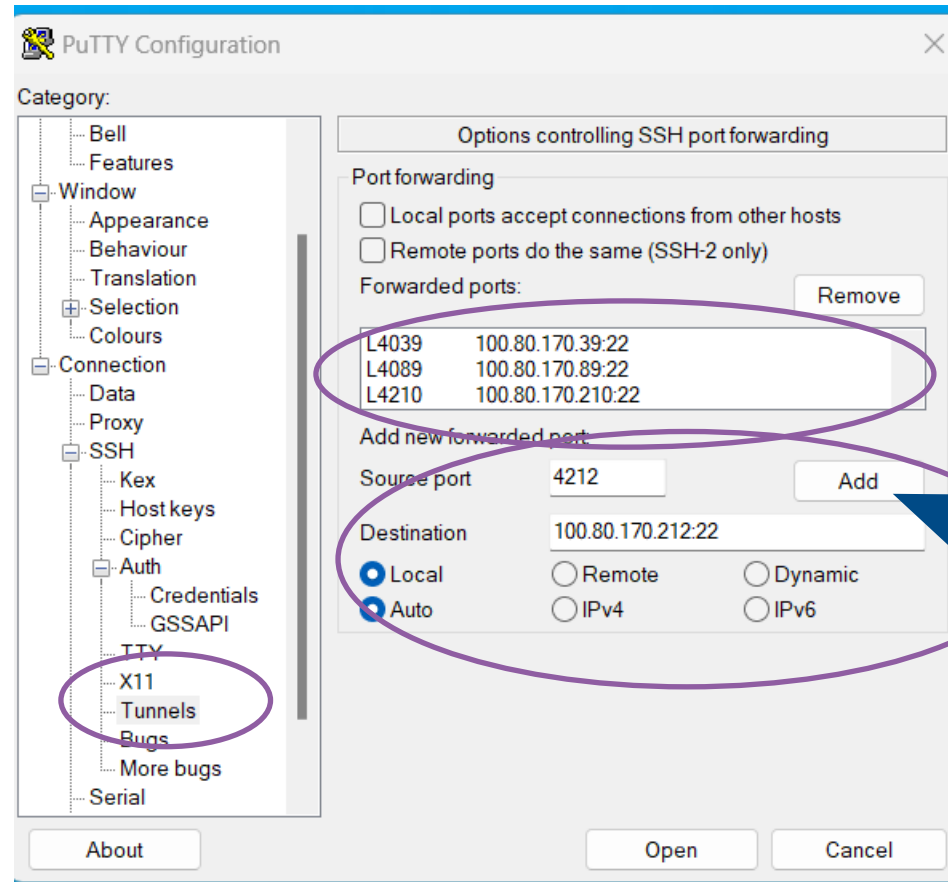
Select appropriate proxy type

Connection with Putty – JumpHost configuration – 4



Select your private key

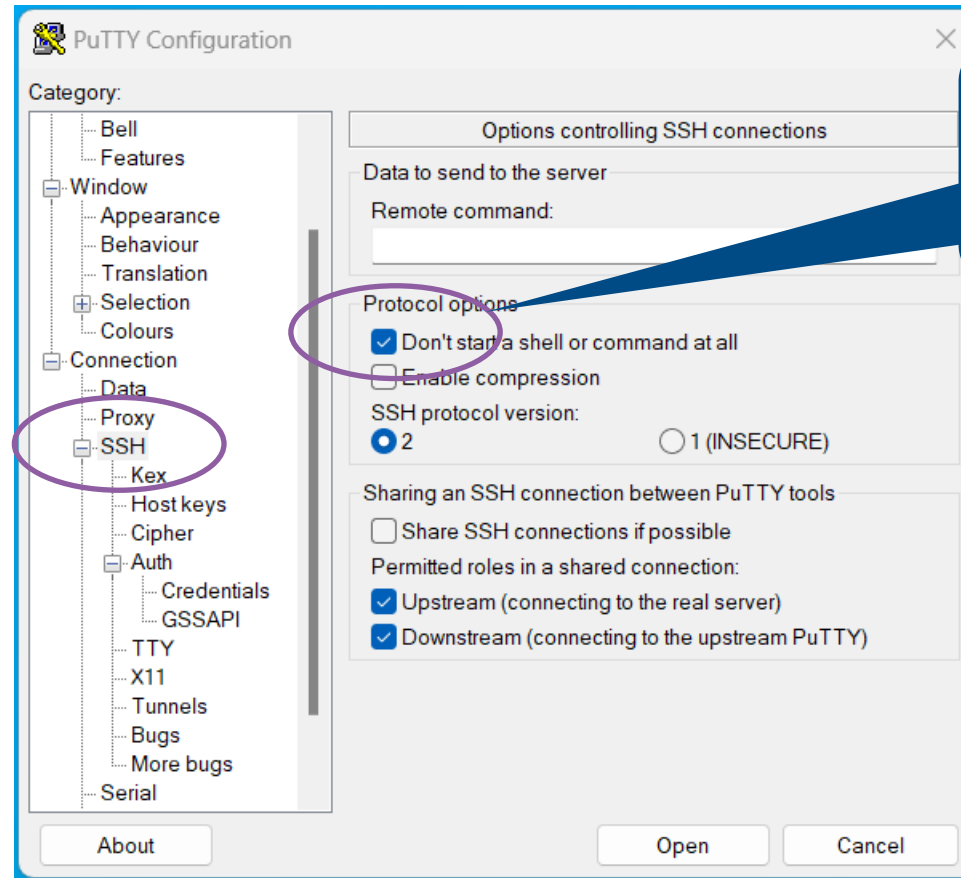
