

# Getting Started

Everything you need to access and begin using TAU HPC resources.

- [Access & Setup](#)
  - [Accessing the System](#)
  - [Palo Alto VPN](#)
- [Environment](#)
  - [Environment Modules](#)
  - [Storage and Scratch](#)
- [First Steps](#)
  - [Finding Your Account and Partition](#)
  - [Submitting Your First Job](#)
  - [Useful Tools](#)

# Access & Setup

How to connect to the HPC cluster and set up your environment.

# Accessing the System

## Requirements

- You must be a member of the **power** group
- Valid Tel Aviv University username and password
- If connecting from outside the TAU network: [TAU VPN](#) must be active

## Login Node

All access to the cluster is via SSH through the login node:

```
slurmlogin.tau.ac.il
```

Your connection is automatically load-balanced across **powerslurm-login**, **powerslurm-login2**, and **powerslurm-login3**.

## Connecting via SSH

```
ssh your_username@slurmlogin.tau.ac.il
```

With an SSH key:

```
ssh -i /path/to/your/private_key your_username@slurmlogin.tau.ac.il
```

## Important

- Do **not** run compute jobs on the login node — use `sbatch` or `srun`
- The login node is shared — heavy processes will be killed

# Palo Alto VPN

TAU uses Palo Alto GlobalProtect VPN with two-factor authentication (Google Authenticator).

Required if connecting to the cluster from outside the TAU network.

## Enrollment

1. Go to <https://mytau.tau.ac.il/GetResource.php> and register your mobile phone
2. Install **Google Authenticator** on your mobile device
3. Scan the QR code provided during enrollment

## Download

Download the appropriate version for your system:

- [PanGPLinux-6.2.9-c4.tgz](#)
- [PanGPLinux-6.3.3-c22.tgz](#)

## Install

### RHEL/Rocky/CentOS:

```
tar -xzf PanGPLinux-6.x.x-cx.tgz
yum localinstall GlobalProtect_UI_rpm-*.rpm
```

### Debian/Ubuntu:

```
tar -xzf PanGPLinux-6.x.x-cx.tgz
dpkg -i GlobalProtect_UI_deb-*.deb
```

# Configure

1. Open the GlobalProtect client
2. Enter gateway address: **vpn.tau.ac.il**
3. Log in with your TAU credentials
4. Enter the code from Google Authenticator when prompted

## Troubleshooting: SSL Error on Ubuntu 22.04+

If you see an SSL error after connecting, apply this fix:

Open `/usr/lib/ssl/openssl.cnf` and add:

```
[openssl_init]
ssl_conf = ssl_sect

[ssl_sect]
system_default = system_default_sect

[system_default_sect]
Options = UnsafeLegacyRenegotiation
```

Restart the GlobalProtect app.

# Environment

Managing software modules and storage on the cluster.

# Environment Modules

Environment Modules let you dynamically load and unload software packages without conflicts between versions. Always load the modules your job needs before running it.

## Common Commands

### Finding Modules

```
# List all available modules
module avail

# Search for a specific module
module avail gcc

# Get detailed info including dependencies
module spider gcc/gcc-12.1.0
```

### Loading and Unloading

```
# Load a module
module load gcc/gcc-12.1.0

# List currently loaded modules
module list

# Unload a specific module
module unload gcc/gcc-12.1.0

# Unload all modules
module purge
```

# Inspecting a Module

```
# See what environment variables a module will set
module show gcc/gcc-12.1.0
```

## In Job Scripts

Load modules inside your job script, after the `#SBATCH` directives:

```
#!/bin/bash
#SBATCH --job-name=my_job
#SBATCH ...

module purge
module load gcc/gcc-12.1.0

./my_program
```

Starting with `module purge` ensures your job is not affected by any modules loaded in your shell session.

# Storage and Scratch

## Home Directory

Every user has a personal home directory at `/home/your_username`. This is your default working directory when you log in.

- Backed up via Legato — see [TAU backup info](#)
- For purchasing additional storage, see [storage pricing](#)
- NetApp storage includes snapshots for file recovery

## Scratch Partitions

Scratch partitions are shared, high-speed temporary storage available across the cluster:

- `/scratch100`
- `/scratch200`
- `/scratch300`

Use scratch for intermediate files during a job run — not for long-term storage.

## Local Scratch

Some compute nodes and workstations have a local `/localscratch` partition. This is node-local storage — faster than shared scratch but only accessible from that specific node.

If your job uses `/localscratch`, you must clean up after yourself. Add this to your job script:

```
export CACHEDIR=/localscratch/${USER}_${SLURM_JOB_ID}
mkdir -p $CACHEDIR

cleanup() {
```

```
rm -rf -- "$CACHEDIR" || true  
}  
trap cleanup EXIT INT TERM HUP
```

# Important

- **Scratch is not backed up** — do not store anything you cannot afford to lose
- Clean up scratch files after your job completes
- Do not use scratch as a permanent storage location

# First Steps

Your first steps on the cluster — finding your resources and submitting your first job.

# Finding Your Account and Partition

Before submitting jobs, you need to know which account and partition you have access to. These are required parameters for every job submission.

## Check Your Partitions

Run this command on the login node:

```
check_my_partitions
```

This lists all partitions and accounts you are authorized to use.

## Key Concepts

- **Account** — your billing/group account (e.g. `public-users_v2`). Required for all job submissions.
- **Partition** — the queue your job runs in (e.g. `power-general-shared-pool`). Determines which nodes are available and what resource limits apply.
- **QOS** — Quality of Service, controls priority and limits (e.g. `public`). Usually matches your partition.

## Useful Commands

```
# View all partitions and their status
sinfo

# View partition details including limits
```

```
scontrol show partition power-general-shared-pool
```

```
# View your running and pending jobs
```

```
squeue -u your_username
```

# Need Access?

If `check_my_partitions` returns nothing or you are missing a partition, contact the HPC team at [hpc@tauex.tau.ac.il](mailto:hpc@tauex.tau.ac.il).

# Submitting Your First Job

Once you have your account and partition from `check_my_partitions`, you are ready to submit your first job.

## A Minimal Job Script

Create a file called `first_job.sh`:

```
#!/bin/bash
#SBATCH --job-name=first_job
#SBATCH --account=public-users_v2
#SBATCH --partition=power-general-shared-pool
#SBATCH --qos=public
#SBATCH --time=00:10:00
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=2G
#SBATCH --output=first_job_%j.out
#SBATCH --error=first_job_%j.err

echo "Hello from $(hostname)"
echo "Job ID: $SLURM_JOB_ID"
```

## Submit It

```
sbatch first_job.sh
```

Slurm will return a job ID:

```
Submitted batch job 12345
```

# Monitor It

```
# Check job status
squeue -u your_username

# View output once the job completes
cat first_job_12345.out
```

## Job States

- **PD** — Pending, waiting for resources
- **R** — Running
- **CG** — Completing
- **CD** — Completed
- **F** — Failed

## Next Steps

Once your first job runs successfully, see [Running Jobs](#) for arrays, GPU jobs, interactive sessions, and more.

## See Also

- [HPC Helper Toolkit](#) — AI-powered tool to help with QOS configuration, job submission, and more

First Steps

# Useful Tools

External tools and resources to help with TAU HPC usage.

## HPC Helper Toolkit

An AI-powered toolkit to help with common HPC tasks including QOS configuration, job submission, and more.

[HPC Helper Toolkit for TAU](#)