



# Install K3S in NVIDIA Jetson Nano and configure it for Plexus

Docs / Install K3S in NVIDIA Jetson Nano and configure it for Plexus

## Contents

Get started

Workload submission and management

Run AI and ML applications on Plexus

Reference

Manage Plexus user accounts and permissions

Manage Plexus billing

Plexus Premium - multi-tenancy features

Add a new data center cluster to Plexus

Configure a K8s cluster to meet Plexus requirements

Install K3S in NVIDIA Jetson Nano and configure it for Plexus

Map on-premises clusters

Responses to real Plexus customer questions

How to enable full performance read capabilities in Plexus target clusters

Application container attributes

Run HPC applications on Plexus

Virtual data center

Apache Airflow demo pipelines

Distributed learning frameworks on Plexus

## Install K3S in NVIDIA Jetson Nano and configure it for Plexus

### Overview

NVIDIA® Jetson Nano™ Developer Kit is a small, powerful computer that lets you run multiple neural networks in parallel in conjunction with Plexus.

### STEP 1: Install Jetson Nano Developer Kit

Download the Jetson Nano Developer Kit SD image and write it to the Jetson SD card. For detailed instructions, see [Getting Started with Jetson Nano Developer Kit](#).

### Linux configuration

1. Configure the IP address of the server:

◦ `jetson-nano server IP`, i.e. **192.168.1.140**

2. Open the ports for NFS and API server:

◦ `"sudo ufw allow 2049 && sudo ufw allow 6443"`

Or disable the firewall:

◦ `"sudo ufw stop && sudo ufw disable"`

### STEP 2: Install Kubernetes with K3S

1. Launch K3S using a script installation, or by executing the binary:

◦ `sudo k3s-arm64 server`

2. Get kubectl in k3s-arm64:

◦ `sudo k3s kubectl get nodes`

NOTE: You can apply aliases with the following commands:

`alias kubectl="sudo /home/nano/k3s-arm64 kubectl"`

`alias kubeup="sudo /home/nano/k3s-arm64 server"`

For more information, see [K3S](#).

### Access the cluster from a computer in the same network

1. Download the **kubeconfig** file stored at `/etc/rancher/k3s/k3s.yaml` to your computer

2. Change the cluster server field of the **kubeconfig** file by using the Jetson Nano IP, i.e.

`https://192.168.1.140:6443`

### STEP 3: Configure K8S

1. Install nvidia-docker

2. Enable GPU Support by using k8s-device-plugin

3. Enable the GPU Kubernetes plugin in K3S over ARM:

◦ Click the **DOWNLOAD** button below to download **nvidia-device-plugin-arm.yaml**



◦ `Kubectl apply nvidia-device-plugin-arm.yaml`

6. Label the node **GPU Plexus node**:

◦ `kubectl label node node-name node-role.kubernetes.io/plexus-worker-type=plexus-gpu-worker`

### STEP 4: Install NFS server

Enter the following commands in your Ubuntu NFS Storage Server terminal:

1. `sudo apt-get install nfs-kernel-server`

2. `mkdir -p /nfs && chown nobody:nogroup /nfs`

3. Add the following string to the `/etc/exports` file:

◦ `"/nfs <IP_ADDRESS1>(rw,sync,no_subtree_check) <IP_ADDRESS2>(rw,sync,no_subtree_check)"`

4. Update the NFS table:

◦ `exportfs -ra`

### STEP 5: Launch NFS client daemonset

1. Click the **DOWNLOAD** button below to download **nfs-client.yaml**



2. `Kubectl apply nfs-client.yaml`

**NOTE:** You call allow access from other nodes by opening the following port:

`sudo ufw allow 2049`

For detailed instructions, see [NFS Storage](#).

## About Core Scientific

Core Scientific is a leader in infrastructure and software solutions for Artificial Intelligence and Blockchain.