Connection with Putty – JumpHost configuration – 4



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

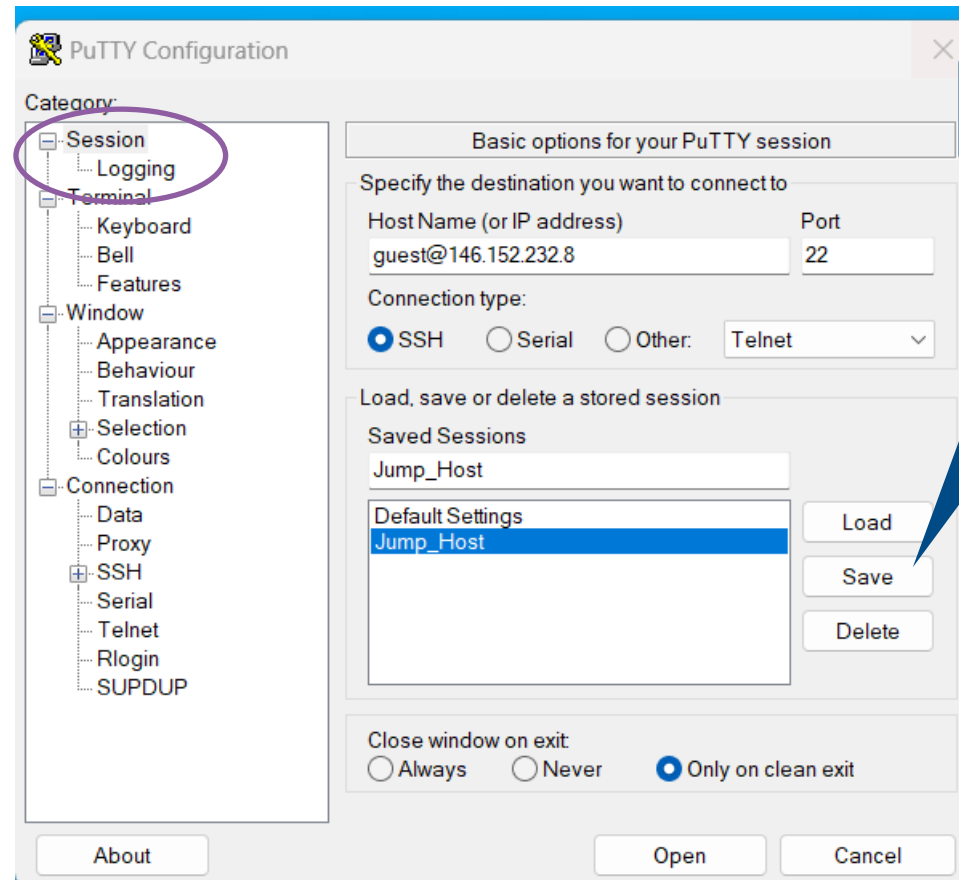
Don't forget to click 'add'

Connection with Putty – JumpHost configuration – 5



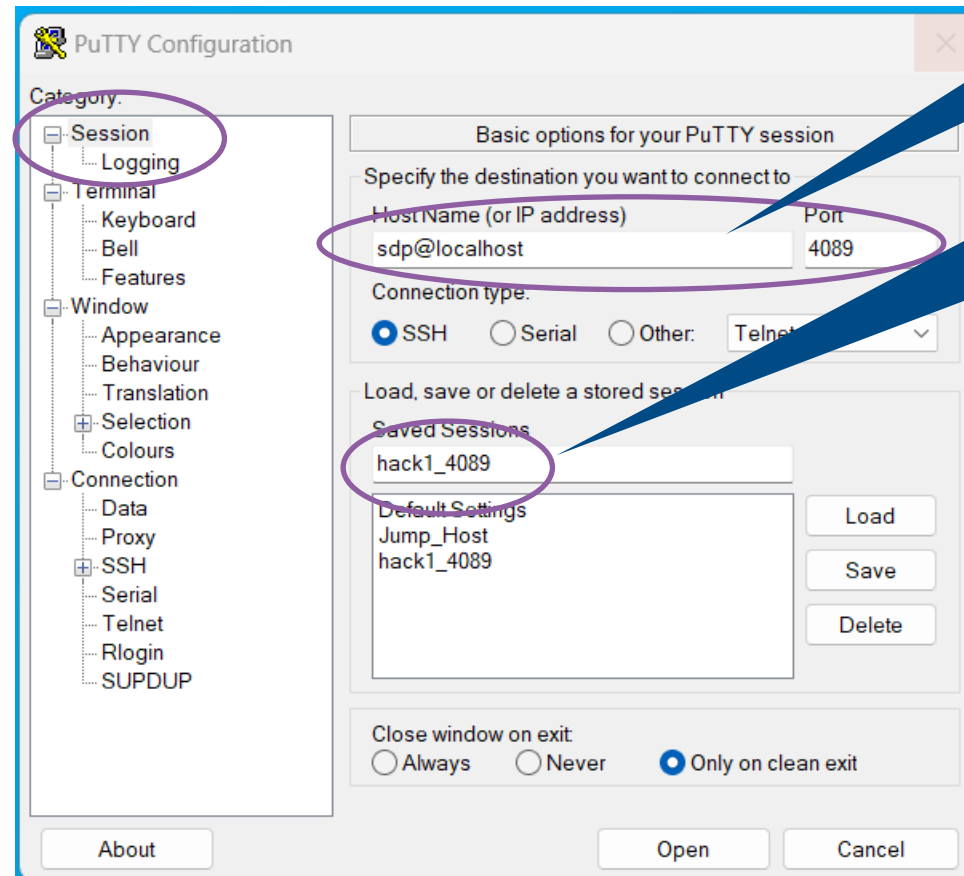
Check “Don’t start a shell or command at all”

Connection with Putty – JumpHost configuration – 6



Do not forget to save!

Connection with Putty – Node configuration - 1

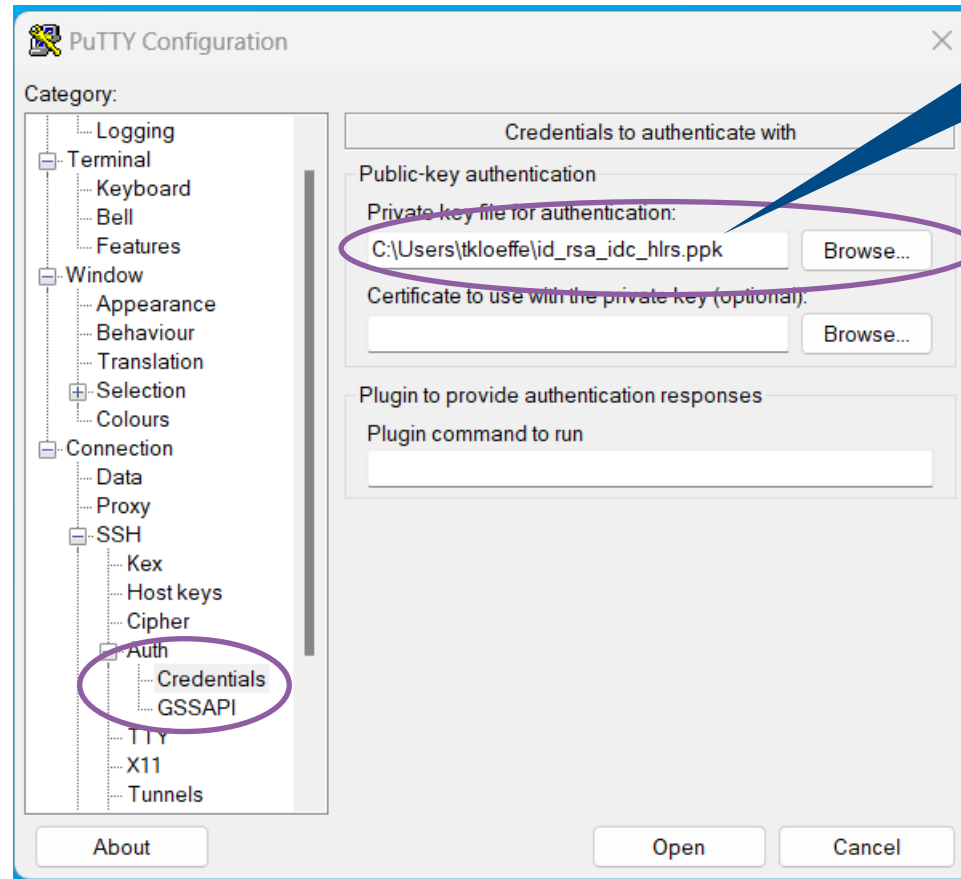


sdp@localhost
Port:4089

Click "Default Settings"
"Load"
Enter "hack1_4089"
"Save"

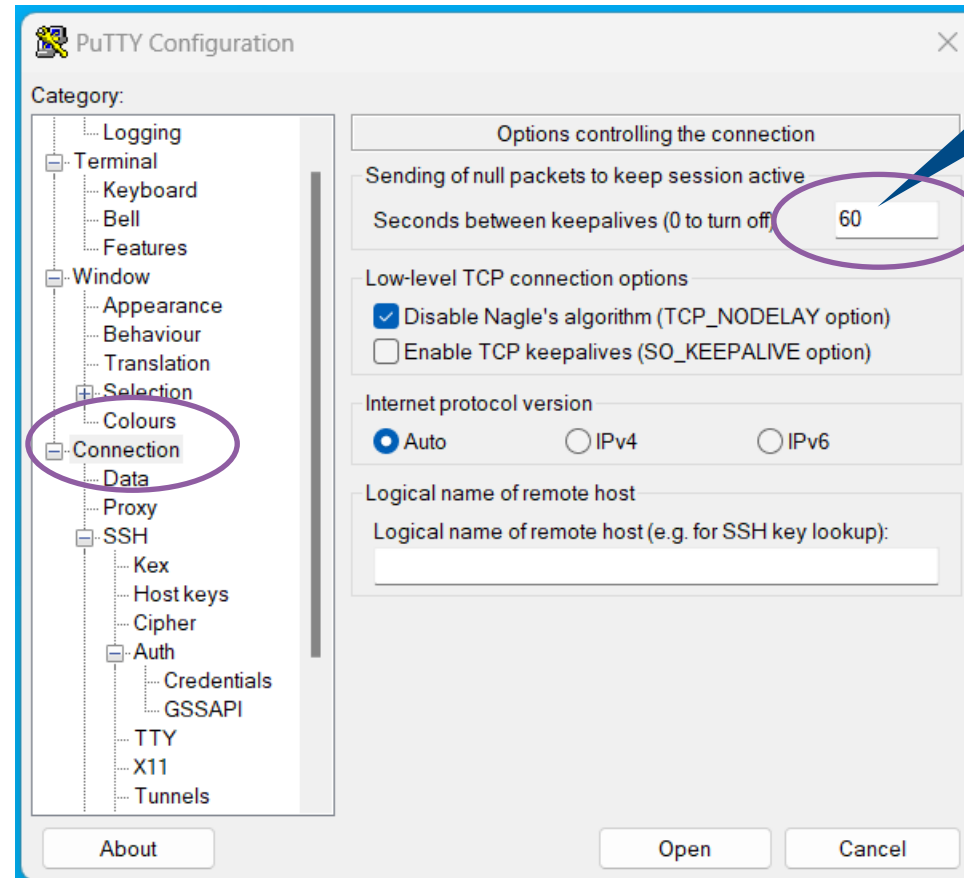
Connection with Putty – Node configuration - 2

Enter your private key again



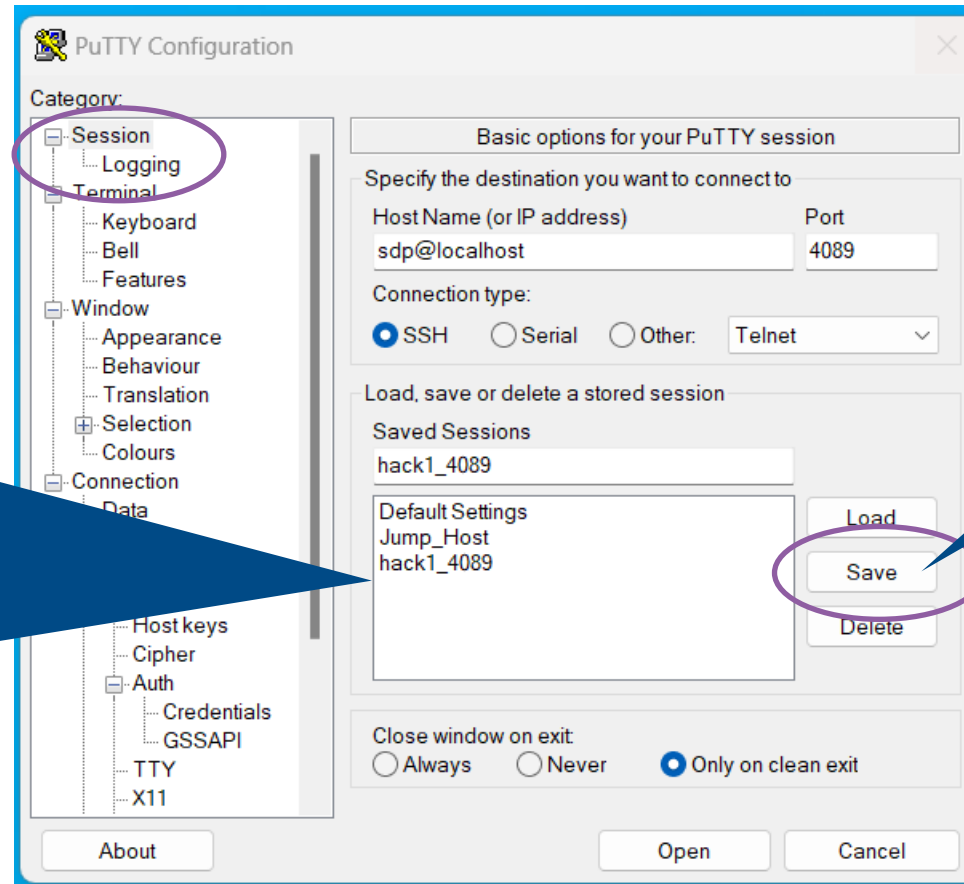
Connection with Putty – Node configuration - 3

Enter: 60



Connection with Putty – Node configuration - 4

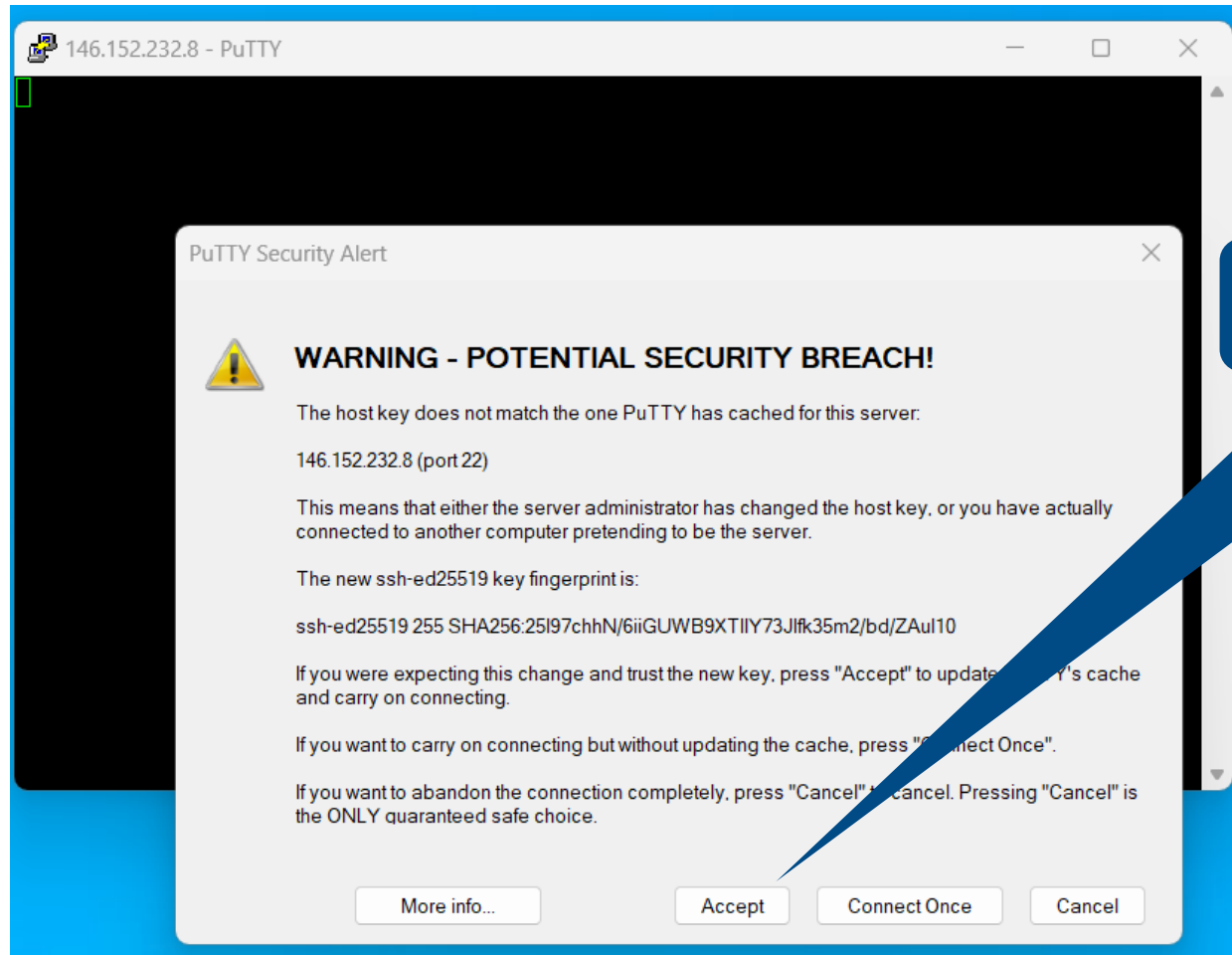
Repeat this for
hack4_4039
Port: 4039
hack3_4212
Port: 4212
hack2_4210
Port: 4210
(Load hack1_4089,
change port, change
name, click save)



Do not forget to save!

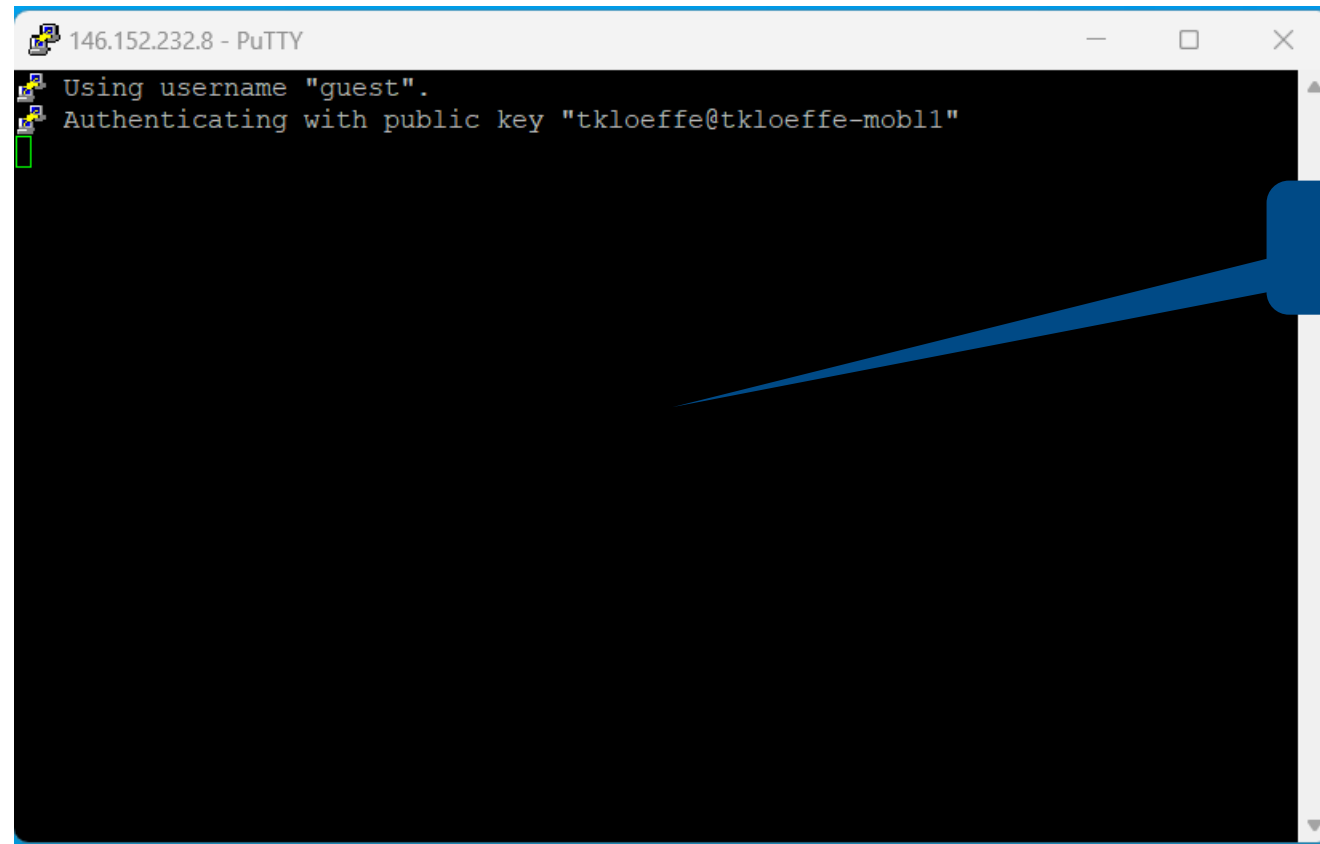
Connection with Putty – Connect to Jump_Host - 1

Load Jump_Host
click 'Open'



Accept

Connection with Putty – Connect to Jump_Host - 2

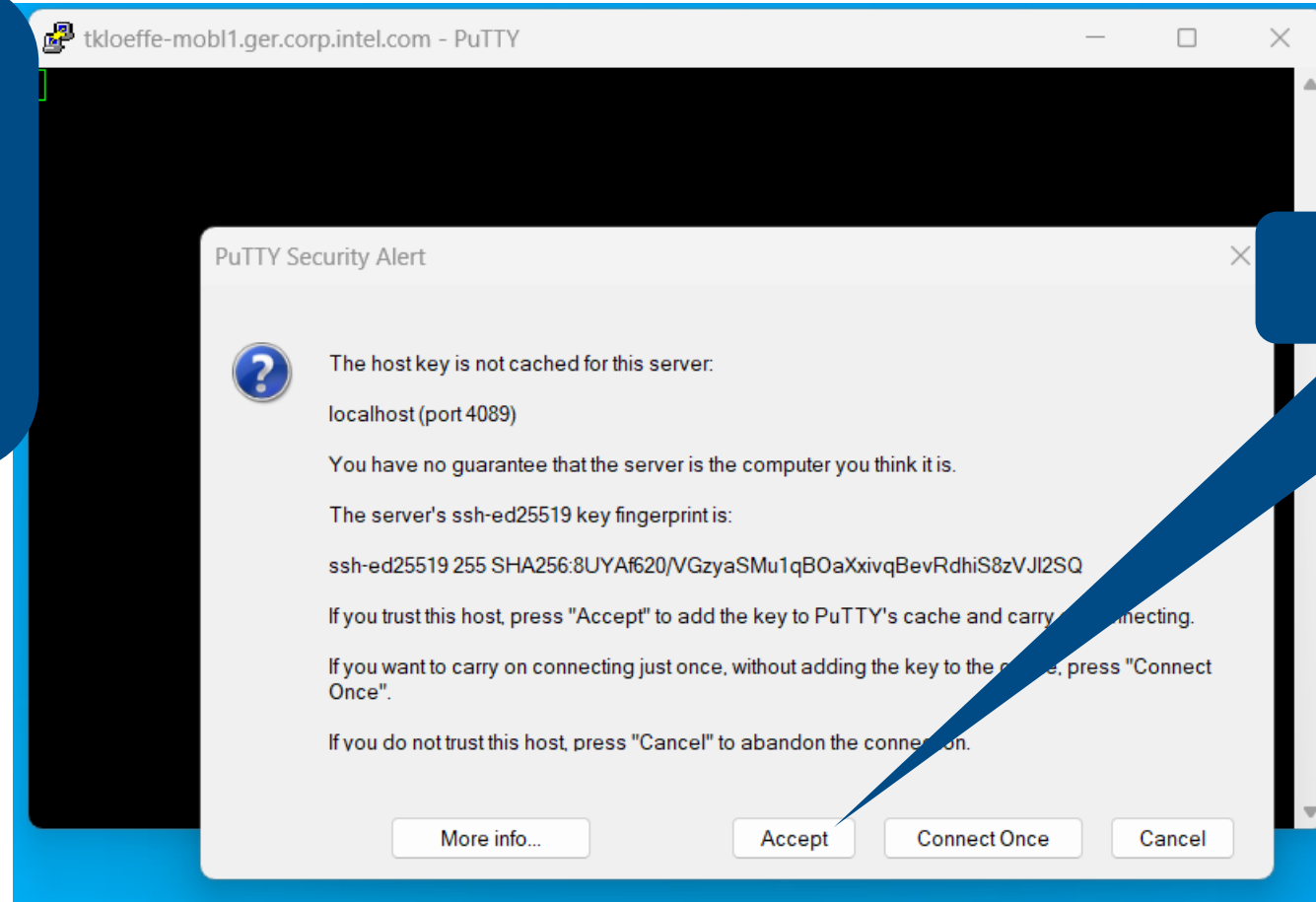


A screenshot of a PuTTY terminal window titled "146.152.232.8 - PuTTY". The terminal displays the following text: "Using username 'guest'." followed by "Authenticating with public key 'tkloeffe@tkloeffe-mobl1'". A green cursor is visible on the line following the second message.

Never close this window!

Connection with Putty – Connect to node - 1

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



Connection with Putty – Connect to node - 1

```
sdp@hack1: ~  
IMAGE_NAME="intel/intel-extension-for-tensorflow"  
IMAGE_TAG="gpu"  
docker pull "$IMAGE_NAME:$IMAGE_TAG"  
  
4. Run the TensorFlow Docker image:  
docker run -it --rm \  
-v "$CURRENT_DIR":/workspace \  
-v /dev/dri/by-path:/dev/dri/by-path \  
--device /dev/dri \  
--privileged \  
"$IMAGE_NAME:$IMAGE_TAG"  
  
Once inside the Docker container, you can run your desired AI workloads.  
  
System Information:  
Hostname: hack1  
CPU: Intel(R) Xeon(R) Platinum 8480+  
RAM: 503Gi  
Disk Space: /dev/nvme0n1p1 6.8T 1%  
Available GPU: Intel Corporation Device 0bda (rev 2f) 3a:00 Display controller  
-----  
Last login: Mon Sep 11 21:13:54 2023 from 100.64.17.5  
sdp@hack1:~$
```

hack1-hack4

Hurray!

intel®